IEC 61131-9

Edition 2.0 2022-05

INTERNATIONAL STANDARD

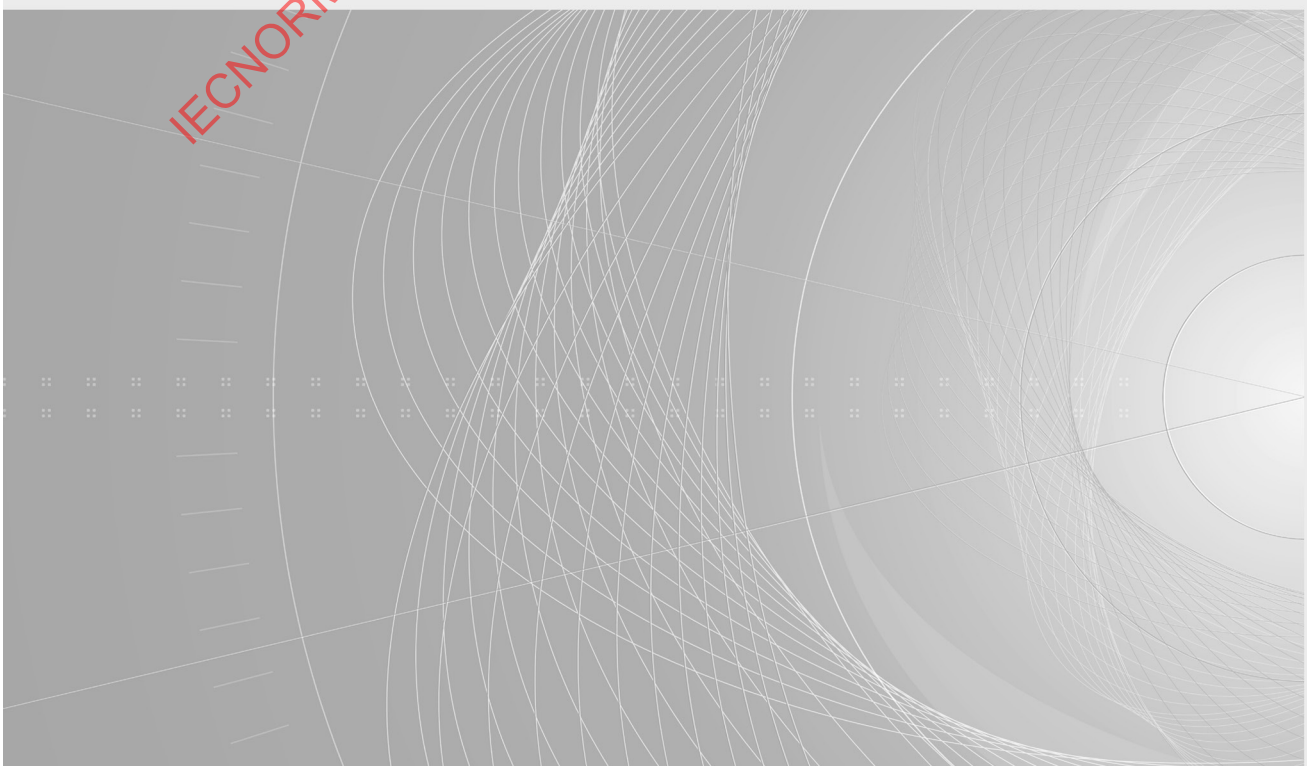
NORME INTERNATIONALE

colour inside

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Programmable controllers – Part 9: Single-drop digital communication interface for small sensors and actuators (SDCI)

Automates programmables – Partie 9: Interface de communication numérique point à point pour petits capteurs et actionneurs (SDCI)



IEC 61131-9:2022-05(en-fr)

THIS PUBLICATION IS COPYRIGHT PROTECTED
Copyright © 2022 IEC, Geneva, Switzerland

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

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

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

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

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

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

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

IEC Just Published - webstore.iec.ch/justpublished
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

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

IEC Products & Services Portal - products.iec.ch
Discover our powerful search engine and read freely all the publications previews. With a subscription you will always have access to up to date content tailored to your needs.

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

A propos de l'IEC
La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC
Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Recherche de publications IEC - webstore.iec.ch/advsearchform
La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études, …). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished
Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et une fois par mois par email.

Service Clients - webstore.iec.ch/csc
Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: sales@iec.ch.

IEC Products & Services Portal - products.iec.ch
Découvrez notre puissant moteur de recherche et consultez gratuitement tous les aperçus des publications. Avec un abonnement, vous aurez toujours accès à un contenu à jour adapté à vos besoins.

Electropedia - www.electropedia.org
Le premier dictionnaire d'électrotechnologie en ligne au monde, avec plus de 22 300 articles terminologiques en anglais et en français, ainsi que les termes équivalents dans 19 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

IEC 61131-9

Edition 2.0 2022-05

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Programmable controllers – Part 9: Single-drop digital communication interface for small sensors and actuators (SDCI)

Automates programmables – Partie 9: Interface de communication numérique point à point pour petits capteurs et actionneurs (SDCI)

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

ICS 25.040.40; 35.240.50

ISBN 978-2-8322-1261-5

Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.

® Registered trademark of the International Electrotechnical Commission
Marque déposée de la Commission Electrotechnique Internationale

CONTENTS

FOREWORD 20
INTRODUCTION 22
0.1 General 22
0.2 Patent declaration 23
1 Scope 24
2 Normative references 24
3 Terms, definitions, symbols, abbreviated terms, and conventions 25
3.1 Terms and definitions 25
3.2 Symbols and abbreviated terms 29
3.3 Conventions 31
3.3.1 General 31
3.3.2 Service parameters 32
3.3.3 Service procedures 32
3.3.4 Service attributes 32
3.3.5 Figures 33
3.3.6 Tables 33
3.3.7 Transmission octet order 33
3.3.8 Behavioral descriptions 33
4 Overview of SDCI (IO-Link™) 35
4.1 Purpose of technology 35
4.2 Positioning within the automation hierarchy 35
4.3 Wiring, connectors and power 36
4.4 Communication features of SDCI 36
4.5 Role of a Master 39
4.6 SDCI configuration 40
4.7 Mapping to fieldbuses and/or other upper-level systems 40
4.8 Standard structure 40
5 Physical Layer (PL) 41
5.1 General 41
5.1.1 Basics 41
5.1.2 Topology 41
5.2 Physical layer services 42
5.2.1 Overview 42
5.2.2 PL services 43
5.3 Transmitter/Receiver 45
5.3.1 Description method 45
5.3.2 Electrical requirements 45
5.3.3 Timing requirements 50
5.4 Power supply 54
5.4.1 Power supply options 54
5.4.2 Port Class B 55
5.4.3 Power-on requirements 56
5.5 Medium 56
5.5.1 Connectors 56
5.5.2 Cable 57
6 Standard Input and Output (SIO) 58

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7 Data link layer (DL) 59
7.1 General 59
7.2 Data link layer services 60
7.2.1 DL-B services 60
7.2.2 DL-A services 71
7.3 Data link layer protocol 75
7.3.1 Overview 75
7.3.2 DL-mode handler 76
7.3.3 Message handler 84
7.3.4 Process Data handler 92
7.3.5 On-request Data handler 95
7.3.6 ISDU handler 98
7.3.7 Command handler 102
7.3.8 Event handler 105
8 Application layer (AL) 108
8.1 General 108
8.2 Application layer services 109
8.2.1 AL services within Master and Device 109
8.2.2 AL Services 110
8.3 Application layer protocol 118
8.3.1 Overview 118
8.3.2 On-request Data transfer 118
8.3.3 Event processing 124
8.3.4 Process Data cycles 127
9 System Management (SM) 128
9.1 General 128
9.2 System Management of the Master 128
9.2.1 Overview 128
9.2.2 SM Master services 130
9.2.3 SM Master protocol 135
9.3 System Management of the Device 143
9.3.1 Overview 143
9.3.2 SM Device services 145
9.3.3 SM Device protocol 151
10 Device 158
10.1 Overview 158
10.2 Process Data Exchange (PDE) 159
10.3 Parameter Manager (PM) 159
10.3.1 General 159
10.3.2 Parameter Manager state machine 159
10.3.3 Dynamic parameter 162
10.3.4 Single parameter 162
10.3.5 Block Parameter 163
10.3.6 Concurrent parameterization access 166
10.3.7 Command handling 166
10.4 Data Storage (DS) 167
10.4.1 General 167
10.4.2 Data Storage state machine 167

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.4.3 DS configuration 169
10.4.4 DS memory space 169
10.4.5 DS Index_List 169
10.4.6 DS parameter availability 170
10.4.7 DS without ISDU 170
10.4.8 DS parameter change indication 170
10.5 Event Dispatcher (ED) 170
10.6 Device features 170
10.6.1 General 170
10.6.2 Device backward compatibility 170
10.6.3 Protocol revision compatibility 171
10.6.4 Visual SDCI indication 171
10.6.5 Parameter access locking 171
10.6.6 Data Storage locking 171
10.6.7 Locking of local parameter entries 171
10.6.8 Locking of local user interface 171
10.6.9 Offset time 171
10.6.10 Data Storage concept 172
10.6.11 Block Parameter 172
10.7 Device reset options 172
10.7.1 Overview 172
10.7.2 Device reset 173
10.7.3 Application reset 173
10.7.4 Restore factory settings 174
10.7.5 Back-to-box 174
10.8 Device design rules and constraints 174
10.8.1 General 174
10.8.2 Process Data 174
10.8.3 Communication loss 175
10.8.4 Direct Parameter 175
10.8.5 ISDU communication channel 175
10.8.6 DeviceID rules related to Device variants 175
10.8.7 Protocol constants 176
10.9 IO Device description (IODD) 176
10.10 Device diagnosis 176
10.10.1 Concepts 176
10.10.2 Events 177
10.10.3 Visual indicators 178
10.11 Device connectivity 179
11 Master 179
11.1 Overview 179
11.1.1 Positioning of Master and Gateway Applications 179
11.1.2 Structure, applications, and services of a Master 180
11.1.3 Object view of a Master and its Ports 181
11.2 Services of the Standardized Master Interface (SMI) 182
11.2.1 Overview 182
11.2.2 Structure of SMI service arguments 183
11.2.3 Concurrency and prioritization of SMI services 184
11.2.4 SMI_MasterIdentification 185

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.2.5 SMI_PortConfiguration 186
11.2.6 SMI_ReadbackPortConfiguration 187
11.2.7 SMI_PortStatus 189
11.2.8 SMI_DSToParServ 190
11.2.9 SMI_ParServToDS 191
11.2.10 SMI_DeviceWrite 193
11.2.11 SMI_DeviceRead 194
11.2.12 SMI_ParamWriteBatch 195
11.2.13 SMI_ParamReadBatch 197
11.2.14 SMI_PortPowerOffOn 198
11.2.15 SMI_DeviceEvent 200
11.2.16 SMI_PortEvent 201
11.2.17 SMI_PDIn 201
11.2.18 SMI_PDOut 203
11.2.19 SMI_PDInOut 204
11.2.20 SMI_PDInIQ 205
11.2.21 SMI_PDOutIQ 207
11.2.22 SMI_PDReadbackOutIQ 208
11.3 Configuration Manager (CM) 209
11.3.1 Coordination of Master applications 209
11.3.2 State machine of the Configuration Manager 212
11.4 Data Storage (DS) 217
11.4.1 Overview 217
11.4.2 DS data object 217
11.4.3 Backup and Restore 217
11.4.4 DS state machine 217
11.4.5 Parameter selection for Data Storage 224
11.5 On-request Data exchange (ODE) 224
11.6 Diagnosis Unit (DU) 225
11.6.1 General 225
11.6.2 Device specific Events 226
11.6.3 Port specific Events 226
11.6.4 Dynamic diagnosis status 227
11.6.5 Best practice recommendations 227
11.7 PD Exchange (PDE) 228
11.7.1 General 228
11.7.2 Process Data input mapping 228
11.7.3 Process Data output mapping 229
11.7.4 Process Data invalid/valid qualifier status 230
12 Holistic view on Data Storage 231
12.1 User point of view 231
12.2 Operations and preconditions 232
12.2.1 Purpose and objectives 232
12.2.2 Preconditions for the activation of the Data Storage mechanism 232
12.2.3 Preconditions for the types of Devices to be replaced 232
12.2.4 Preconditions for the parameter sets 232
12.3 Commissioning 233
12.3.1 On-line commissioning 233
12.3.2 Off-site commissioning 233

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

12.4 Backup Levels 234
12.4.1 Purpose 234
12.4.2 Overview 234
12.4.3 Commissioning ("Disable") 235
12.4.4 Production ("Backup/Restore") 235
12.4.5 Production ("Restore") 236
12.5 Use cases 236
12.5.1 Device replacement ("Backup/Restore") 236
12.5.2 Device replacement ("Restore") 236
12.5.3 Master replacement 236
12.5.4 Project replication 237
13 Integration 237
13.1 Generic Master model for system integration 237
13.2 Role of gateway applications 238
13.2.1 Clients 238
13.2.2 Coordination 238
13.3 Security 238
13.4 Special gateway applications 238
13.4.1 Changing Device configuration including Data Storage 238
13.4.2 Parameter server and recipe control 239
13.5 Port and Device Configuration Tool (PDCT) 239
13.5.1 Strategy 239
13.5.2 Accessing Masters via SMI 239
13.5.3 Basic layout examples 239
Annex A (normative) Codings, timing constraints, and errors 241
A.1 General structure and encoding of M-sequences 241
A.1.1 Overview 241
A.1.2 M-sequence control (MC) 241
A.1.3 Checksum/M-sequence type (CKT) 242
A.1.4 User data (PD or OD) 242
A.1.5 Checksum/status (CKS) 243
A.1.6 Calculation of the checksum 244
A.2 M-sequence types 245
A.2.1 Overview 245
A.2.2 M-sequence TYPE_0 245
A.2.3 M-sequence TYPE_1_x 245
A.2.4 M-sequence TYPE_2_x 247
A.2.5 M-sequence type 3 249
A.2.6 M-sequence type usage for STARTUP, PREOPERATE and OPERATE modes 249
A.3 Timing constraints 251
A.3.1 General 251
A.3.2 Bit time 251
A.3.3 UART frame transmission delay of Master (Ports) 251
A.3.4 UART frame transmission delay of Devices 251
A.3.5 Response time of Devices 251
A.3.6 M-sequence time 251
A.3.7 Cycle time 252
A.3.8 Idle time 253

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.3.9 Recovery time 253
A.4 Errors and remedies 253
A.4.1 UART errors 253
A.4.2 Wake-up errors 253
A.4.3 Transmission errors 253
A.4.4 Protocol errors 254
A.5 General structure and encoding of ISDUs 254
A.5.1 Overview 254
A.5.2 I-Service 254
A.5.3 Extended length (ExtLength) 255
A.5.4 Index and Subindex 256
A.5.5 Data 256
A.5.6 Check ISDU (CHKPDU) 256
A.5.7 ISDU examples 257
A.6 General structure and encoding of Events 259
A.6.1 General 259
A.6.2 StatusCode type 1 (no details) 259
A.6.3 StatusCode type 2 (with details) 260
A.6.4 EventQualifier 261
A.6.5 EventCode 262
Annex B (normative) Parameters and commands 263
B.1 Direct Parameter pages 1 and 2 263
B.1.1 Overview 263
B.1.2 MasterCommand 264
B.1.3 MasterCycleTime and MinCycleTime 265
B.1.4 M-sequenceCapability 266
B.1.5 RevisionID (RID) 267
B.1.6 ProcessDataIn 267
B.1.7 ProcessDataOut 268
B.1.8 VendorID (VID) 268
B.1.9 DeviceID (DID) 268
B.1.10 FunctionID (FID) 268
B.1.11 SystemCommand 269
B.1.12 Device specific Direct Parameter page 2 269
B.2 Predefined Device parameters 269
B.2.1 Overview 269
B.2.2 SystemCommand 272
B.2.3 DataStorageIndex 273
B.2.4 DeviceAccessLocks 274
B.2.5 ProfileCharacteristic 276
B.2.6 PDInputDescriptor 276
B.2.7 PDOutputDescriptor 276
B.2.8 VendorName 276
B.2.9 VendorText 276
B.2.10 ProductName 276
B.2.11 ProductID 276
B.2.12 ProductText 276
B.2.13 SerialNumber 277
B.2.14 HardwareRevision 277

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

B.2.15 FirmwareRevision 277
B.2.16 ApplicationSpecificTag 277
B.2.17 FunctionTag 277
B.2.18 LocationTag 277
B.2.19 ErrorCount 277
B.2.20 DeviceStatus 277
B.2.21 DetailedDeviceStatus 278
B.2.22 ProcessDataInput 279
B.2.23 ProcessDataOutput 279
B.2.24 OffsetTime 279
B.2.25 Profile parameter (reserved) 280
B.2.26 Preferred Index 280
B.2.27 Extended Index 280
B.2.28 Profile specific Index (reserved) 280
Annex C (normative) ErrorTypes (ISDU errors) 281
C.1 General 281
C.2 Application related ErrorTypes 281
C.2.1 Overview 281
C.2.2 Device application error – no details 282
C.2.3 Index not available 282
C.2.4 Subindex not available 282
C.2.5 Service temporarily not available 282
C.2.6 Service temporarily not available – local control 282
C.2.7 Service temporarily not available – device control 282
C.2.8 Access denied 282
C.2.9 Parameter value out of range 282
C.2.10 Parameter value above limit 282
C.2.11 Parameter value below limit 282
C.2.12 Parameter length overrun 283
C.2.13 Parameter length underrun 283
C.2.14 Function not available 283
C.2.15 Function temporarily unavailable 283
C.2.16 Invalid parameter set 283
C.2.17 Inconsistent parameter set 283
C.2.18 Application not ready 283
C.2.19 Vendor specific 283
C.3 Derived ErrorTypes 283
C.3.1 Overview 283
C.3.2 Master – Communication error 284
C.3.3 Master – ISDU timeout 284
C.3.4 Device Event – ISDU error 284
C.3.5 Device Event – ISDU illegal service primitive 284
C.3.6 Master – ISDU checksum error 284
C.3.7 Master – ISDU illegal service primitive 284
C.3.8 Device Event – ISDU buffer overflow 284
C.4 SMI related ErrorTypes 285
C.4.1 Overview 285
C.4.2 ArgBlock unknown 285
C.4.3 Incorrect ArgBlock content type 285

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

C.4.4 Device not communicating 285
C.4.5 Service unknown 285
C.4.6 Process Data not accessible 285
C.4.7 Insufficient memory 285
C.4.8 Incorrect Port number 285
C.4.9 Incorrect ArgBlock length 286
C.4.10 Master busy 286
C.4.11 Inconsistent DS data 286
C.4.12 Device or Master error 286
Annex D (normative) EventCodes (diagnosis information) 287
D.1 General 287
D.2 EventCodes for Devices 287
D.3 EventCodes for Ports 289
Annex E (normative) Coding of ArgBlocks 291
E.1 General 291
E.2 MasterIdent 292
E.3 PortConfigList 293
E.4 PortStatusList 295
E.5 On-request_Data 297
E.6 DS_Data 298
E.7 DeviceParBatch 298
E.8 IndexList 299
E.9 PortPowerOffOn 299
E.10 PDIn 299
E.11 PDOut 300
E.12 PDInOut 300
E.13 PDInIQ 301
E.14 PDOutIQ 301
E.15 DeviceEvent 302
E.16 PortEvent 302
E.17 VoidBlock 302
E.18 JobError 303
Annex F (normative) Data types 304
F.1 General 304
F.2 Basic data types 304
F.2.1 General 304
F.2.2 BooleanT 304
F.2.3 UIntegerT 304
F.2.4 IntegerT 305
F.2.5 Float32T 307
F.2.6 StringT 308
F.2.7 OctetStringT 308
F.2.8 TimeT 309
F.2.9 TimeSpanT 310
F.3 Composite data types 311
F.3.1 General 311
F.3.2 ArrayT 311
F.3.3 RecordT 311
Annex G (normative) Structure of the Data Storage data object 315

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex H (normative) Master and Device conformity 316
H.1 Electromagnetic compatibility (EMC) requirements 316
H.1.1 General 316
H.1.2 Operating conditions 316
H.1.3 Performance criteria 316
H.1.4 Required immunity tests 317
H.1.5 Required emission tests 318
H.1.6 Test configurations for Master 318
H.1.7 Test configurations for Devices 320
H.2 Test strategies for conformity 321
H.2.1 Test of a Device 321
H.2.2 Test of a Master 322
Annex I (informative) Residual error probabilities 323
I.1 Residual error probability of the SDCI data integrity mechanism 323
I.2 Derivation of EMC test conditions 323
Annex J (informative) Example sequence of an ISDU transmission 325
Annex K (informative) Recommended methods for detecting parameter changes 327
K.1 CRC signature 327
K.2 Revision counter 327
Bibliography 328

Figure 1 – Example of a confirmed service 33
Figure 2 – Memory storage and transmission order for WORD based data types 33
Figure 3 – Example of a nested state 34
Figure 4 – SDCI compatibility with IEC 61131-2 35
Figure 5 – Domain of the SDCI technology within the automation hierarchy 35
Figure 6 – Generic Device model for SDCI (Master's view) 37
Figure 7 – Relationship between nature of data and transmission types 38
Figure 8 – Object transfer at the application layer level (AL) 39
Figure 9 – Logical structure of Master and Device 40
Figure 10 – Three wire connection system 41
Figure 11 – Topology of SDCI 42
Figure 12 – Physical layer (Master) 42
Figure 13 – Physical layer (Device) 43
Figure 14 – Line driver reference schematics 45
Figure 15 – Receiver reference schematics 45
Figure 16 – Reference schematics for SDCI 3-wire connection system 46
Figure 17 – Voltage level definitions 46
Figure 18 – Switching thresholds 47
Figure 19 – Inrush current and charge (example) 48
Figure 20 – Power-on timing for Power 1 50
Figure 21 – Format of an SDCI UART frame 50
Figure 22 – Eye diagram for the 'H' and 'L' detection 51
Figure 23 – Eye diagram for the correct detection of a UART frame 52
Figure 24 – Wake-up request 54

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 25 – Class A and B Port definitions 55
Figure 26 – Pin layout front view 57
Figure 27 – Reference schematics for effective line capacitance and loop resistance 58
Figure 28 – Structure and services of the data link layer (Master) 59
Figure 29 – Structure and services of the data link layer (Device) 60
Figure 30 – State machines of the data link layer 76
Figure 31 – Example of an attempt to establish communication 76
Figure 32 – Failed attempt to establish communication 77
Figure 33 – Retry strategy to establish communication 78
Figure 34 – Fallback procedure 79
Figure 35 – State machine of the Master DL-mode handler 80
Figure 36 – Submachine 1 to establish communication 80
Figure 37 – State machine of the Device DL-mode handler 83
Figure 38 – SDCI message sequences 85
Figure 39 – Overview of M-sequence types 86
Figure 40 – State machine of the Master message handler 87
Figure 41 – Submachine "Response 3" of the message handler 88
Figure 42 – Submachine "Response 8" of the message handler 88
Figure 43 – Submachine "Response 15" of the message handler 88
Figure 44 – State machine of the Device message handler 91
Figure 45 – Interleave mode for the segmented transmission of Process Data 93
Figure 46 – State machine of the Master Process Data handler 93
Figure 47 – State machine of the Device Process Data handler 94
Figure 48 – State machine of the Master On-request Data handler 96
Figure 49 – State machine of the Device On-request Data handler 97
Figure 50 – Structure of the ISDU 98
Figure 51 – State machine of the Master ISDU handler 100
Figure 52 – State machine of the Device ISDU handler 101
Figure 53 – State machine of the Master command handler 103
Figure 54 – State machine of the Device command handler 104
Figure 55 – State machine of the Master Event handler 106
Figure 56 – State machine of the Device Event handler 107
Figure 57 – Structure and services of the application layer (Master) 109
Figure 58 – Structure and services of the application layer (Device) 109
Figure 59 – OD state machine of the Master AL 119
Figure 60 – OD state machine of the Device AL 121
Figure 61 – Sequence diagram for the transmission of On-request Data 122
Figure 62 – Sequence diagram for On-request Data in case of errors 123
Figure 63 – Sequence diagram for On-request Data in case of timeout 123
Figure 64 – Event state machine of the Master AL 124
Figure 65 – Event state machine of the Device AL 125
Figure 66 – Single Event scheduling 126
Figure 67 – Sequence diagram for output Process Data 127

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 68 – Sequence diagram for input Process Data 128
Figure 69 – Structure and services of the Master System Management 129
Figure 70 – Sequence chart of the use case "Port x setup" 130
Figure 71 – Main state machine of the Master System Management 136
Figure 72 – SM Master submachine CheckCompatibility_1 138
Figure 73 – Activity for state "CheckVxy" 140
Figure 74 – Activity for state "CheckCompV10" 140
Figure 75 – Activity for state "CheckComp" 141
Figure 76 – Activity (write parameter) in state "RestartDevice" 141
Figure 77 – SM Master submachine checkSerNum_3 142
Figure 78 – Activity (check SerialNumber) for state CheckSerNum_31 143
Figure 79 – Structure and services of the System Management (Device) 144
Figure 80 – Sequence chart of the use case "INACTIVE – SIO – SDCI – SIO" 145
Figure 81 – State machine of the Device System Management 152
Figure 82 – Sequence chart of a regular Device startup 155
Figure 83 – Sequence chart of a Device startup in compatibility mode 156
Figure 84 – Sequence chart of a Device startup when compatibility fails 157
Figure 85 – Structure and services of a Device 158
Figure 86 – Parameter Manager (PM) state machine 160
Figure 87 – Positive and negative parameter checking result 162
Figure 88 – Positive Block Parameter download with Data Storage request 164
Figure 89 – Negative Block Parameter download 165
Figure 90 – Data Storage (DS) state machine 167
Figure 91 – Data Storage request message sequence 169
Figure 92 – Cycle timing 172
Figure 93 – Event flow in case of successive errors 178
Figure 94 – Device LED indicator timing 179
Figure 95 – Generic relationship of SDCI and automation technology 180
Figure 96 – Structure, applications and services of a Master 181
Figure 97 – Object model of Master and Ports 182
Figure 98 – SMI services 182
Figure 99 – Coordination of Master applications 210
Figure 100 – Sequence diagram of start-up via Configuration Manager 212
Figure 101 – State machine of the Configuration Manager 213
Figure 102 – Activity for state "CheckPortMode_0" 216
Figure 103 – Main state machine of the Data Storage mechanism 218
Figure 104 – Submachine "UpDownload_2" of the Data Storage mechanism 219
Figure 105 – Data Storage submachine "Upload_7" 220
Figure 106 – Data Storage upload sequence diagram 220
Figure 107 – Data Storage submachine "Download_10" 221
Figure 108 – Data Storage download sequence diagram 221
Figure 109 – State machine of the On-request Data Exchange 225
Figure 110 – DeviceEvent flow control 226

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 111 – Port Event flow control 226
Figure 112 – SDCI diagnosis information propagation via Events 227
Figure 113 – Principles of Process Data Input mapping 228
Figure 114 – Port Qualifier Information (PQI) 229
Figure 115 – Principles of Process Data Output mapping 230
Figure 116 – Propagation of PD qualifier status between Master and Device 231
Figure 117 – Active and backup parameter 233
Figure 118 – Off-site commissioning 233
Figure 119 – Generic Master model for system integration 238
Figure 120 – PDCT via gateway application 239
Figure 121 – Example 1 of a PDCT display layout 240
Figure 122 – Example 2 of a PDCT display layout 240
Figure A.1 – M-sequence control 241
Figure A.2 – Checksum/M-sequence type octet 242
Figure A.3 – Checksum/status octet 243
Figure A.4 – Principle of the checksum calculation and compression 244
Figure A.5 – M-sequence TYPE_0 245
Figure A.6 – M-sequence TYPE_1_1 245
Figure A.7 – M-sequence TYPE_1_2 246
Figure A.8 – M-sequence TYPE_1_V 246
Figure A.9 – M-sequence TYPE_2_1 247
Figure A.10 – M-sequence TYPE_2_2 247
Figure A.11 – M-sequence TYPE_2_3 248
Figure A.12 – M-sequence TYPE_2_4 248
Figure A.13 – M-sequence TYPE_2_5 248
Figure A.14 – M-sequence TYPE_2_V 249
Figure A.15 – M-sequence timing 252
Figure A.16 – I-Service octet 254
Figure A.17 – Check of ISDU integrity via CHKPDU 257
Figure A.18 – Examples of request formats for ISDUs 257
Figure A.19 – Examples of response ISDUs 258
Figure A.20 – Examples of read and write request ISDUs 259
Figure A.21 – Structure of StatusCode type 1 259
Figure A.22 – Structure of StatusCode type 2 260
Figure A.23 – Indication of activated Events 261
Figure A.24 – Structure of the EventQualifier 261
Figure B.1 – Classification and mapping of Direct Parameters 263
Figure B.2 – MinCycleTime 265
Figure B.3 – M-sequenceCapability 266
Figure B.4 – RevisionID 267
Figure B.5 – ProcessDataIn 267
Figure B.6 – Index space for ISDU data objects 269
Figure B.7 – Structure of the OffsetTime 279

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure E.1 – Assignment of ArgBlock identifiers 291
Figure F.1 – Coding example of small UIntegerT 305
Figure F.2 – Coding example of large UIntegerT 305
Figure F.3 – Coding examples of IntegerT 307
Figure F.4 – Singular access of StringT 308
Figure F.5 – Coding example of OctetStringT 309
Figure F.6 – Definition of TimeT 309
Figure F.7 – Example of an ArrayT data structure 311
Figure F.8 – Example 2 of a RecordT structure 313
Figure F.9 – Example 3 of a RecordT structure 313
Figure F.10 – Write requests for Example 3 314
Figure H.1 – Test setup for electrostatic discharge (Master) 319
Figure H.2 – Test setup for RF electromagnetic field (Master) 319
Figure H.3 – Test setup for fast transients (Master) 319
Figure H.4 – Test setup for RF common mode (Master) 320
Figure H.5 – Test setup for electrostatic discharges (Device) 320
Figure H.6 – Test setup for RF electromagnetic field (Device) 321
Figure H.7 – Test setup for fast transients (Device) 321
Figure H.8 – Test setup for RF common mode (Device) 321
Figure I.1 – Residual error probability for the SDCI data integrity mechanism 323
Figure J.1 – Example for ISDU transmissions 325

Table 1 – Service assignments of Master and Device 43
Table 2 – PL_SetMode 43
Table 3 – PL_WakeUp 44
Table 4 – PL_Transfer 44
Table 5 – Electrical characteristics of a receiver 47
Table 6 – Electrical characteristics of a Master Port 48
Table 7 – Electrical characteristics of a Device 49
Table 8 – Power-on timing 50
Table 9 – Dynamic characteristics of the transmission 52
Table 10 – Wake-up request characteristics 54
Table 11 – Electrical characteristics of a Master Port Class B 55
Table 12 – Master pin assignments 56
Table 13 – Device pin assignments 57
Table 14 – Cable characteristics 58
Table 15 – Cable conductor assignments 58
Table 16 – Service assignments within Master and Device 61
Table 17 – DL_ReadParam 61
Table 18 – DL_WriteParam 62
Table 19 – DL_Read 63
Table 20 – DL_Write 63
Table 21 – DL_ISDUTransport 64

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 22 – DL_ISDUAbort 65
Table 23 – DL_PDOutputUpdate 65
Table 24 – DL_PDOutputTransport 66
Table 25 – DL_PDInputUpdate 66
Table 26 – DL_PDInputTransport 67
Table 27 – DL_PDCycle 67
Table 28 – DL_SetMode 68
Table 29 – DL_Mode 69
Table 30 – DL_Event 69
Table 31 – DL_EventConf 70
Table 32 – DL_EventTrigger 70
Table 33 – DL_Control 70
Table 34 – DL-A services within Master and Device 71
Table 35 – OD 72
Table 36 – PD 73
Table 37 – EventFlag 74
Table 38 – PDInStatus 74
Table 39 – MHInfo 74
Table 40 – ODTrig 75
Table 41 – PDTrig 75
Table 42 – Wake-up procedure and retry characteristics 78
Table 43 – Fallback timing characteristics 79
Table 44 – State transition table of the Master DL-mode handler 81
Table 45 – State transition table of the Device DL-mode handler 83
Table 46 – State transition table of the Master message handler 89
Table 47 – State transition table of the Device message handler 92
Table 48 – State transition table of the Master Process Data handler 94
Table 49 – State transition table of the Device Process Data handler 95
Table 50 – State transition table of the Master On-request Data handler 96
Table 51 – State transition table of the Device On-request Data handler 97
Table 52 – FlowCTRL definitions 99
Table 53 – State transition table of the Master ISDU handler 100
Table 54 – State transition table of the Device ISDU handler 102
Table 55 – Control codes 103
Table 56 – State transition table of the Master command handler 103
Table 57 – State transition table of the Device command handler 105
Table 58 – Event memory 105
Table 59 – State transition table of the Master Event handler 107
Table 60 – State transition table of the Device Event handler 108
Table 61 – AL services within Master and Device 110
Table 62 – AL_Read 110
Table 63 – AL_Write 111
Table 64 – AL_Abort 112

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 65 – AL_GetInput 113
Table 66 – AL_NewInput 114
Table 67 – AL_SetInput 114
Table 68 – AL_PDCycle 114
Table 69 – AL_GetOutput 115
Table 70 – AL_NewOutput 115
Table 71 – AL_SetOutput 116
Table 72 – AL_Event 117
Table 73 – AL_Control 118
Table 74 – States and transitions for the OD state machine of the Master AL 119
Table 75 – States and transitions for the OD state machine of the Device AL 121
Table 76 – State and transitions of the Event state machine of the Master AL 124
Table 77 – State and transitions of the Event state machine of the Device AL 125
Table 78 – SM services within the Master 131
Table 79 – SM_SetPortConfig 131
Table 80 – Definition of the InspectionLevel (IL) 132
Table 81 – Definitions of the Target Modes 132
Table 82 – SM_GetPortConfig 133
Table 83 – SM_PortMode 134
Table 84 – SM_Operate 135
Table 85 – State transition table of the Master System Management 137
Table 86 – State transition table of the Master submachine CheckCompatibility_1 139
Table 87 – State transition table of the Master submachine checkSerNum_3 142
Table 88 – SM services within the Device 146
Table 89 – SM_SetDeviceCom 146
Table 90 – SM_GetDeviceCom 147
Table 91 – SM_SetDeviceIdent 148
Table 92 – SM_GetDeviceIdent 149
Table 93 – SM_SetDeviceMode 150
Table 94 – SM_DeviceMode 150
Table 95 – State transition table of the Device System Management 153
Table 96 – State transition table of the PM state machine 160
Table 97 – Sequence of parameter checks 163
Table 98 – Steps and rules for Block Parameter checking 165
Table 99 – Prioritized ISDU responses on command parameters 166
Table 100 – State transition table of the Data Storage state machine 168
Table 101 – Overview on reset options and their impact on Devices 173
Table 102 – Overview of the protocol constants for Devices 176
Table 103 – Classification of Device diagnosis incidents 177
Table 104 – Timing for LED indicators 179
Table 105 – SMI services 183
Table 106 – SMI_MasterIdentification 185
Table 107 – SMI_PortConfiguration 186

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 108 – SMI_ReadbackPortConfiguration 188
Table 109 – SMI_PortStatus 189
Table 110 – SMI_DSToParServ 190
Table 111 – SMI_ParServToDS 192
Table 112 – SMI_DeviceWrite 193
Table 113 – SMI_DeviceRead 194
Table 114 – SMI_ParamWriteBatch 196
Table 115 – SMI_ParamReadBatch 197
Table 116 – SMI_PortPowerOffOn 199
Table 117 – SMI_DeviceEvent 200
Table 118 – SMI_PortEvent 201
Table 119 – SMI_PDIn 202
Table 120 – SMI_PDOut 203
Table 121 – SMI_PDInOut 204
Table 122 – SMI_PDInIQ 206
Table 123 – SMI_PDOutIQ 207
Table 124 – SMI_PDReadbackOutIQ 208
Table 125 – Internal variables and Events controlling Master applications 210
Table 126 – State transition table of the Configuration Manager 213
Table 127 – States and transitions of the Data Storage state machines 222
Table 128 – State transition table of the ODE state machine 225
Table 129 – Recommended Data Storage Backup Levels 234
Table 130 – Criteria for backing up parameters ("Backup/Restore") 235
Table 131 – Criteria for backing up parameters ("Restore") 236
Table A.1 – Values of communication channel 241
Table A.2 – Values of R/W 242
Table A.3 – Values of M-sequence types 242
Table A.4 – Data types for user data 243
Table A.5 – Values of PD status 243
Table A.6 – Values of the Event flag 244
Table A.7 – M-sequence types for the STARTUP mode 249
Table A.8 – M-sequence types for the PREOPERATE mode 249
Table A.9 – M-sequence types for the OPERATE mode (legacy protocol) 250
Table A.10 – M-sequence types for the OPERATE mode 250
Table A.11 – Recommended MinCycleTimes 252
Table A.12 – Definition of the nibble "I-Service" 255
Table A.13 – ISDU syntax 255
Table A.14 – Definition of nibble Length and octet ExtLength 256
Table A.15 – Use of Index formats 256
Table A.16 – Mapping of EventCodes (type 1) 260
Table A.17 – Values of INSTANCE 261
Table A.18 – Values of SOURCE 262
Table A.19 – Values of TYPE 262

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table A.20 – Values of MODE 262
Table B.1 – Direct Parameter pages 1 and 2 264
Table B.2 – Types of MasterCommands 265
Table B.3 – Possible values of MasterCycleTime and MinCycleTime 266
Table B.4 – Values of ISDU 266
Table B.5 – Values of SIO 268
Table B.6 – Permitted combinations of BYTE and Length 268
Table B.7 – Implementation rules for parameters and commands 270
Table B.8 – Index assignment of data objects (Device parameter) 270
Table B.9 – Coding of SystemCommand (ISDU) 272
Table B.10 – DataStorageIndex assignments 273
Table B.11 – Structure of Index_List 274
Table B.12 – Device locking possibilities 275
Table B.13 – DeviceStatus parameter 278
Table B.14 – DetailedDeviceStatus (Index 0x0025) 279
Table B.15 – Time base coding and values of OffsetTime 280
Table C.1 – ErrorTypes 281
Table C.2 – Derived ErrorTypes 284
Table C.3 – SMI related ErrorTypes 285
Table D.1 – EventCodes for Devices 287
Table D.2 – EventCodes for Ports 289
Table E.1 – ArgBlock types and their ArgBlockIDs 292
Table E.2 – MasterIdent 293
Table E.3 – PortConfigList 294
Table E.4 – PortStatusList 296
Table E.5 – On-request_Data 297
Table E.6 – DS_Data 298
Table E.7 – DeviceParBatch 298
Table E.8 – IndexList 299
Table E.9 – PortPowerOffOn 299
Table E.10 – PDIn 300
Table E.11 – PDOut 300
Table E.12 – PDInOut 301
Table E.13 – PDInIQ 301
Table E.14 – PDOutIQ 302
Table E.15 – DeviceEvent 302
Table E.16 – PortEvent 302
Table E.17 – VoidBlock 303
Table E.18 – JobError 303
Table F.1 – BooleanT 304
Table F.2 – BooleanT coding 304
Table F.3 – UIntegerT 305
Table F.4 – IntegerT 305

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table F.5 – IntegerT coding (8 octets) 306
Table F.6 – IntegerT coding (4 octets) 306
Table F.7 – IntegerT coding (2 octets) 306
Table F.8 – IntegerT coding (1 octet) 306
Table F.9 – Float32T 307
Table F.10 – Coding of Float32T 307
Table F.11 – StringT 308
Table F.12 – OctetStringT 308
Table F.13 – TimeT 309
Table F.14 – Coding of TimeT 310
Table F.15 – TimeSpanT 310
Table F.16 – Coding of TimeSpanT 310
Table F.17 – Structuring rules for ArrayT 311
Table F.18 – Example for the access of an ArrayT 311
Table F.19 – Structuring rules for RecordT 312
Table F.20 – Example 1 for the access of a RecordT 312
Table F.21 – Example 2 for the access of a RecordT 312
Table F.22 – Example 3 for the access of a RecordT 313
Table G.1 – Structure of the stored DS data object 315
Table G.2 – Associated header information for stored DS data objects 315
Table H.1 – EMC test conditions for SDCI 317
Table H.2 – EMC test levels 317
Table K.1 – Proper CRC generator polynomials 327

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

PROGRAMMABLE CONTROLLERS –

Part 9: Single-drop digital communication interface for small sensors and actuators (SDCI)

FOREWORD

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

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

IEC 61131-9 has been prepared by subcommittee 65B: Measurement and control devices, of IEC technical committee 65: Industrial-process measurement, control and automation. It is an International Standard.

This second edition cancels and replaces the first edition published in 2013. This edition constitutes a technical revision.

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

a) requirements for Port Class B have been refined;
b) best practice patterns for Data Storage (backup of Device parameters) and Device reset functions have been introduced;
c) a new Standardized Master Interface (SMI) for PLC, IT, and an engineering tool access have been added.

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

Draft	Report on voting
65B/1218/FDIS	65B/1219/RVD

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

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

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

A list of all parts in the IEC 61131 series, published under the general title *Programmable controllers*, can be found on the IEC website.

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

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

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

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

INTRODUCTION

0.1 General

IEC 61131-9 is part of a series of standards on programmable controllers and the associated peripherals and should be read in conjunction with the other parts of the series.

Where a conflict exists between this and other IEC standards (except basic safety standards), the provisions of this document should be considered to govern in the area of programmable controllers and their associated peripherals.

The increased use of micro-controllers embedded in low-cost sensors and actuators has provided opportunities for adding diagnosis and configuration data to support increasing application requirements.

The driving force for the SDCI (IO-Link™[1]) technology is the need of these low-cost sensors and actuators to exchange this diagnosis and configuration data with a controller (PC or PLC) using a low-cost, digital communication technology while maintaining backward compatibility with the current DI/DO signals.

In fieldbus concepts, the SDCI technology defines a generic interface for connecting sensors and actuators to a Master unit, which can be combined with gateway capabilities to become a fieldbus remote I/O node.

Any SDCI compliant Device can be attached to any available interface Port of the Master. SDCI compliant Devices perform physical to digital conversion in the Device, and then communicate the result directly in a standard format using "coded switching" of the 24 V I/O signalling line, thus removing the need for different DI, DO, AI, AO modules and a variety of cables.

Physical topology is point-to-point from each Device to the Master using three wires over distances up to 20 m. The SDCI physical interface is backward compatible with the usual 24 V I/O signalling specified in IEC 61131-2. Transmission rates of 4,8 kbit/s, 38,4 kbit/s and 230,4 kbit/s are supported.

The Master of the SDCI interface detects, identifies, and manages Devices plugged into its Ports.

Tools allow the association of Devices with their corresponding electronic I/O Device Descriptions (IODD) and their subsequent configuration to match the application requirements.

The SDCI technology specifies three different levels of diagnostic capabilities: for immediate response by automated needs during the production phase, for medium term response by operator intervention, or for longer term commissioning and maintenance via extended diagnosis information.

The structure of this document is described in 4.8.

Conformity with IEC 61131-9 cannot be claimed unless the requirements of Annex H are met.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

---

[1] IO-Link™ is a trade name of the "IO-Link Community". This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the trade name holder or any of its products. Compliance with this document does not require use of the registered logos for IO-Link™. Use of the registered logos for IO-Link™ requires permission of the "IO-Link Community".

Terms of general use are defined in IEC 61131-1 or in the IEC 60050 series [1][2]. More specific terms are defined in this document.

0.2 Patent declaration

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

The holder of this patent right has assured IEC that s/he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with IEC. Information may be obtained from the patent database available at http://patents.iec.ch.

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

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

2 Numbers in square brackets refer to the Bibliography.

PROGRAMMABLE CONTROLLERS –

Part 9: Single-drop digital communication interface for small sensors and actuators (SDCI)

1 Scope

This part of IEC 61131 specifies a single-drop digital communication interface technology for small sensors and actuators SDCI (commonly known as IO-Link™[3]), which extends the traditional digital input and digital output interfaces as defined in IEC 61131-2 towards a point-to-point communication link for the exchange of complex data in both directions. This technology also enables the transfer of parameters to Devices and the delivery of identification and diagnostic information from the Devices to the automation system.

This technology is mainly intended for use with simple sensors and actuators in factory automation, which include small and cost-effective microcontrollers.

This document specifies the SDCI communication services and protocol (physical layer, data link layer and application layer in accordance with the ISO/OSI reference model) for both SDCI Masters and Devices.

This document also includes EMC test requirements.

This document does not cover communication interfaces or systems incorporating multiple point or multiple drop linkages, or integration of SDCI into higher level systems such as fieldbuses.

2 Normative references

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

IEC 60664 (all parts), *Insulation coordination for equipment within low-voltage supply systems*

IEC 60947-5-2, *Low-voltage switchgear and controlgear – Part 5-2: Control circuit devices and switching elements – Proximity switches*

IEC 61000-4-2, *Electromagnetic compatibility (EMC) – Part 4-2: Testing and measurement techniques – Electrostatic discharge immunity test*

IEC 61000-4-3, *Electromagnetic compatibility (EMC) – Part 4-3 : Testing and measurement techniques – Radiated, radio-frequency, electromagnetic field immunity test*

IEC 61000-4-4, *Electromagnetic compatibility (EMC) – Part 4-4: Testing and measurement techniques – Electrical fast transient/burst immunity test*

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

---

3 IO-Link™ is a trade name of the "IO-Link Community". This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the trade name holder or any of its products. Compliance with this document does not require use of the registered logos for IO-Link™. Use of the registered logos for IO-Link™ requires permission of the "IO-Link Community".

IEC 61000-4-5, *Electromagnetic compatibility (EMC) – Part 4-5: Testing and measurement techniques – Surge immunity test*

IEC 61000-4-6, *Electromagnetic compatibility (EMC) – Part 4-6: Testing and measurement techniques – Immunity to conducted disturbances, induced by radio-frequency fields*

IEC 61000-4-11, *Electromagnetic compatibility (EMC) – Part 4-11: Testing and measurement techniques – Voltage dips, short interruptions and voltage variations immunity tests for equipment with input current up to 16 A per phase*

IEC 61000-6-2, *Electromagnetic compatibility (EMC) – Part 6-2: Generic standards – Immunity standard for industrial environments*

IEC 61000-6-4, *Electromagnetic compatibility (EMC) – Part 6-4: Generic standards – Emission standard for industrial environments*

IEC 61076-2-101, *Connectors for electronic equipment – Product requirements – Part 2-101: Circular connectors – Detail specification for M12 connectors with screw-locking*

IEC 61010-2-201, *Safety requirements for electrical equipment for measurement, control, and laboratory use – Part 2-201: Particular requirements for control equipment*

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

IEC 61131-2, *Industrial-process measurement and control – Programmable controllers – Part 2: Equipment requirements and tests*

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3 Terms, definitions, symbols, abbreviated terms, and conventions

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 61131-1 and IEC 61131-2 and the following apply.

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

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

3.1.1
address
part of the M-sequence control to reference data within data categories of a communication channel

3.1.2
application layer
AL
<SDCI> part of the protocol responsible for the transmission of Process Data objects and On-request Data objects

3.1.3
Block Parameter
consistent parameter access via multiple Indices or Subindices

3.1.4
checksum
<SDCI> complementary part of the overall data integrity measures in the data link layer in addition to the UART parity bit

3.1.5
CHKPDU
integrity protection data within an ISDU communication channel generated through XOR processing the octets of a request or response

3.1.6
coded switching
SDCI communication, based on the standard binary signal levels of IEC 61131-2

3.1.7
COM1
SDCI communication mode with transmission rate of 4,8 kbit/s

3.1.8
COM2
SDCI communication mode with transmission rate of 38,4 kbit/s

3.1.9
COM3
SDCI communication mode with transmission rate of 230,4 kbit/s

3.1.10
COMx
one out of three possible SDCI communication modes COM1, COM2, or COM3

3.1.11
communication channel
logical connection between Master and Device

Note 1 to entry: Four communication channels are defined: process channel, page and ISDU channel (for parameters), and diagnosis channel.

3.1.12
communication error
unexpected disturbance of the SDCI transmission protocol

3.1.13
cycle time
time to transmit an M-sequence between a Master and its Device including the following idle time

3.1.14
Device
single passive peer to a Master such as a sensor or actuator

Note 1 to entry: Uppercase "Device" is used for SDCI equipment, while lowercase "device" is used in a generic manner.

3.1.15
Direct Parameter
directly (page) addressed parameter transferred acyclically via the page communication channel without acknowledgment

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.16
dynamic parameter
part of a Device's parameter set defined by on-board user interfaces such as teach-in buttons or control panels in addition to the static parameters

3.1.17
Event
instance of a change of conditions in a Device

Note 1 to entry: Uppercase "Event" is used for SDCI Events, while lowercase "event" is used in a generic manner.

Note 2 to entry: An Event is indicated via the Event flag within the Device's status cyclic information, then acyclic transfer of Event data (typically diagnosis information) is conveyed through the diagnosis communication channel.

3.1.18
fallback
transition of a Port from coded switching to switching signal mode

3.1.19
inspection level
degree of verification for the Device identity

3.1.20
interleave
segmented cyclic data exchange for Process Data with more than 2 octets through subsequent cycles

3.1.21
Indexed Service Data Unit
ISDU
data unit used for acyclic acknowledged transmission of parameters that can be segmented in a number of M-sequences

3.1.22
legacy Device
Device designed in accordance with [8]

3.1.23
legacy Master
Master designed in accordance with [8]

3.1.24
M-sequence
sequence of two messages comprising a Master message and its subsequent Device message

3.1.25
M-sequence control
first octet in a Master message indicating the read/write operation, the type of the communication channel, and the address, for example offset or flow control

3.1.26
M-sequence error
unexpected or wrong message content, or no response

3.1.27
M-sequence type
one particular M-sequence format out of a set of specified M-sequence formats

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.28
Master
active peer connected through Ports to one up to n Devices and which provides an interface to the gateway to the upper-level communication systems or PLCs

Note 1 to entry: Uppercase "Master" is used for SDCI equipment, while lowercase "master" is used in a generic manner.

3.1.29
message
<SDCI> sequence of UART frames transferred either from a Master to its Device or vice versa following the rules of the SDCI protocol

3.1.30
On-request Data
OD
acyclically transmitted data upon request of the Master application consisting of parameters or Event data

3.1.31
physical layer
first layer of the ISO-OSI reference model, which provides the mechanical, electrical, functional and procedural means to activate, maintain, and de-activate physical connections for bit transmission between data-link entities

Note 1 to entry: The physical layer also provides means for wake-up and fallback procedures.

[SOURCE: ISO/IEC 7498-1:1994 [5], 7.7.2, modified – Text extracted from 7.7.2, Note 1 to entry added]

3.1.32
Port
communication medium interface of the Master to one Device

3.1.33
Process Data
PD
input or output values from or to a discrete or continuous automation process cyclically transferred with high priority and in a configured schedule automatically between Master and Device

3.1.34
Process Data cycle
complete transfer of all Process Data from or to an individual Device that can comprise several cycles in case of segmentation (interleave)

3.1.35
single parameter
independent parameter access via one single Index or Subindex

3.1.36
SIO
Port operation mode in accordance with digital input and output defined in IEC 61131-2 that is established after power-up or fallback or unsuccessful communication attempts

3.1.37
static parameter
part of a Device's parameter set to be saved in a Master for the case of replacement without engineering tools

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.38
switching signal
binary signal from or to a Device when in SIO mode (as opposed to the "coded switching" SDCI communication)

3.1.39
System Management
SM
<SDCI> means to control and coordinate the internal communication layers and the exceptions within the Master and its Ports, and within each Device

3.1.40
UART frame
<SDCI> bit sequence starting with a start bit, followed by eight bits carrying a data octet, followed by an even parity bit and ending with one stop bit

3.1.41
wake-up
procedure for causing a Device to change its mode from SIO to SDCI

3.1.42
wake-up request
WURQ
physical layer service used by the Master to initiate wake-up of a Device, and put it in a receive ready state

3.2 Symbols and abbreviated terms

Δf_{DTRM}	permissible deviation from data transfer rate (measured in %)
ΔVS	power supply ripple (measured in V)
AL	application layer
BEP	bit error probability
C/Q	connection for communication (C) or switching (Q) signal (SIO)
CL_{eff}	effective total cable capacity (measured in nF)
CQ	input capacity at C/Q connection (measured in nF)
DI	digital input
DL	data link layer
DO	digital output
f_{DTR}	data transfer rate (measured in bit/s)
H/L	high/low signal at receiver output
I/O	input/output
ILL	input load current at input C/Q to $V0$ (measured in A)
IODD	IO Device Description (see 10.9)
$IP24_M$	extra DC supply current for Devices
IQ	driver current in saturated operating status ON (measured in A)
IQH	driver current on high-side driver in saturated operating status ON (measured in A)
IQL	driver current on low-side driver in saturated operating status ON (measured in A)
$IQPK$	maximum driver current in unsaturated operating status ON (measured in A)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

$IQPKH$	maximum driver current on high-side driver in unsaturated operating status ON (measured in A)
$IQPKL$	maximum driver current on low-side driver in unsaturated operating status ON (measured in A)
IQQ	quiescent current at input C/Q to $V0$ with inactive output drivers (measured in A)
IQ_{WU}	amplitude of Master's wake-up request current (measured in A)
IS	supply current at V+ (measured in A)
$ISIR$	current pulse supply capability at V+ (measured in A)
LED	light emitting diode
L−	power supply (−)
L+	power supply (+)
N24	24 V extra power supply (−)
NRZ	non-return to zero
n_{WU}	wake-up retry count
On/Off	driver's ON/OFF switching signal
OD	On-request Data
OVD	signal overload detect
P24	24 V extra power supply (+)
PD	Process Data
PDCT	Port and Device configuration tool
PL	physical layer
PLC	programmable logic controller
PS	power supply (measured in V)
QIS_D	power-up charge consumption
r	time to reach a stable level with reference to the beginning of the start bit (measured in T_{BIT})
RL_{eff}	loop resistance of cable (measured in Ω)
s	time to exit a stable level with reference to the beginning of the start bit (measured in T_{BIT})
SDCI	single-drop digital communication interface
SIO	standard input output (digital switching mode) [IEC 61131-2]
SM	system management
SMI	standardized Master interface
t_1	UART frame transfer delay on Master (measured in T_{BIT})
t_2	UART frame transfer delay on Device (measured in T_{BIT})
t_A	response delay on Device (measured in T_{BIT})
T_{BIT}	bit time (measured in s)
t_{CYC}	cycle time on M-sequence level (measured in s)
t_{DF}	fall time (measured in s)
T_{DMT}	delay time while establishing Master Port communication (measured in T_{BIT})
T_{DR}	rise time (measured in s)

IECNORM.COM Click to view the full PDF of IEC 61131-9:2022

T_{DSIO}	delay time on Device for transition to SIO mode following wake-up request (measured in s)
T_{DWU}	wake-up retry delay (measured in s)
$t_{M\text{-}sequence}$	M-sequence duration (measured in T_{BIT})
t_{idle}	idle time between two M-sequences (measured in s)
t_H	detection time for high level (measured in s)
t_L	detection time for low level (measured in s)
t_{ND}	noise suppression time (measured in s)
T_{RDL}	wake-up readiness following power ON (measured in s)
T_{REN}	receive enable (measured in s)
T_{SD}	device detect time (measured in s)
T_{WU}	pulse duration of wake-up request (measured in s)
UART	universal asynchronous receiver transmitter
UML	Unified Modelling Language [ISO/IEC 19505 series]
$V+$	voltage at L+
$V0$	voltage at L−
$VD+_L$	voltage drop on the line between the L+ connections on Master and Device (measured in V)
$VD0_L$	voltage drop on the line between the L− connections on Master and Device (measured in V)
VDQ_L	voltage drop on the line between the C/Q connections on Master and Device (measured in V)
$VHYS$	hysteresis of receiver threshold voltage (measured in V)
VI	input voltage at connection C/Q with reference to $V0$ (measured in V)
VIH	input voltage range at connection C/Q for high signal (measured in V)
VIL	input voltage range at connection C/Q for low signal (measured in V)
$VP24_M$	extra DC supply voltage for Devices
VRQ	residual voltage on driver in saturated operating status ON (measured in V)
$VRQH$	residual voltage on high-side driver in operating status ON (measured in V)
$VRQL$	residual voltage on low-side driver in saturated operating status ON (measured in V)
VTH	threshold voltage of receiver with reference to $V0$ (measured in V)
$VTHH$	threshold voltage of receiver for safe detection of a high signal (measured in V)
$VTHL$	threshold voltage of receiver for safe detection of a low signal (measured in V)
WURQ	wake-up request pulse

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.3 Conventions

3.3.1 General

The service model, service primitives, and the diagrams shown in this document are entirely abstract descriptions. The implementation of the services may reflect individual issues and may be different.

3.3.2 Service parameters

Service primitives are used to represent service provider/consumer interactions (ISO/IEC 10731 [18]). They convey parameters which indicate the information available in the provider/consumer interaction. In any particular interface, not each and every parameter needs to be explicitly stated.

The service specification in this document uses a tabular format to describe the component parameters of the 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:

1) parameter name,
2) request primitive (.req),
3) indication primitive (.ind),
4) response primitive (.rsp), and
5) confirmation primitive (.cnf).

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 can or cannot 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.

– Parameter is never present.

S Parameter is a selected item.

Some entries are further qualified by items in brackets. These can 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 related to the parameter and its use.

3.3.3 Service procedures

The procedures are defined in terms of:

– the interactions between application entities through the exchange of protocol data units, and

– the interactions between a communication layer service provider and a communication layer service consumer in the same system through the invocation of service primitives.

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

3.3.4 Service attributes

The nature of the different (Master and Device) services is characterized by attributes. All services are defined from the view of the affected layer towards the layer above.

I Initiator of a service (towards the layer above)

R Receiver (responder) of a service (from the layer above)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.3.5 Figures

For figures that show the structure and services of protocol layers, the following conventions are used:

– an arrow with just a service name represents both a request and the corresponding confirmation, with the request being in the direction of the arrow,

– a request without confirmation, as well as all indications and responses are labelled as such (i.e. service.req, service.ind, service.rsp).

Figure 1 shows the example of a confirmed service.

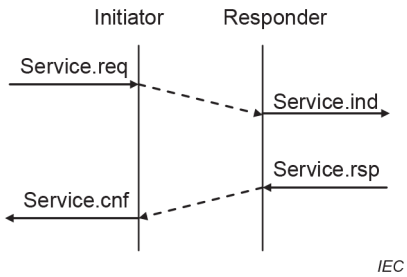


Figure 1 – Example of a confirmed service

3.3.6 Tables

Some tables in this document show highlighted areas (gray) to increase the information content (structure) and to indicate reserved or special content.

3.3.7 Transmission octet order

Figure 2 shows how word based data types are transferred from memory to transmission medium and vice versa (i.e. most significant octet transmitted first, see 7.3.3.2 and 7.3.6.1).

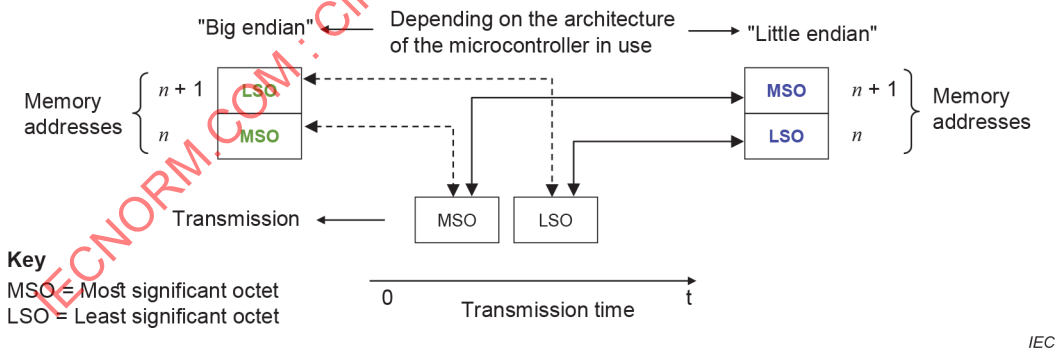


Figure 2 – Memory storage and transmission order for WORD based data types

3.3.8 Behavioral descriptions

For the behavioral descriptions, the notations of UML 2 (ISO/IEC 19505 series [19]) are used (e.g. state, sequence, activity, timing diagrams, guard conditions).

State diagrams are the primary source for implementations whereas sequence charts illustrate certain use cases. Characteristics of state diagrams are

– triggers or events coming from external requests ("calls") or internal changes such as timeouts,

– [guard(s)] as Boolean expressions for exits of states,

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

– numbered transitions describing actions in addition to the triggers within separate state-transition tables.

The layout of these tables follows IEC TR 62390 [14].

In this document, the concept of "nested states" with superstates and substates is used as shown in the example of Figure 3.

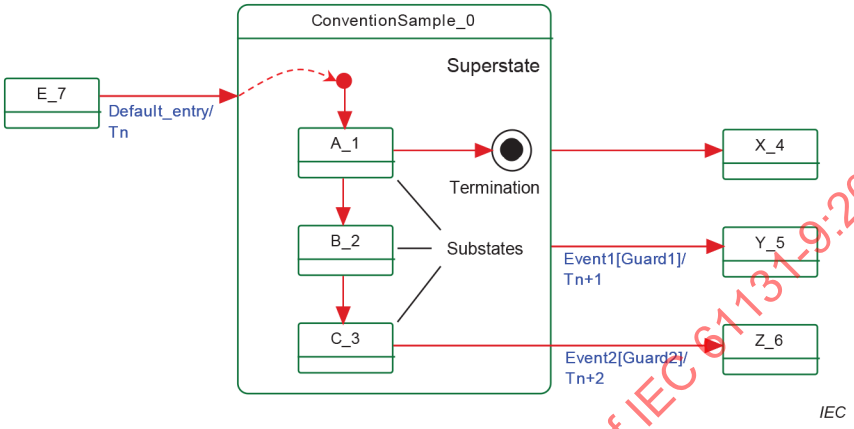


Figure 3 – Example of a nested state

UML 2 allows hierarchies of states with superstates and substates. The highest superstate represents the entire state machine.

This concept allows for simplified modelling since the content of superstates can be moved to a separate drawing. An eyeglasses icon usually represents this content.

Compared to "flat" state machines, a particular set of rules shall be observed for "nested states":

a) A transition to the edge of a superstate (e.g. Default_entry) implies transition to the initial substate (e.g. A_1).

b) Transition to a termination state inside a superstate implies a transition without event and guard to a state outside (e.g.X_4). The superstate will become inactive.

c) A transition from any of the substates (e.g. A_1, B_2, or C_3) to a state outside (Y_5) can take place whenever Event1 occurs and Guard1 is true. This is helpful in case of common errors within the substates. The superstate will become inactive.

d) A transition from a particular substate (e.g. C_3) to a state outside (Z_6) can take place whenever Event2 occurs and Guard2 is true. The superstate will become inactive.

Owing to UML design tool restrictions the following exceptions apply.

For state diagrams, a service parameter (in capital letters) is attached to the service name via an underscore character, such as for example in DL_SetMode_INACTIVE.

For sequence diagrams, the service primitive is attached via an underscore character instead of a dot, and the service parameter is added in parenthesis, such as for example in DL_Event_ind (OPERATE).

Timing constraints are labelled "tm(time in ms)".

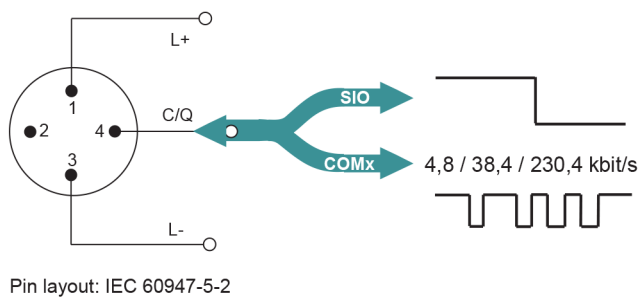
Asynchronously received service calls are not modelled in detail within state diagrams.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

4 Overview of SDCI (IO-Link™)

4.1 Purpose of technology

Figure 4 shows the basic concept of SDCI.



Pin	Signal	Definition	Standard
1	L+	24 V	IEC 61131-2
2	I/Q	Not connected, DI, or DO	IEC 61131-2
3	L-	0 V	IEC 61131-2
4	Q	"Switching signal" (SIO)	IEC 61131-2
	C	"Coded switching" (COM1, COM2, COM3)	IEC 61131-9

IEC

Figure 4 – SDCI compatibility with IEC 61131-2

The single-drop digital communication interface technology for small sensors and actuators SDCI (commonly known as IO-Link™) defines a migration path from the existing digital input and digital output interfaces for switching 24 V Devices as defined in IEC 61131-2 towards a point-to-point communication link. Thus, for example, digital I/O modules in existing fieldbus peripherals can be replaced by SDCI Master modules providing both classic DI/DO interfaces and SDCI. Analog transmission technology can be replaced by SDCI combining its robustness, parameterization, and diagnostic features with the saving of digital/analog and analog/digital conversion efforts.

4.2 Positioning within the automation hierarchy

Figure 5 shows the domain of the SDCI technology within the automation hierarchy.

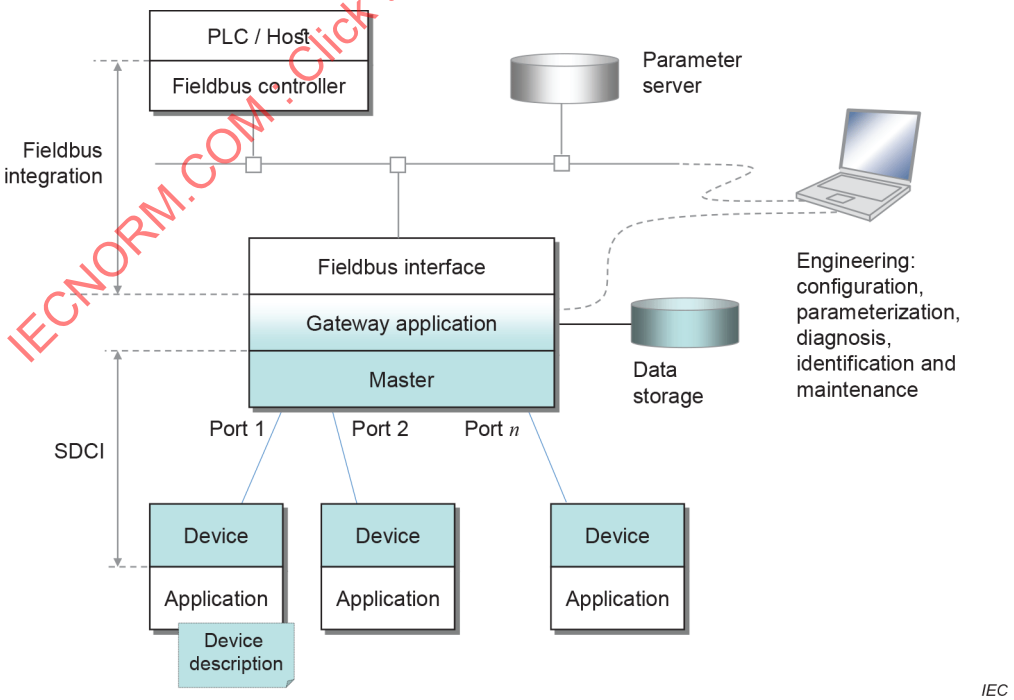


Figure 5 – Domain of the SDCI technology within the automation hierarchy

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The SDCI technology defines a generic interface for connecting sensors and actuators to a Master unit, which can be combined with gateway capabilities to become a fieldbus remote I/O node.

Starting point for the design of SDCI is the classic 24 V digital input (DI) defined in IEC 61131-2 and output interface (DO) specified in Table 6. Thus, SDCI offers connectivity of classic 24 V sensors ("switching signals") as a default operational mode. Additional connectivity is provided for actuators when a Port has been configured into "single-drop communication mode".

Many sensors and actuators nowadays are already equipped with microcontrollers offering a UART interface that can be extended by addition of a few hardware components and protocol software to support SDCI communication. This second operational mode uses "coded switching" of the 24 V I/O signalling line. Once activated, the SDCI mode supports parameterization, cyclic data exchange, diagnosis reporting, identification and maintenance information, and external parameter storage for Device backup and fast reload of replacement devices. Sensors and actuators with SDCI capability are referred to as "Devices" in this document. To improve start-up performance these Devices usually provide non-volatile storage for parameters.

NOTE Configuration and parameterization of Devices is supported through an XML-based device description (see [6]), which is not part of this document.

4.3 Wiring, connectors and power

The default connection (Port Class A) comprises four pins (see Figure 4). The default wiring for Port Class A complies with IEC 60947-5-2 and uses only three wires for 24 V, 0 V, and a signal line. The fourth wire may be used as an additional signal line complying with IEC 61131-2.

Five pins connections (Port Class B) are specified for Devices requiring additional power from an independant 24 V power supply.

NOTE A Port Class A Device using the fourth wire is not compatible with a Port Class B Master.

Maximum length of cables is 20 m, shielding is not required.

4.4 Communication features of SDCI

The generic Device model is shown in Figure 6 and explained in this subclause.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

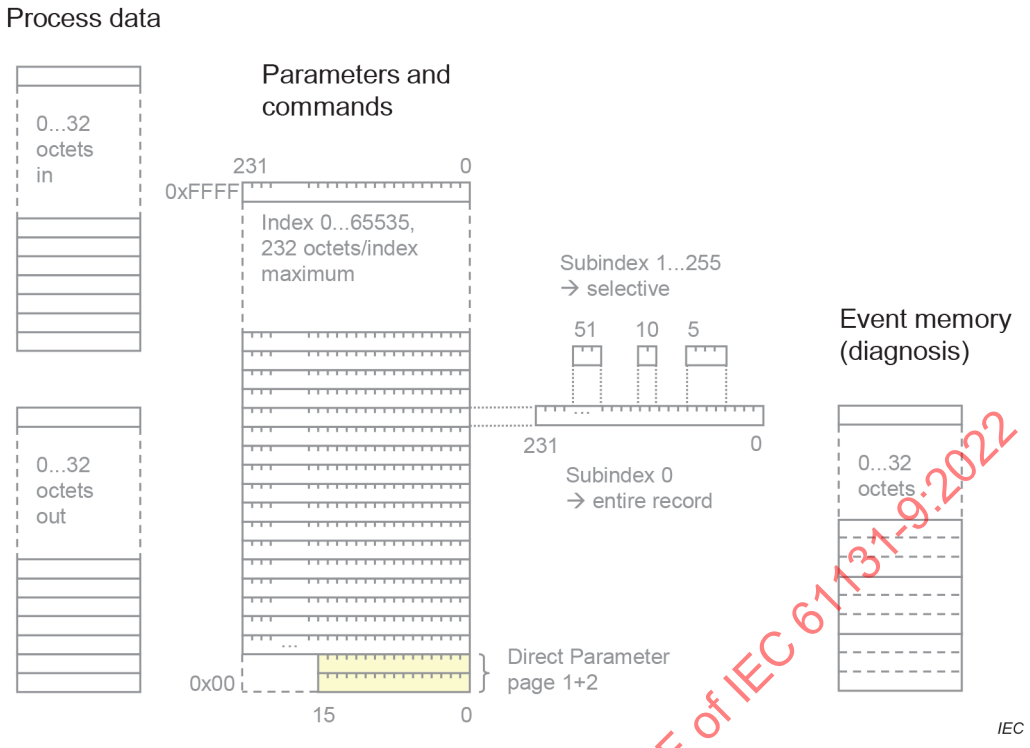


Figure 6 – Generic Device model for SDCI (Master's view)

A Device can receive Process Data (out) to control a discrete or continuous automation process or send Process Data (in) representing its current state or measurement values. The Device usually provides parameters enabling the user to configure its functions to satisfy particular needs. To support this case a large parameter space is defined with access via an Index (0 to 65 535; with a predefined organization) and a Subindex (0 to 255).

The first two index entries 0 and 1 are reserved for the Direct Parameter pages 1 and 2 with a maximum of 16 octets each. Parameter page 1 is mainly dedicated to Master commands such as Device startup and fallback, retrieval of Device specific operational and identification information. Parameter page 2 allows for a maximum of 16 octets of Device specific parameters.

The other Indices (2 to 65 535) each allow access to one record having a maximum size of 232 octets. Subindex 0 specifies transmission of the complete record addressed by the Index; other subindices specify transfer of selected data items within the record.

Within a record, individual data items can start on any bit offset, and their length can range from 1 bit to 232 octets, but the total number of data items in the record cannot exceed 255. The organization of data items within a record is specified in the IO Device Description (IODD).

All changes of Device condition that require reporting or intervention are stored within an Event memory before transmission. An Event flag is then set in the cyclic data exchange to indicate the existence of an Event.

Communication between a Master and a Device is point-to-point and is based on the principle of a Master first sending a request message and then a Device sending a response message (see Figure 38). Both messages together are called an M-sequence. Several M-sequence types are defined to support user requirements for data transmission (see Figure 39).

Data of various categories are transmitted through separate communication channels within the data link layer, as shown in Figure 7.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

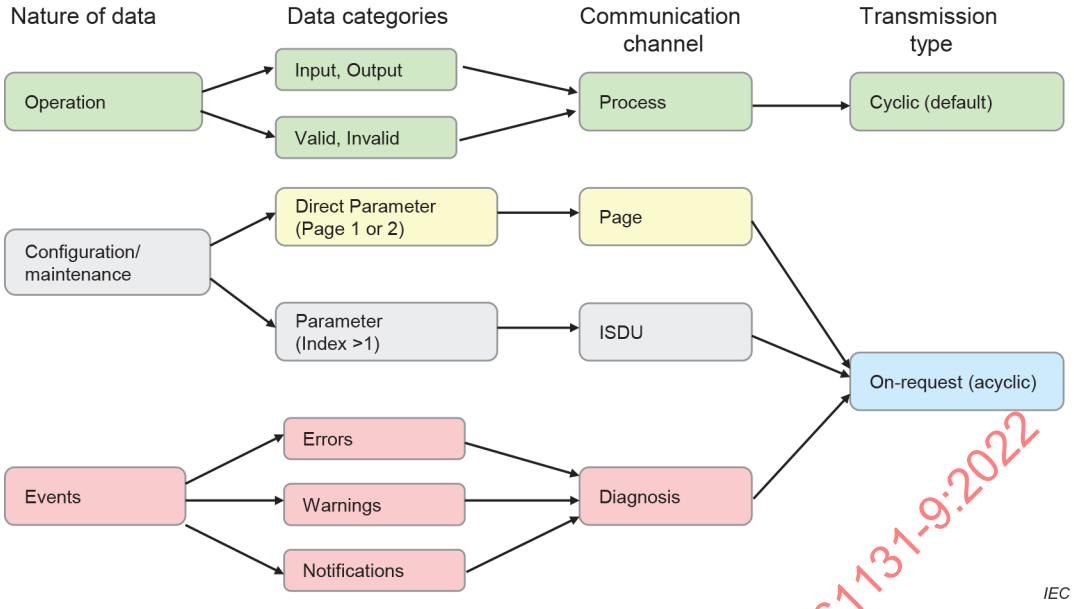


Figure 7 – Relationship between nature of data and transmission types

– Operational data such as Device inputs and outputs is transmitted through a process channel using cyclic transfer. Operational data can also be associated with qualifiers such as valid/invalid.

– Configuration and maintenance parameters are transmitted using acyclic transfers. A page channel is provided for direct access to parameter pages 1 and 2, and an ISDU channel is used for accessing additional parameters and commands.

– Device Events are transmitted using acyclic transfers through a diagnosis channel. Device Events are reported using three severity levels, error, warning, and notification.

The first octet of a Master message controls the data transfer direction (read/write) and the type of communication channel.

Figure 8 shows each Port of a Master has its own data link layer which interfaces to a common master application layer. Within the application layer, the services of the data link layer are translated into actions on Process Data objects (input/output), On-request Data objects (read/write), and events. Master applications include a Configuration Manager (CM), Data Storage (DS) mechanism, Diagnosis Unit (DU), On-request Data Exchange (ODE), and a Process Data Exchange (PDE).

System Management checks identification of the connected Devices and adjusts Ports and Devices to match the chosen configuration and the properties of the connected Devices. It controls the state machines in the application (AL) and data link layers (DL), for example at start-up.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

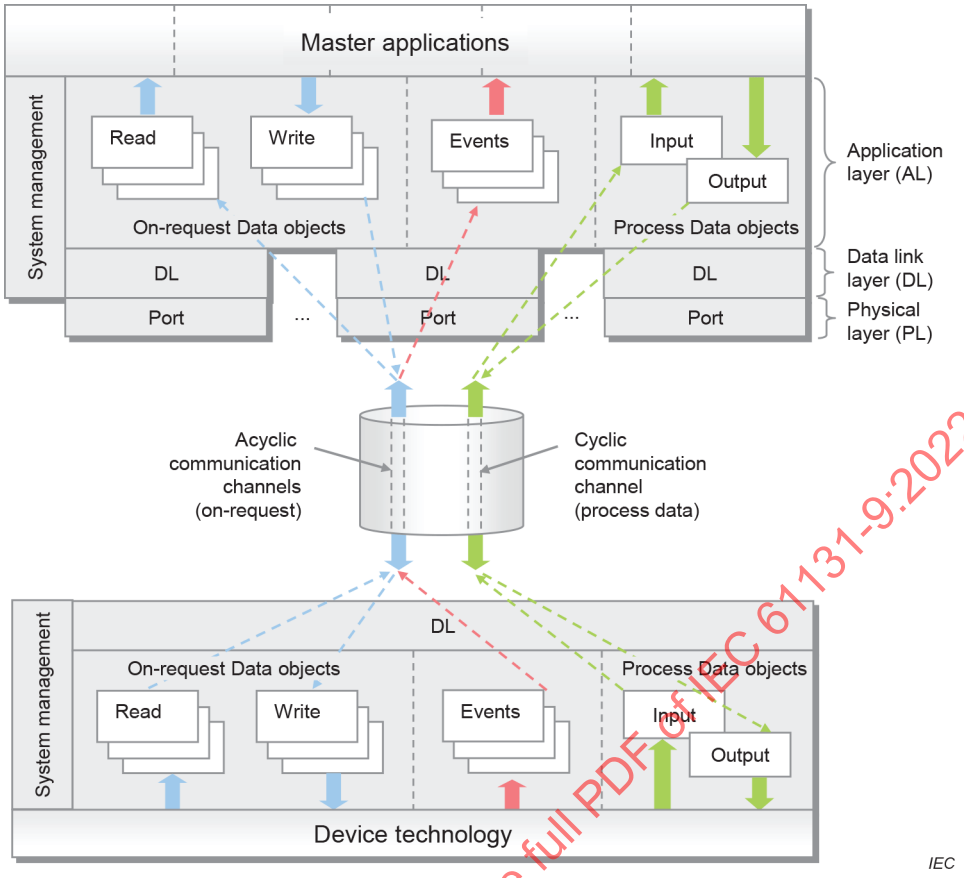


Figure 8 – Object transfer at the application layer level (AL)

4.5 Role of a Master

A Master accommodates 1 to n Ports and their associated data link layers. During start-up it changes the Ports to the user-selected Port modes, which can be DEACTIVATED, IOL_MANUAL, IOL_AUTOSTART, DI_C/Q, or DO_C/Q. If communication is requested, the Master uses a special wake-up current pulse to initiate communication with the Device. The Master then auto-adjusts the transmission rate to COM1, COM2, or COM3 (see Table 9) and checks the "personality" of the connected Device, i.e. its VendorID, DeviceID, and communication properties.

If there is a mismatch between the Device parameters and the stored parameter set within the Master, the parameters in the Device are overwritten (see 11.4) or the stored parameters within the Master are updated depending on the configuration.

The Master is responsible for the assembling and disassembling of all data from or to the Devices (see Clause 11).

The Master provides a Data Storage area of at least 2 048 octets per Device for backup of Device data (see 11.4). The Master may combine this Device data together with all other relevant data for its own operation and make this data available for higher level applications for Master backup purpose or recipe control (see 13.4.2).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

4.6 SDCI configuration

Engineering support for a Master is usually provided by a Port and Device Configuration Tool (PDCT). The PDCT configures both Port properties and Device properties (see parameters shown in Figure 6). It combines both an interpreter of the I/O Device Description (IODD) and a configurator (see Clause 13). The IODD provides all the necessary properties to establish communication and the necessary parameters and their boundaries to establish the desired function of a sensor or actuator. The PDCT also supports the compilation of the Process Data for propagation on the fieldbus and vice versa.

4.7 Mapping to fieldbuses and/or other upper-level systems

Specifications for integration of Masters into upper-level systems such as a fieldbus system, i.e. the definition of gateway functions for exchanging data with upper-level entities, is out of the scope of this document. However, it is mandatory for all functions of this document to be made available to the users by a particular integration according to the capability level of the upper-level system technology except for those functions that are declared explicitly as optional.

EXAMPLE These functions include mapping of the Process Data exchange, realization of program-controlled parameterization or a remote parameter server, or the propagation of diagnosis information.

The integration of a PDCT into engineering tools of a particular fieldbus or other upper-level system is out of the scope of this document.

4.8 Standard structure

Figure 9 shows the logical structure of the Master and Device. Clause 5 specifies the Physical Layer (PL) of SDCI, Clause 6 specifies details of the SIO mode. Clause 7 specifies Data Link Layer (DL) services, protocol, wake-up, M-sequences, and the DL layer handlers. Clause 8 specifies the services and the protocol of the Application Layer (AL) and Clause 9 the System Management (SM) responsibilities.

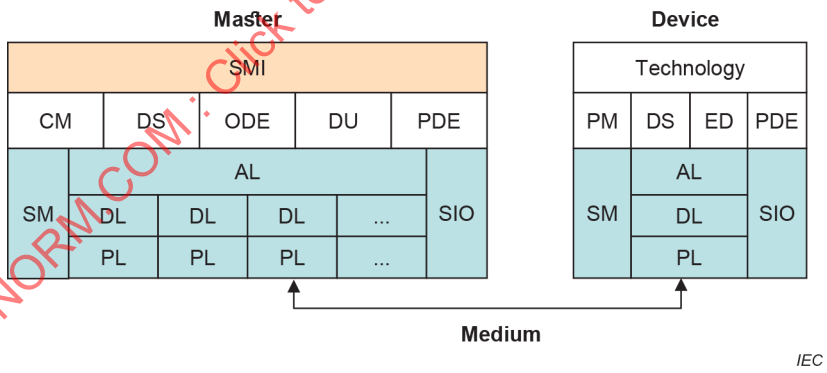


Figure 9 – Logical structure of Master and Device

Clause 10 specifies Device applications and features. These include Process Data Exchange (PDE), Parameter Management (PM), Data Storage (DS), and Event Dispatcher (ED). Technology specific Device applications are not part of this document. They may be specified in profiles for particular Device families.

Clause 11 specifies Master applications and features. These include Process Data Exchange (PDE), On-request Data Exchange (ODE), Configuration Management (CM), Data Storage (DS) and Diagnosis Unit (DU). A Standardized Master Interface (SMI) ensures uniform behavior via specified services and allows for usage of one PDCT (Master tool) for different Master brands.

Clause 12 provides a holistic best practice view on Data Storage behavior of both Master and Device.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Clause 13 outlines integration aspects of SDCI into various automation and IT realms.

Several normative and informative annexes are included. Annex A defines the available M-sequence types. Annex B describes the parameters of the Direct Parameter page and the fixed Device parameters. Annex C lists the error types in case of acyclic transmissions and Annex D the EventCodes (diagnosis information of Devices). Annex E specifies the coding of argument blocks for the SMI services. Annex F specifies the available basic and composite data types. Annex G defines the structure of Data Storage objects. Annex H deals with conformity and electromagnetic compatibility test requirements and Annex I provides graphs of residual error probabilities, demonstrating the level of SDCI's data integrity. The informative Annex J provides an example of the sequence of acyclic data transmissions. The informative Annex K explains two recommended methods for detecting parameter changes in the context of Data Storage.

5 Physical Layer (PL)

5.1 General

5.1.1 Basics

The 3-wire connection system of SDCI is based on the specifications in IEC 60947-5-2. The three lines are used as follows: (L+) for the 24 V power supply, (L–) for the ground line, and (C/Q) for the switching signal (Q) or SDCI communication (C), as shown in Figure 10.

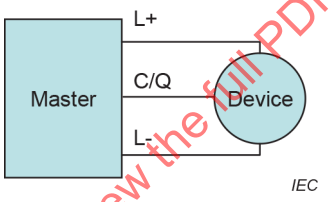


Figure 10 – Three wire connection system

NOTE 1 Binary sensors compliant with IEC 60947-5-2 are compatible with the SDCI 3-wire connection system (including from a power consumption point of view).

Support of the SDCI 3-wire connection system is mandatory for the Master. Ports with this characteristic are called Port Class A.

Port Class A uses a four-pin connector. The fourth wire may be used as an additional signal line complying with IEC 61131-2. Its support is optional in both Masters and Devices.

Five wire connections (Port Class B) are specified for Devices requiring additional power from an independant 24 V power supply (see 5.5.1).

NOTE 2 A Port Class A Device using the fourth wire is not compatible with a Port Class B Master.

5.1.2 Topology

The SDCI system topology uses point-to-point links between a Master and its Devices as shown in Figure 11. The Master can have multiple Ports for the connection of Devices. Only one Device shall be connected to each Port.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

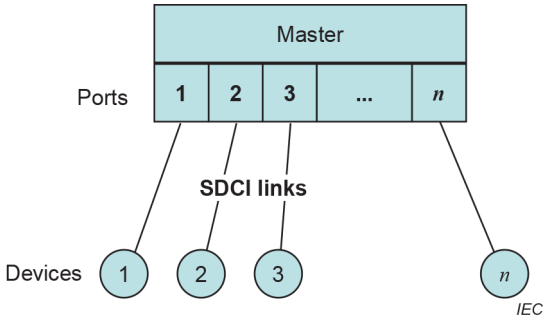


Figure 11 – Topology of SDCI

5.2 Physical layer services

5.2.1 Overview

Figure 12 shows an overview of the Master's physical layer and its service primitives.

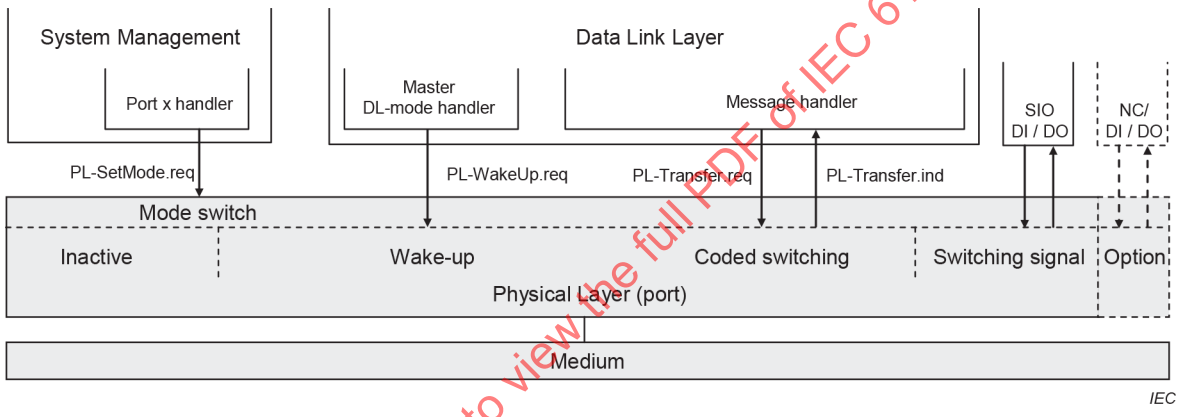


Figure 12 – Physical layer (Master)

The physical layer specifies the operation of the C/Q line in Figure 4 and the associated line driver (transmitter) and receiver of a particular Port. The Master operates this line in three main modes (see Figure 12): inactive, "Switching signal" (DI/DO), or "Coded switching" (COMx). The service PL-SetMode.req is responsible for switching into one of these modes.

If the Port is in inactive mode, the C/Q line shall be high impedance (floating). In SIO mode, the Port can be used as a standard input or output interface according to the definitions of IEC 61131-2 or in Table 6 respectively. The communication layers of SDCI are bypassed as shown in Figure 12; the signals are directly processed within the Master application. In SDCI mode, the service PL_WakeUp.req creates a special signal pattern (current pulse) that can be detected by an SDCI enabled Device connected to this Port (see 5.3.3.3).

Figure 13 shows an overview of the Device's physical layer and its service primitives.

The physical layer of a Device according to Figure 13 follows the same principle, except that there is no inactive state. By default, at power on or cable reconnection, the Device shall operate in the SIO mode, as a digital input (from a Master's point of view). The Device shall always be able to detect a wake-up current pulse (wake-up request) except during a permanent inactive state. The service PL_WakeUp.ind reports successful detection of the wake-up request (usually a microcontroller interrupt), which is required for the Device to switch to the SDCI mode.

A special MasterCommand (fallback) sent via SDCI causes the Device to switch back to SIO mode.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

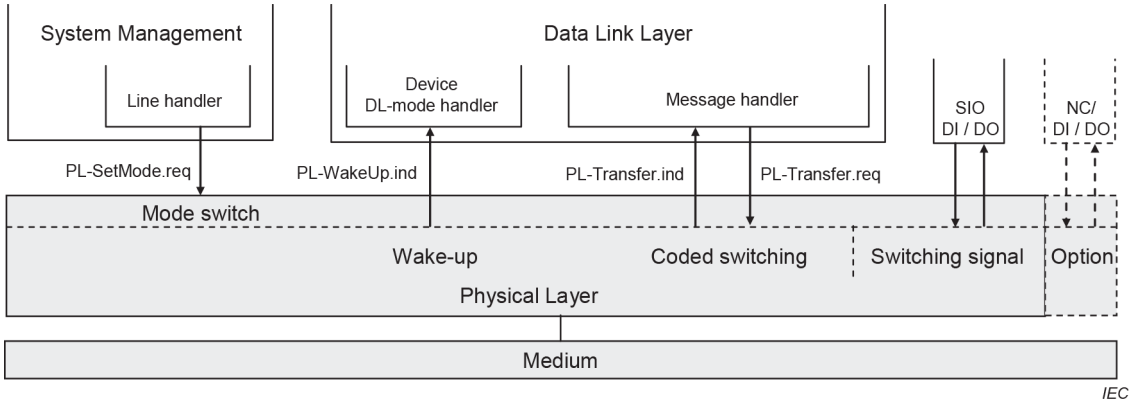


Figure 13 – Physical layer (Device)

Subsequently, those services are specified that are provided by the PL to System Management and to the Data Link Layer (see Figure 85 and Figure 96 for a complete overview of all the services). Table 1 lists the assignments of Master and Device to their roles as initiator or receiver for the individual PL services.

Table 1 – Service assignments of Master and Device

Service name	Master	Device
PL-SetMode	R	R
PL-WakeUp	R	I
PL-Transfer	I / R	R / I
Key (see 3.3.4) I Initiator of service R Receiver (Responder) of service		

5.2.2 PL services

5.2.2.1 PL_SetMode

The PL-SetMode service is used to set up the electrical characteristics and configurations of the Physical Layer. The parameters of the service primitives are listed in Table 2.

Table 2 – PL_SetMode

Parameter name	.req
Argument TargetMode	M M

Argument

The service-specific parameters of the service request are transmitted in the argument.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TargetMode

This parameter indicates the requested operation mode

Permitted values:
INACTIVE (C/Q line in high impedance),
DI (C/Q line in digital input mode),
DO (C/Q line in digital output mode),
COM1 (C/Q line in COM1 mode),
COM2 (C/Q line in COM2 mode),
COM3 (C/Q line in COM3 mode).

5.2.2.2 PL_WakeUp

The PL-WakeUp service initiates or indicates a specific sequence which prepares the Physical Layer to send and receive communication requests (see 5.3.3.3). This unconfirmed service has no parameters. Its success can only be verified by a Master by attempting to communicate with the Device. The service primitives are listed in Table 3.

Table 3 – PL_WakeUp

Parameter name	.req	.ind
<none>		

5.2.2.3 PL_Transfer

The PL-Transfer service is used to exchange the SDCI data between Data Link Layer and Physical Layer. The parameters of the service primitives are listed in Table 4.

Table 4 – PL_Transfer

Parameter name	.req	ind.
Argument		
Data	M	M
Result (+)		S
Result (−)		S
Status		M

Argument

The service-specific parameters of the service request are transmitted in the argument.

Data

This parameter contains the data value which is transferred over the SDCI interface.

Permitted values: 0…255.

Result (+):

This selection parameter indicates that the service request has been executed successfully.

Result (−):

This selection parameter indicates that the service request failed.

Status

This parameter contains supplementary information on the transfer status.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Permitted values:
PARITY_ERROR (UART detected a parity error),
FRAMING_ERROR (invalid UART stop bit detected),
OVERRUN (octet collision within the UART).

5.3 Transmitter/Receiver

5.3.1 Description method

The physical layer is specified by means of electrical and timing requirements. Electrical requirements specify signal levels and currents separately for Master and Device in the form of reference schematics. Timing requirements specify the signal transmission process (specifically the receiver) and a special signal detection function.

5.3.2 Electrical requirements

5.3.2.1 General

The line driver is specified by a reference schematic diagram corresponding to Figure 14. On the Master side, a transmitter comprises a combination of two line drivers and one current sink. On the Device side, in its simplest form, the transmitter takes the form of a p-switching driver. As an option there can be an additional n-switching or non-switching driver (this also allows the option of push-pull output operation).

In operating status ON the descriptive variables are the residual voltage *VRQ*, the standard driver current *IQ*, and the peak current *IQPK*. The source is controlled by the On/Off signal. An overload current event is indicated at the "Overload" output (OVD). This feature can be used for the current pulse detection (wake-up).

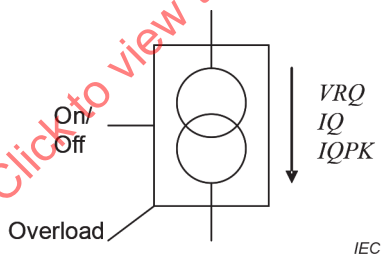


Figure 14 – Line driver reference schematics

The receiver is specified by a reference schematic diagram according to Figure 15. It performs the function of a comparator and is specified by its switching thresholds *VTH* and a hysteresis *VHYS* between the switching thresholds. The output indicates the logic level (High or Low) at the receiver input.

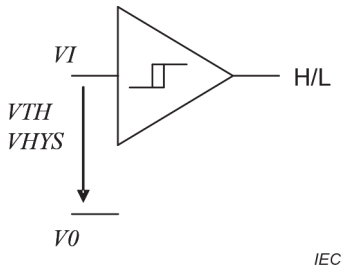
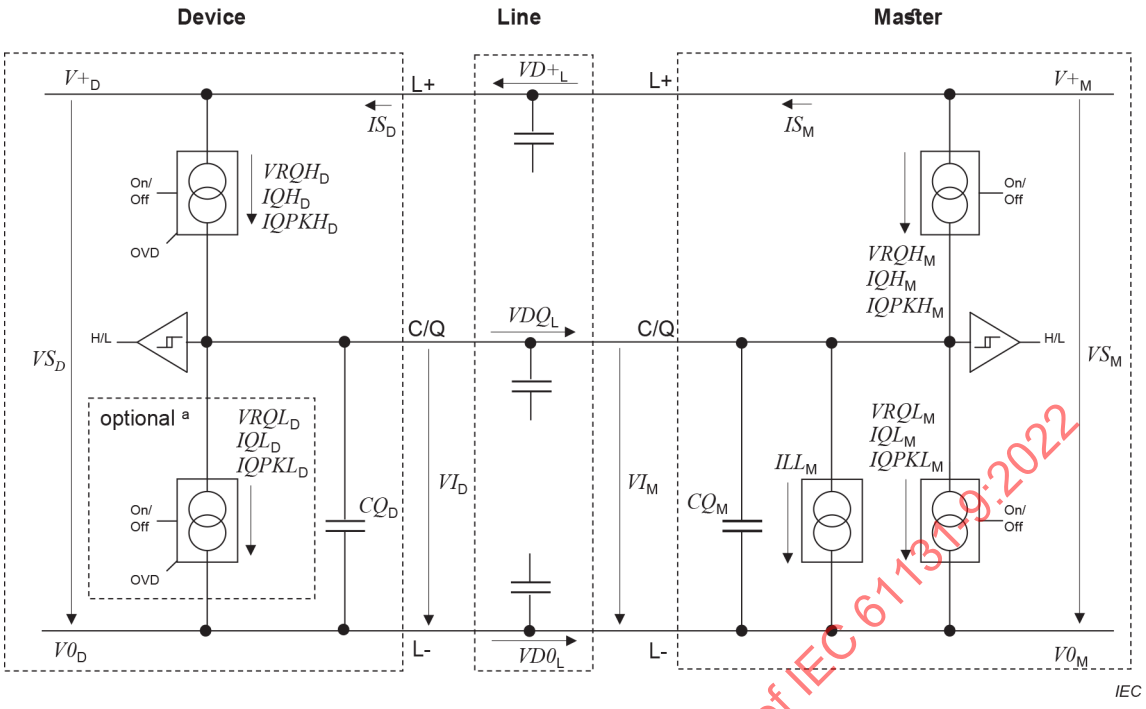


Figure 15 – Receiver reference schematics

Figure 16 shows the reference schematics for the interconnection of Master and Device for the SDCI 3-wire connection system.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022



[a] Optional: low-side driver (push-pull only)

Figure 16 – Reference schematics for SDCI 3-wire connection system

The subsequent illustrations and parameter tables refer to the voltage level definitions in Figure 17. The parameter indices refer to the Master (M), Device (D) or line (L). The voltage drops on the line $VD+_L$, VDQ_L and $VD0_L$ are implicitely specified in 5.5 through cable parameters.

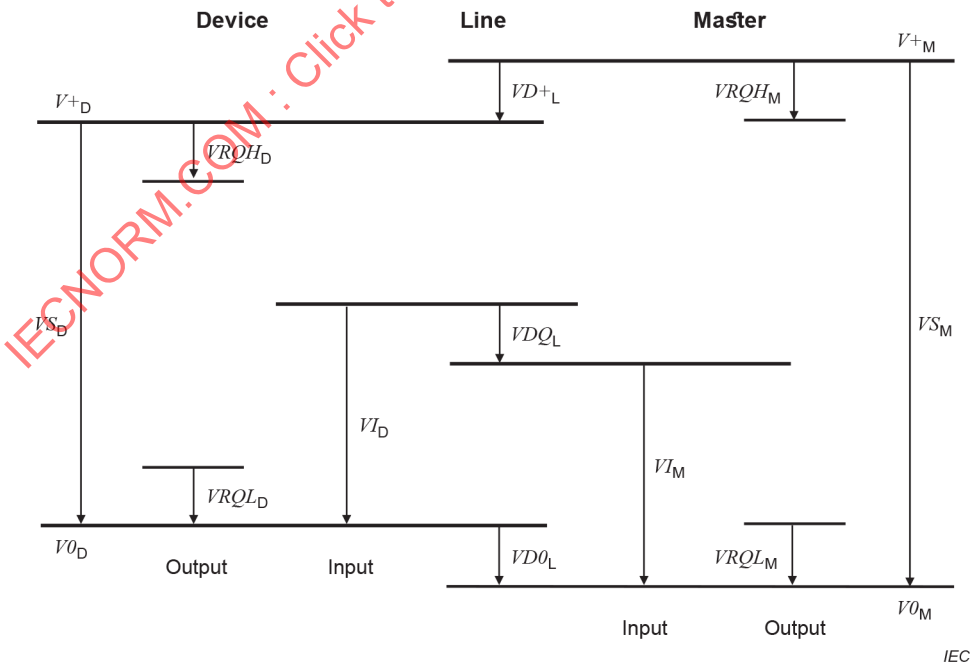


Figure 17 – Voltage level definitions

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

5.3.2.2 Receiver

The voltage range and switching threshold definitions are the same for Master and Device. The definitions in Table 5 apply (see also 5.4.1).

Table 5 – Electrical characteristics of a receiver

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
$VTHH_{D,M}$	Input threshold 'H'	10,5	n/a	13	V	See NOTE 1
$VTHL_{D,M}$	Input threshold 'L'	8	n/a	11,5	V	See NOTE 1
$VHYS_{D,M}$	Hysteresis between input thresholds 'H' and 'L'	0	n/a	n/a	V	Shall not be negative See NOTE 2
VIL_D	Permissible voltage range 'L'	$V0_D$ 1−,0	n/a	n/a	V	With reference to relevant negative supply voltage See NOTE 3
VIL_M	Permissible voltage range 'L'	$V0_M$	n/a	n/a	V	
VIH_D	Permissible voltage range 'H'	n/a	n/a	$V+_D$ + 1,0	V	With reference to relevant positive supply voltage. See NOTE 3
VIH_M	Permissible voltage range 'H'	n/a	n/a	$V+_M$	V	
NOTE 1 Thresholds are compatible with the definitions of type 1 digital inputs in IEC 61131-2. NOTE 2 Hysteresis voltage $VHYS = VTHH - VTHL$. NOTE 3 Owing to 5.4.1 the Master receiver signals VI_M are always within permitted supply ranges.						

Figure 18 demonstrates the switching thresholds for the detection of Low and High signals.

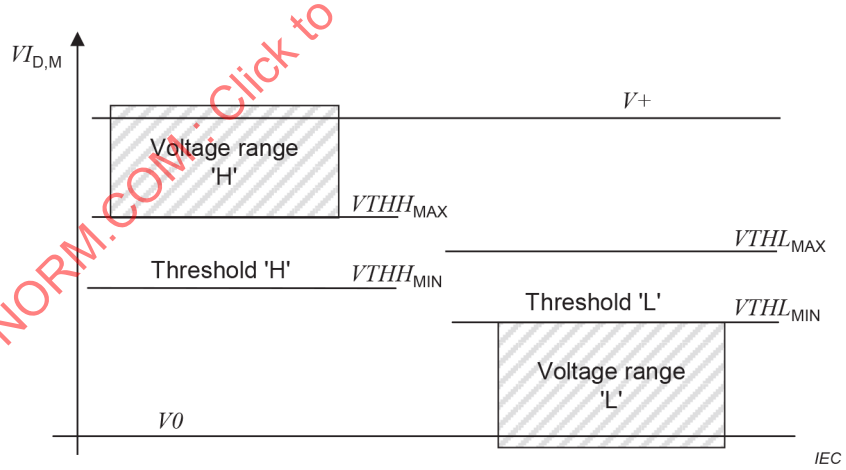


Figure 18 – Switching thresholds

5.3.2.3 Master Port

The definitions in Table 6 are valid for the electrical characteristics of a Master Port.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 6 – Electrical characteristics of a Master Port

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
VS_M	Supply voltage for Devices	20	24	30	V	See Figure 17
IS_M	Supply current for Devices	200	n/a	n/a	mA	See 5.4.1
$ISIR_M$	Current pulse capability for Devices	400	n/a	n/a	mA	See Figure 19
ILL_M	Load or discharge current for					
	0 V < VI_M < 5 V	0	n/a	15	mA	
	5 V < VI_M < 15 V	5/2	n/a	15	mA	See NOTE 1
	15 V < VI_M < 30 V	5	n/a	15	mA	
$VRQH_M$	Residual voltage 'H'	n/a	n/a	3	V	Voltage drop relating to VI_M at maximum driver current IQH_M
$VRQL_M$	Residual voltage 'L'	n/a	n/a	3	V	Voltage drop relating to $V0_M$ at maximum driver current IQL_M
IQH_M	DC driver current 'H'	100	n/a	n/a	mA	
$IQPKH_M$	Output peak current 'H'	500	n/a	n/a	mA	Absolute value See NOTE 2
IQL_M	DC driver current 'L'	100	n/a	n/a	mA	
$IQPKL_M$	Output peak current 'L'	500	n/a	n/a	mA	Absolute value See NOTE 2
CQ_M	Input capacitance	n/a	n/a	1,0	nF	f = 0 MHz to 4 MHz

NOTE 1 A minimum current of 2 mA for DI mode is compatible with the definition of type 1 digital inputs in IEC 61131-2. In communication mode, for the range 5 V < VI_M < 15 V, the minimum current is 5 mA instead of 2 mA in order to achieve short enough slew rates for pure p-switching Devices.

NOTE 2 Wake-up request current (5.3.3.3).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The Master shall provide a charge of 400 mA × 50 ms = 20 mAs within the first 50 ms after power-on without any overload-shutdown. After 50 ms, the specific current limitation of the Master or system applies.

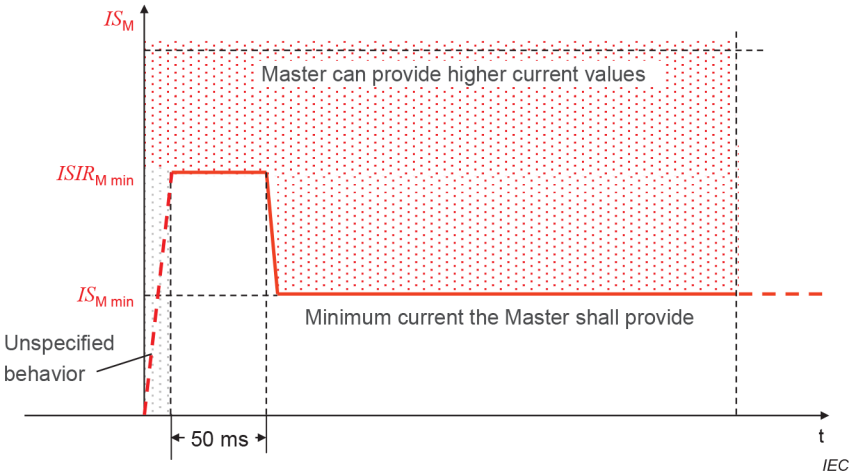


Figure 19 – Inrush current and charge (example)

5.3.2.4 Device

The definitions in Table 7 are valid for the electrical characteristics of a Device.

Table 7 – Electrical characteristics of a Device

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
VS_D	Supply voltage	18	24	30	V	See Figure 17
QIS_D	Power-up charge consumption	n/a	n/a	70	mAs	See Equation (1) and Table 8
ΔVS_D	Ripple	n/a	n/a	1,3	V	Peak-to-peak absolute value limits shall not be exceeded. f_{ripple} = DC to 100 kHz
$VRQH_D$	Residual voltage 'H'	n/a	n/a	3	V	Voltage drop compared with V_{L+D} (IEC 60947-5-2)
$VRQL_D$	Residual voltage 'L'	n/a	n/a	3	V	Voltage drop compared with $V0_D$
IQH_D	DC driver current P-switching output ("On" state)	50	n/a	minimum ($IQPKL_M$)	mA	Minimum value due to fallback to digital input in accordance with IEC 61131-2, type 2
IQL_D	DC driver current N-switching output ("On" state)	0	n/a	minimum ($IQPKH_M$)	mA	Only for push-pull output stages
IQQ_D	Quiescent current to $V0_D$ ("Off" state)	0	n/a	15	mA	Pull-down or residual current with deactivated output driver stages
CQ_D	Input capacitance	0	n/a	1,0	nF	Effective capacitance between C/Q and L+ or L− of Device in receive state

The Device shall be able to reach a stable operational state (ready for wake-up) consuming the maximum charge according to Equation (1).

$$QIS_D = ISIR_M \times 50\text{ms} + (T_{RDL} - 50\text{ms}) \times IS_M \quad (1)$$

Figure 20 shows how the power-on behavior of a Device is defined by the ramp-up time of the Power 1 supply and by the Device internal time to get ready for the wake-up operation.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

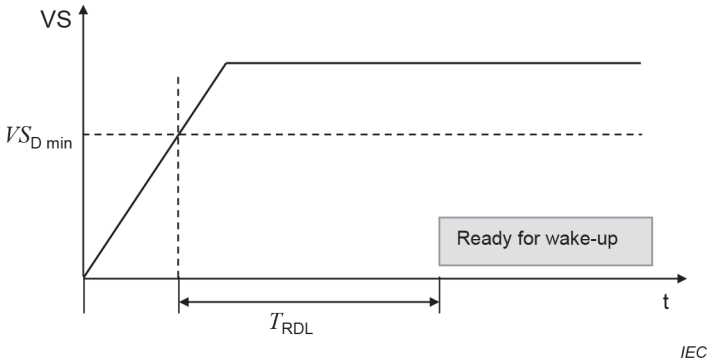


Figure 20 – Power-on timing for Power 1

Upon power-on it is mandatory for a Device to reach the wake-up ready state within the time limits specified in Table 8.

Table 8 – Power-on timing

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
T_{RDL}	Wake-up readiness following power-on	n/a	n/a	300	ms	Device ramp-up time until it is ready for wake-up signal detection (See NOTE)
NOTE Equivalent to the time delay before availability in IEC 60947-5-2.						

The value of 1 nF for input capacitance CQ_D is applicable for a transmission rate of 230,4 kbit/s. It can be relaxed to a maximum of 10 nF in case of push-pull stage design when operating at lower transmission rates, provided that all dynamic parameter requirements in 5.3.3.2 are met.

5.3.3 Timing requirements

5.3.3.1 Transmission method

The "Non-Return to Zero" (NRZ) modulation is used for the bit-by-bit coding. A logic value "1" corresponds to a voltage difference of 0 V between the C/Q line and L− line. A logic value "0" corresponds to a voltage difference of +24 V between the C/Q line and L− line.

The open-circuit level on the C/Q line is 0 V with reference to L−. A start bit has logic value "0", i.e. +24 V with reference to L−.

A UART frame is used for the "data octet"-by-"data octet" coding. The format of the SDCI UART frame is a bit string structured as shown in Figure 21.

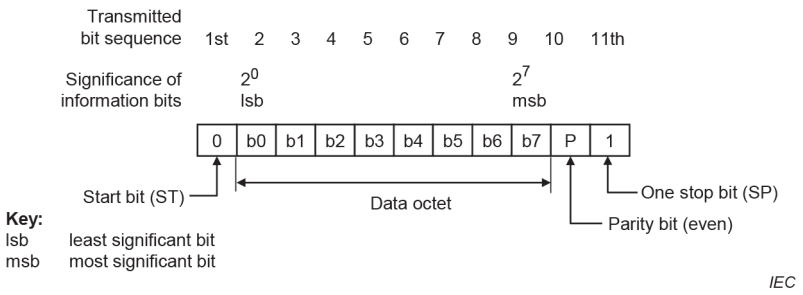


Figure 21 – Format of an SDCI UART frame

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

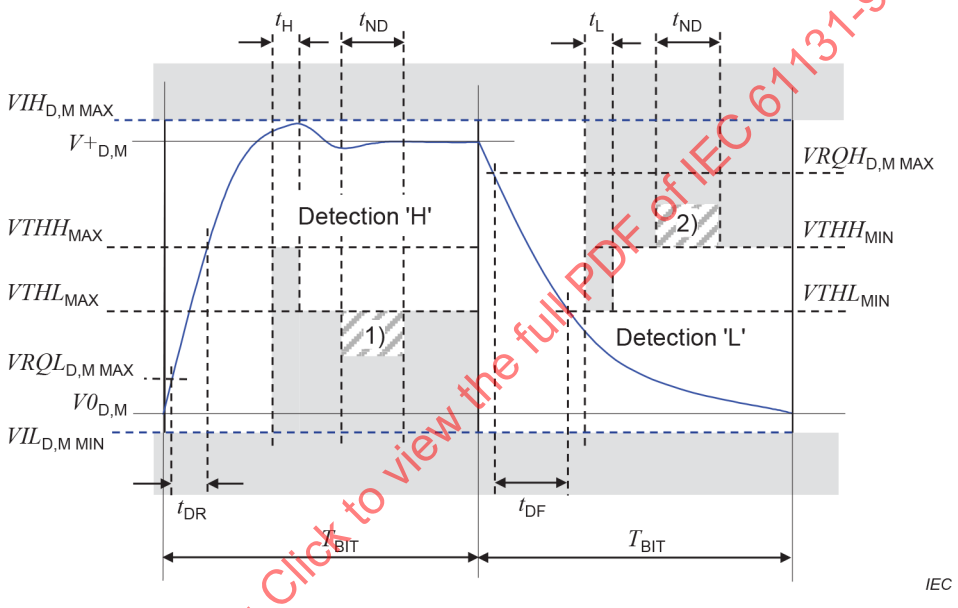
The definition of the UART frame format is based on ISO 1177 [20] and ISO/IEC 2022 [16].

5.3.3.2 Transmission characteristics

The timing characteristics of transmission are demonstrated in the form of an eye diagram with the permissible signal ranges (see Figure 22). These ranges are applicable for the receiver in both the Master and the Device.

Regardless of boundary conditions, the transmitter shall generate a voltage characteristic on the receiver's C/Q connection that is within the permissible range of the eye diagram.

The receiver shall detect bits as a valid signal shape within the permissible range of the eye diagram on the C/Q connection. Signal shapes in the "no detection" areas (below $VTHL_{MAX}$ or above $VTHH_{MIN}$ and within t_{ND}) shall not lead to invalid bits.



Key

1) = no detection 'L'

2) = no detection 'H'

Figure 22 – Eye diagram for the 'H' and 'L' detection

In order for a UART frame to be detected correctly, a signal characteristic as demonstrated in Figure 23 is required on the receiver side. The signal delay time between the C/Q signal and the UART input shall be considered. Time T_{BIT} always indicates the receiver's bit rate.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

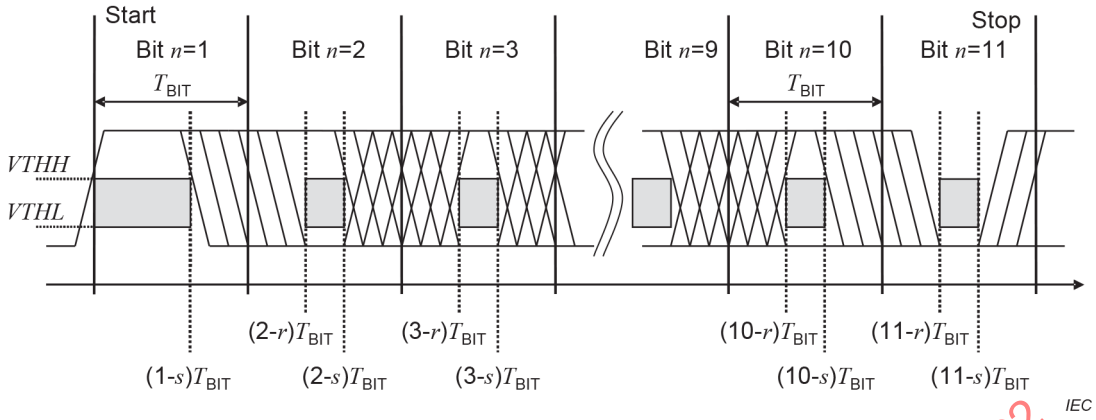


Figure 23 – Eye diagram for the correct detection of a UART frame

For every bit n in the bit sequence (n =1…11) of a UART frame, the time (n-r)T_{BIT} (see Table 9 for values of r) designates the time at the end of which a correct level shall be reached in the 'H' or 'L' ranges as demonstrated in the eye diagram in Figure 22. The time (n-s) T_{BIT} (see Table 9 for values of s) describes the time, which shall elapse before the level changes. Reference shall always be made to the eye diagram in Figure 22, where signal characteristics within a bit time are concerned.

This representation permits a variable weighting of the influence parameters "transmission rate accuracy", "bit-width distortion", and "slew rate" of the receiver.

Table 9 specifies the dynamic characteristics of the transmission.

Table 9 – Dynamic characteristics of the transmission

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
f_{DTR}	transmission rate	n/a	4,8 38,4 230,4	n/a	kbit/s	COM1 COM2 COM3
T_{BIT}	Bit time at 4,8 kbit/s at 38,4 kbit/s at 230,4 kbit/s		 208,33 26,04 4,34		 µs µs µs	
Δf_{DTRM}	Master transmission rate accuracy at 4,8 kbit/s at 38,4 kbit/s at 230,4 kbit/s	 −0,1 −0,1 −0,1	 n/a n/a n/a	 +0,1 +0,1 +0,1	 % % %	Tolerance of the transmission rate of the Master $\Delta T_{BIT}/T_{BIT}$
r	Start of detection time within a bit with reference to the raising edge of the start bit	0,65	n/a	n/a	–	Calculated in each case from the end of a bit at a UART sampling rate of 8
s	End of detection time within a bit with reference to the raising edge of the start bit	n/a	n/a	0,22	–	Calculated in each case from the end of a bit at a UART sampling rate of 8
T_{DR}	Rise time at 4,8 kbit/s at 38,4 kbit/s at 230,4 kbit/s	0 0 0 0	n/a n/a n/a n/a	0,20 41,7 5,2 869	T_{BIT} µs µs ns	With reference to the bit time unit. The minimum values could be critical to meet the requirements in H.1.5.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
t_{DF}	Fall time at 4,8 kbit/s at 38,4 kbit/s at 230,4 kbit/s	0 0 0 0	n/a n/a n/a n/a	0,20 41,7 5,2 869	T_{BIT} µs µs ns	With reference to the bit time unit. The minimum values could be critical to meet the requirements in H.1.5.
t_{ND}	Noise suppression time	n/a	n/a	1/16	T_{BIT}	Permissible duration of a receive signal above/below the detection threshold without detection taking place
t_{H}	Detection time High	1/16	n/a	n/a	T_{BIT}	Duration of a receive signal above the detection threshold for 'H' level
t_{L}	Detection time Low	1/16	n/a	n/a	T_{BIT}	Duration of a receive signal below the detection threshold for 'H' level

The parameters 'r' and 's' apply to the respective Master or Device receiver side. This definition allows for a more flexible definition of oscillator accuracy, bit distortion and slewrate on the Device side. The overall bit-width distortion on the last bit of the UART frame shall provide a correct level in the range of Figure 23.

5.3.3.3 Wake-up current pulse

The wake-up feature is used to request that a Device goes to the COMx mode.

A service call (PL_WakeUp.req) from the DL initiates the wake-up process (see 5.2.2.2).

The wake-up request (WURQ) starts with a current pulse induced by the Master (Port) for a time T_{WU}. The wake-up request comprises the following phases (see Figure 24):

a) Injection of a current IQ_{WU} by the Master depending on the level of the C/Q connection. For an input signal equivalent to logic "1" this is a current source; for an input signal equivalent to logic "0" this is a current sink.

b) Delay time of the Device until it is ready to receive.

The wake-up request pulse can be detected by the Device through a voltage change on the C/Q line or evaluation of the current of the respective driver element within the time T_{WU}. Figure 24 shows examples for Devices with low output power.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

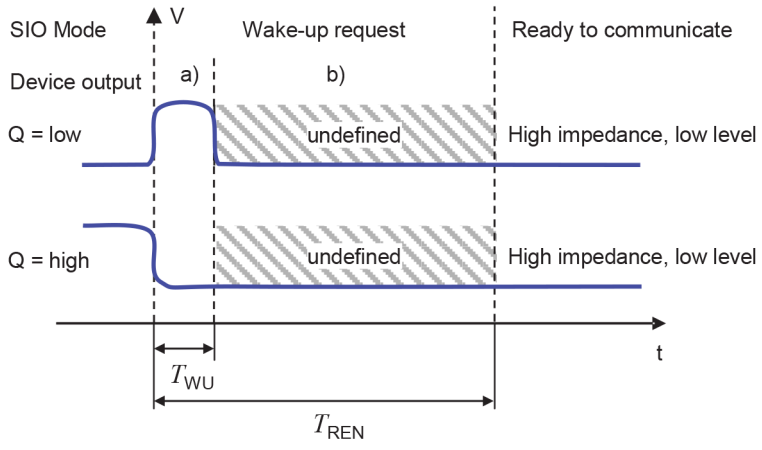


Figure 24 – Wake-up request

Table 10 specifies the current and timing properties associated with the wake-up request. See Table 6 for values of $IQPKL_M$ and $IQPKH_M$.

Table 10 – Wake-up request characteristics

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
IQ_{WU}	Amplitude of Master's wake-up current pulse	$IQPKL_M$ or $IQPKH_M$	n/a	n/a	mA	Current pulse followed by switching status of Device
T_{WU}	Duration of Master's wake-up current pulse	75	n/a	85	µs	Master property
T_{REN}	Receive enable delay	n/a	n/a	500	µs	Device property

5.4 Power supply

5.4.1 Power supply options

The SDCI connection system provides dedicated power lines in addition to the signal line. The communication section of a Device shall always be powered by the Master using the power lines defined in the 3-wire connection system (Power 1).

Manufacturers or vendors shall emphasize this requirement within the user manual of the Master. Any additional measure for further increased robustness is within the responsibility of the designer or manufacturer of the Master.

The minimum supply current available from a Master Port is specified in Table 6.

The application section of the Device can be powered in one of three ways:

– via the power lines of the SDCI 3-wire connection system (Class A Ports), using Power 1,

– via the extra power lines of the SDCI 5-wire connection system (Class B Ports), using an extra power supply at the Master (Power 2) that shall be nonreactive, which means that there is no impact on voltages and currents of Power 1 and on SDCI communications,

– via a local power supply at the Device (design specific) that shall be nonreactive to Power 1, thus guaranteeing correct communication even in case of failing local power supply.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

It is recommended for Devices not to consume more than the minimum current a Master shall support (see Table 6). This ensures easiest handling of Master/Device systems without inquiries, checking, and calculations. Whenever a Device requires more than the minimum current the capabilities of the respective Master Port and of its cabling shall be checked.

5.4.2 Port Class B

Figure 25 shows the layout of the two Port Classes A and B. Class B Ports shall be marked to distinguish them from Class A Ports due to risks deriving from incompatibilities on pin 2 and pin 5.

Power 2 on Port Class B shall meet the following requirements:

– electrical isolation of Power 2 from Power 1,
– degree of isolation according to the IEC 60664 series (clearance and creepage distances),
– electrical safety (SELV) according to IEC 61010-2-201
– direct current with P24 (+) and N24 (-),
– EMC tests shall be performed with maximum ripple and load switching,
– Device shall continue communicating correctly even in case of a failure of Power 2.

A Device designer shall ensure that Power 1 and Power 2 are always electrically isolated even in particular deployments or applications at the customer's site. Violation of this rule at one Port can have an impact on all other Ports.

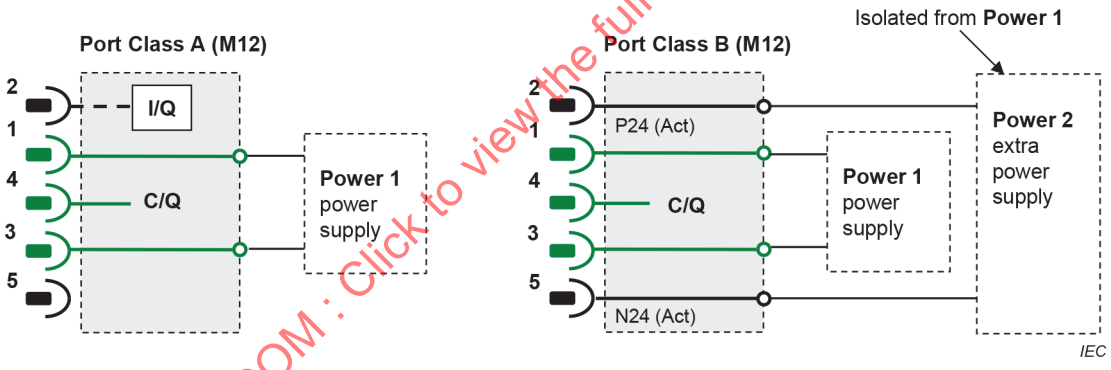


Figure 25 – Class A and B Port definitions

Table 11 shows the electrical characteristics of a Master Port Class B (M12).

Table 11 – Electrical characteristics of a Master Port Class B

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
$VP24_M$	Extra DC supply voltage for Devices	20[a]	24	30	V	
$IP24_M$	Extra DC supply current for Devices	1,6[b]	n/a	3,5[c]	A	

[a] A minimum voltage shall be guaranteed for testing at maximum recommended supply current. At the Device side 18 V shall be available in this case.

[b] Minimum current in order to guarantee a high degree of interoperability.

[c] The recommended maximum current for a wire gauge of 0,34 mm^2 and standard M12 connector is 3,5 A. The maximum current depends on the type of connector, the wire gauge, maximum temperature, and simultaneity factor of the Ports (check user manual of a Master).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

In general, the requirements of Devices shall be checked whether they meet the available capabilities of the Master. In case a simultaneity factor for Master Ports exists, it shall be documented in the user manual and be observed by the user of the Master.

5.4.3 Power-on requirements

The power-on requirements are specified in 5.3.2.3 and 5.3.2.4.

5.5 Medium

5.5.1 Connectors

The Master and Device pin assignment is based on the specifications in IEC 60947-5-2, with extensions specified in the paragraphs below.

Ports Class A use M5, M8, and M12 connectors, with a maximum of four pins.

Ports Class B only use M12 connectors with five pins.

M12 connectors shall mechanically be A-coded according to IEC 61076-2-101.

For legacy or compatibility reasons, direct wiring or different types of connectors may be used instead, provided that they do not violate the electrical characteristics and use signal naming specified in this document.

Female connectors are assigned to the Master. Table 12 lists the pin assignments and Figure 26 shows the layout and mechanical coding for M12, M8, and M5 connections.

Table 12 – Master pin assignments

Pin	Signal	Designation	Remark
1	L+	Power supply (+)	See Table 6
2	I/Q	NC/DI(OSSDe)/DO (Port Class A)	Option 1: NC (not connected) Option 2: DI Option 3: DI, then configured DO Option 4: OSSDe (see [10])
	P24	P24 (Port Class B)	Extra power supply for power Devices (Port Class B)
3	L−	Power supply (−)	See Table 6
4	C/Q	SIO(OSSDe)/SDCI	Standard I/O mode (DI/DO) or SDCI (see Table 6 for electrical characteristics of DO). See [10] for OSSDe definitions.
5	NC	NC (Port Class A)	Shall not be connected on the Master side (Port Class A).
	N24	N24 (Port Class B)	Reference potential to the extra power supply (Port Class B)
NOTE M12 is always a 5-pin version on the Master side (female).			

Figure 26 shows the layout of the two Port Classes A and B. Class B Ports shall be marked to distinguish them from Class A Ports, because of risks deriving from incompatibilities.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

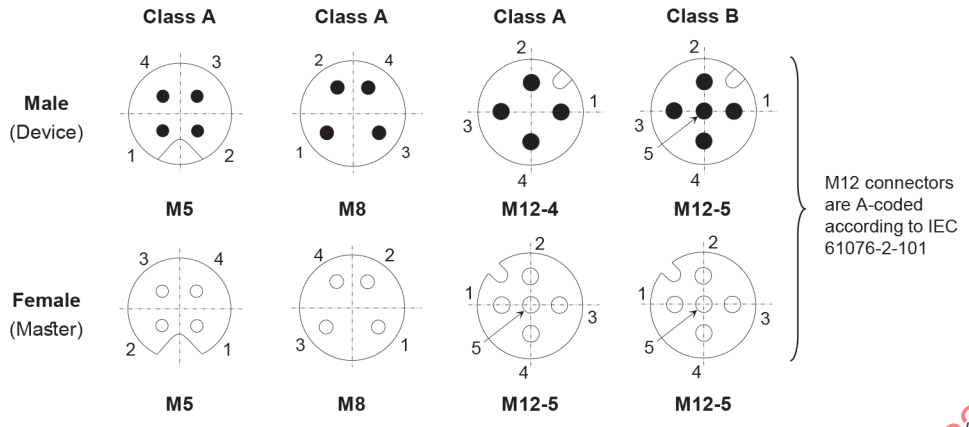


Figure 26 – Pin layout front view

Male connectors are assigned to the Device. Table 13 lists the pin assignments.

Table 13 – Device pin assignments

Pin	Signal	Designation	Remark
1	L+	Power supply (+)	See Table 7
2	I/Q [a]	NC/DI(OSSDe)/DO/ AI/AO (Port Class A)	Option 1: NC (not connected) Option 2: DI (Master's view) Option 3: DO (Master's view) Option 4: Analog signal (I/U) [d] Option 5: OSSDe (see [10])
	P24 [b]	P24 (Port Class B)	Extra power supply for power Devices (Port Class B)
3	L−	Power supply (−)	See Table 7
4	C/Q [c]	SIO(OSSDe)/SDCI	Standard I/O mode (DI/DO) or SDCI (see Table 6 for electrical characteristics of DO). See [10] for OSSDe definitions.
5	NC	NC (Port Class A)	Option 1: Shall not be connected on the Device side (Port Class A).
	N24	N24 (Port Class B)	Option 2: Reference to the extra power supply (Port Class B)

[a] Device signals shall not impact I/Q input of a Master. Devices shall withstand permanent direct current (see Table 6) on the Master side.

[b] Devices relying on Port Class A shall use a 3-wire connection in this case in order to avoid bypassing galvanic isolation.

[c] A Master shall always be able to establish and maintain SDCI communication without interferences.

[d] Typical for U is 0 V to 10 V, 1 V to 5 V, and for I is 0 mA to 20 mA, 4 mA to 20 mA.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

5.5.2 Cable

The transmission medium for SDCI communication is a multi-wired cable with three or more wires. The definitions in the following paragraphs implicitly cover the static voltage definitions in Table 5 and Figure 17. To ensure functional reliability, the cable properties shall comply with Table 14.

Table 14 – Cable characteristics

Property	Minimum	Typical	Maximum	Unit
Length L	0	n/a	20	m
Overall loop resistance RL_{eff} [a]	n/a	n/a	6,0 (for a current of 200 mA) 1,2 (for a current of 1 000 mA)	Ω
Effective line capacitance CL_{eff}	n/a	n/a	3,0	nF (< 1 MHz)

[a] The overall loop resistence shall be rated such that minimum Device supply voltages are guaranteed at maximum supply current (see Table 7).

The loop resistance RL_{eff} and the effective line capacitance CL_{eff} can be measured as demonstrated in Figure 27.

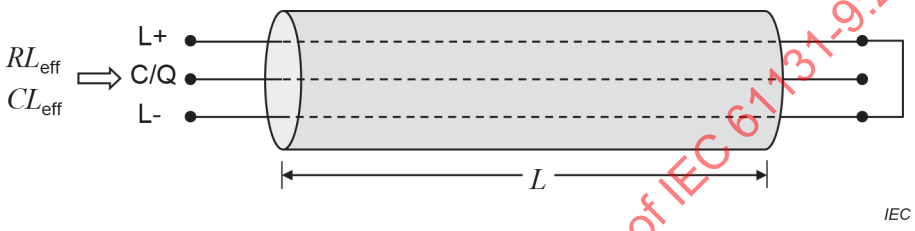


Figure 27 – Reference schematics for effective line capacitance and loop resistance

Table 15 shows the cable conductors and their assigned color codes.

Table 15 – Cable conductor assignments

Signal	Designation	Color	Remark
L−	Power supply (−)	Blue[a]	SDCI 3-wire connection system
C/Q	Communication signal	Black[a]	SDCI 3-wire connection system
L+	Power supply (+)	Brown[a]	SDCI 3-wire connection system
I/Q	DI or DO	White[a]	Optional
P24	Extra power supply (+)	Any other	Optional
N24	Extra power supply (−)	Any other	Optional

[a] Corresponding to IEC 60947-5-2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

6 Standard Input and Output (SIO)

Figure 85 and Figure 96 demonstrate how the SIO mode allows a Device to bypass the SDCI communication layers and to map the DI or DO signal directly into the data exchange message of the upper-level fieldbus or system. Changing between the SDCI and SIO mode is defined by the user configuration or implicitly by the services of the Master applications. The System Management takes care of the corresponding initialization or deactivation of the SDCI communication layers and the physical layer (mode switch). The characteristics of the interfaces for the DI and DO signals are derived from the caracteristics specified in IEC 61131-2 for type 1.

7 Data link layer (DL)

7.1 General

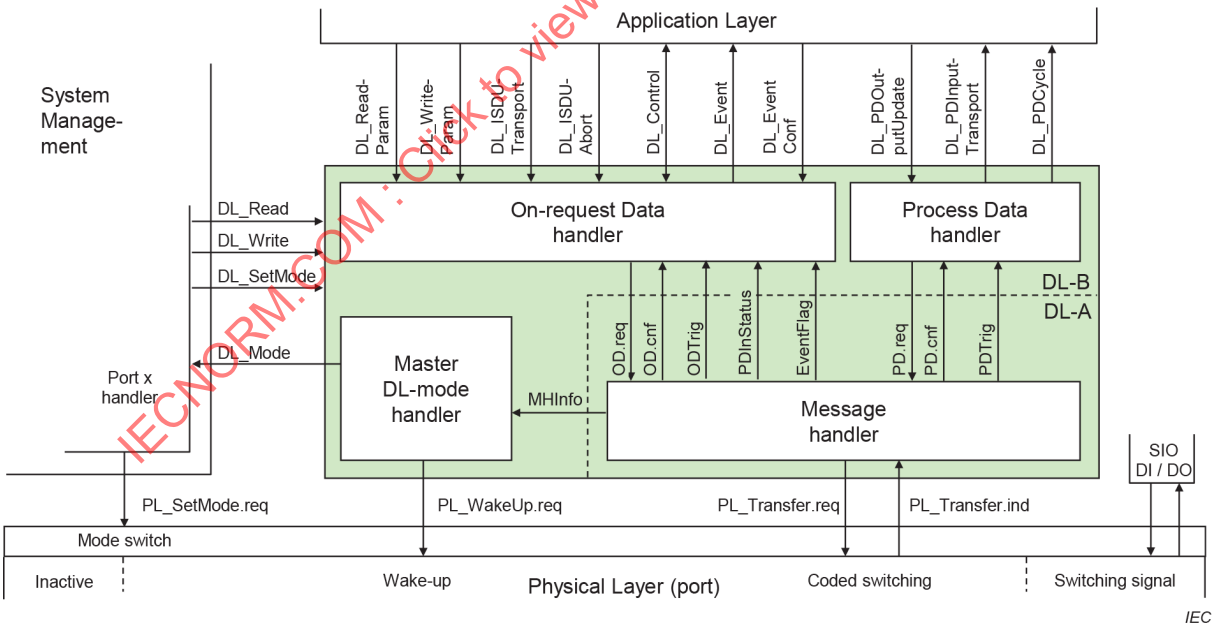
The data link layers of SDCI are concerned with the delivery of messages between a Master and a Device across the physical link. It uses several M-sequence ("message sequence") types for different data categories.

A set of DL-services is available to the application layer (AL) for the exchange of Process Data (PD) and On-request Data (OD). Another set of DL-services is available to System Management (SM) for the retrieval of Device identification parameters and the setting of state machines within the DL. The DL uses PL-Services for controlling the physical layer (PL) and for exchanging UART frames. The DL takes care of the error detection of messages (whether internal or reported from the PL) and the appropriate remedial measures (e.g. retry).

The data link layers are structured due to the nature of the data categories into Process Data handlers and On-request Data handlers which are in turn using a message handler to deal with the requested transmission of messages. The special modes of Master Ports such as wake-up, COMx, and SIO (disable communication) require a dedicated DL-mode handler within the Master DL. The special wake-up signal modulation requires signal detection on the Device side and thus a DL-mode handler within the Device DL. Each handler comprises its own state machine.

The data link layer is subdivided in a DL-A section with its own internal services and a DL-B section with the external services. The DL uses additional internal administrative calls between the handlers which are defined in the "internal items" section of the associated state-transition tables. Figure 28 shows an overview of the structure and the services of the Master's data link layer.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022



NOTE This figure uses the conventions in 3.3.5.

Figure 28 – Structure and services of the data link layer (Master)

Figure 29 shows an overview of the structure and the services of the Device's data link layer.

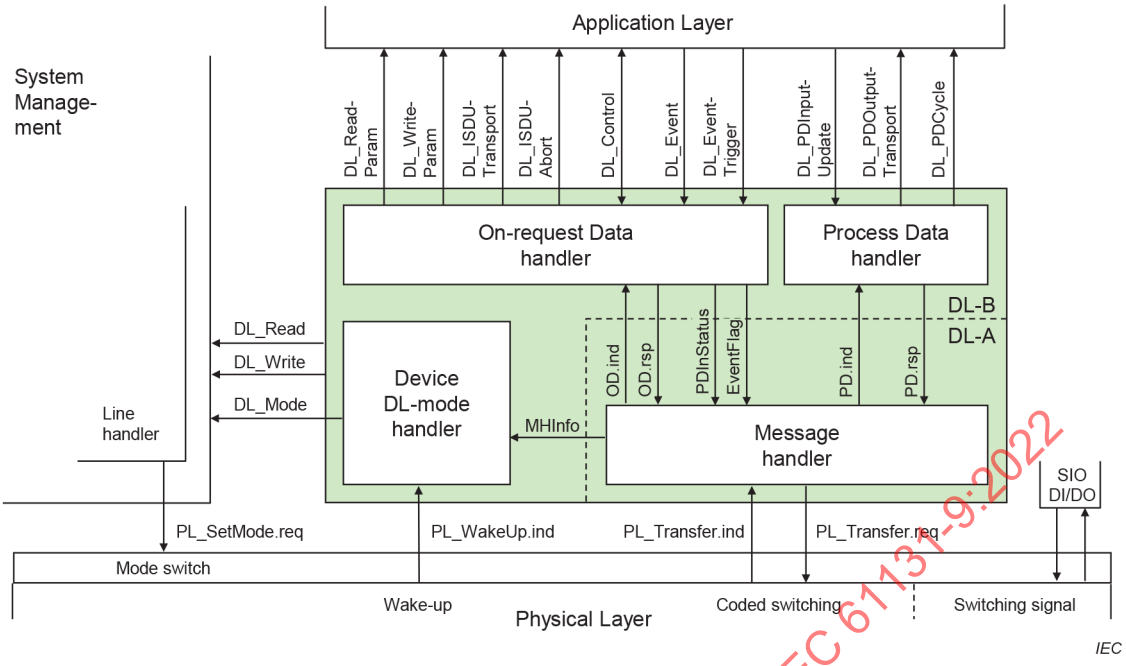


Figure 29 – Structure and services of the data link layer (Device)

7.2 Data link layer services

7.2.1 DL-B services

7.2.1.1 Overview of services within Master and Device

This Clause 7.2.1 defines the services of the data link layer to be provided to the application layer and System Management via its external interfaces. Table 16 lists the assignments of Master and Device to their roles as initiator or receiver for the individual DL services. Empty fields indicate no availability of this service on Master or Device.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 16 – Service assignments within Master and Device

Service name	Master	Device
DL_ReadParam	R	I
DL_WriteParam	R	I
DL_ISDUTransport	R	I
DL_ISDUAbort	R	I
DL_PDOutputUpdate	R	
DL_PDOutputTransport		I
DL_PDInputUpdate		R
DL_PDInputTransport	I	
DL_PDCycle	I	I
DL_SetMode	R	
DL_Mode	I	
DL_Event	I	R
DL_EventConf	R	
DL_EventTrigger		R
DL_Control	I / R	R / I
DL_Read	R	I
DL_Write	R	I
Key (see 3.3.4) I Initiator of service R Receiver (responder) of service		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

See 3.3 for conventions and how to read the service descriptions in 7.2, 8.2, 9.2.2, and 9.3.2.

7.2.1.2 DL_ReadParam

The DL_ReadParam service is used by the AL to read a parameter value from the Device via the page communication channel. The parameters of the service primitives are listed in Table 17.

Table 17 – DL_ReadParam

Parameter name	.req	.cnf	.ind	.rsp
Argument	M		M	
Address	M		M	
Result (+)		S		S
Value		M		M
Result (−)		S		
ErrorInfo		M		

Argument

The service-specific parameters are transmitted in the argument.

Address

This parameter contains the address of the requested Device parameter, i.e. the Device parameter addresses within the page communication channel (see Table B.1).

Permitted values: 0 to 31.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Value

This parameter contains read Device parameter values.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

7.2.1.3 DL_WriteParam

The DL_WriteParam service is used by the AL to write a parameter value to the Device via the page communication channel. The parameters of the service primitives are listed in Table 18.

Table 18 – DL_WriteParam

Parameter name	.req	.cnf	.ind
Argument	M		M
Address	M		M
Value	M		M
Result (+)		S	
Result (−)		S	
ErrorInfo		M	

Argument

The service-specific parameters are transmitted in the argument.

Address

This parameter contains the address of the requested Device parameter, i.e. the Device parameter addresses within the page communication channel.

Permitted values: 16 to 31, in accordance with Device parameter access rights.

Value

This parameter contains the Device parameter value to be written.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

7.2.1.4 DL_Read

The DL_Read service is used by System Management to read a Device parameter value via the page communication channel. The parameters of the service primitives are listed in Table 19.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 19 – DL_Read

Parameter name	.req	.cnf	.ind	.rsp
Argument	M		M	
Address	M		M	
Result (+)		S		S
Value		M		M
Result (−)		S		
ErrorInfo		M		

Argument

The service-specific parameters are transmitted in the argument.

Address

This parameter contains the address of the requested Device parameter, i.e. the Device parameter addresses within the page communication channel (see Table B.1).

Permitted values: 0 to 15, in accordance with Device parameter access rights.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Value

This parameter contains read Device parameter values.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

7.2.1.5 DL_Write

The DL_Write service is used by System Management to write a Device parameter value to the Device via the page communication channel. The parameters of the service primitives are listed in Table 20.

Table 20 – DL_Write

Parameter name	.req	.cnf	.ind
Argument	M		M
Address	M		M
Value	M		M
Result (+)		S	
Result (−)		S	
ErrorInfo		M	

Argument

The service-specific parameters are transmitted in the argument.

Address

This parameter contains the address of the requested Device parameter, i.e. the Device parameter addresses within the page communication channel.

Permitted values: 0 to 15, in accordance with parameter access rights.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Value

This parameter contains the Device parameter value to be written.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

7.2.1.6 DL_ISDUTransport

The DL_ISDUTransport service is used to transport an ISDU. This service is used by the Master to send a service request from the Master application layer to the Device. It is used by the Device to send a service response to the Master from the Device application layer. The parameters of the service primitives are listed in Table 21.

Table 21 – DL_ISDUTransport

Parameter name	.req	.ind	.cnf	.rsp
Argument	M	M		
ValueList	M	M		
Result (+)			S	S
Data			C	C
Qualifier			M	M
Result (−)			S	S
ISDUTransportErrorInfo			M	M

Argument

The service-specific parameters are transmitted in the argument.

ValueList

This parameter contains the relevant operating parameters.

Parameter type: Record.

Index

Permitted values: 2 to 65 535 (See B.2.1 for constraints).

Subindex

Permitted values: 0 to 255.

Data

Parameter type: Octet string.

Direction

Permitted values:
READ (Read operation),
WRITE (Write operation).

Result (+):

This selection parameter indicates that the service has been executed successfully.

Data

Parameter type: Octet string.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Qualifier

Permitted values: an I-Service Device response according to Table A.12.

Result (−):

This selection parameter indicates that the service failed.

ISDUTransportErrorInfo

This parameter contains error information.

Permitted values:

NO_COMM	(no communication available),
STATE_CONFLICT	(service unavailable within current state),
ISDU_TIMEOUT	(ISDU acknowledgment time elapsed, see Table 102),
ISDU_NOT_SUPPORTED	(ISDU not implemented),
VALUE_OUT_OF_RANGE	(Service parameter value violates range definitions).

7.2.1.7 DL_ISDUAbort

The DL_ISDUAbort service aborts the current ISDU transmission. This service has no parameters. The service primitives are listed in Table 22.

Table 22 – DL_ISDUAbort

Parameter name	.req	.cnf
<none>		

The service returns with the confirmation after abortion of the ISDU transmission.

7.2.1.8 DL_PDOutputUpdate

The Master's application layer uses the DL_PDOutputUpdate service to update the output data (Process Data from Master to Device) on the data link layer. The parameters of the service primitives are listed in Table 23.

Table 23 – DL_PDOutputUpdate

Parameter name	.req	.cnf
Argument	M	
OutputData	M	
Result (+)		S
TransportStatus		M
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

OutputData

This parameter contains the Process Data provided by the application layer.

Parameter type: Octet string.

Result (+):

This selection parameter indicates that the service has been executed successfully.

TransportStatus

This parameter indicates whether the data link layer is in a state permitting data to be transferred to the communication partner(s).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Permitted values:
YES (data transmission permitted),
NO (data transmission not permitted).

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

7.2.1.9 DL_PDOutputTransport

The data link layer on the Device uses the DL_PDOutputTransport service to transfer the content of output Process Data to the application layer (from Master to Device). The parameters of the service primitives are listed in Table 24.

Table 24 – DL_PDOutputTransport

Parameter name	.ind
Argument	M
OutputData	M

Argument

The service-specific parameters are transmitted in the argument.

OutputData

This parameter contains the Process Data to be transmitted to the application layer.

Parameter type: Octet string.

7.2.1.10 DL_PDInputUpdate

The Device's application layer uses the DL_PDInputUpdate service to update the input data (Process Data from Device to Master) on the data link layer. The parameters of the service primitives are listed in Table 25.

Table 25 – DL_PDInputUpdate

Parameter name	.req	.cnf
Argument	M	
InputData	M	
Result (+)		S
TransportStatus		M
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

InputData

This parameter contains the Process Data provided by the application layer.

Result (+):

This selection parameter indicates that the service has been executed successfully.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TransportStatus

This parameter indicates whether the data link layer is in a state permitting data to be transferred to the communication partner(s).

Permitted values:
YES (data transmission permitted),
NO (data transmission not permitted).

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

7.2.1.11 DL_PDInputTransport

The data link layer on the Master uses the DL_PDInputTransport service to transfer the content of input data (Process Data from Device to Master) to the application layer. The parameters of the service primitives are listed in Table 26.

Table 26 – DL_PDInputTransport

Parameter name	.ind
Argument	M
InputData	M

Argument

The service-specific parameters are transmitted in the argument.

InputData

This parameter contains the Process Data to be transmitted to the application layer.

Parameter type: Octet string.

7.2.1.12 DL_PDCycle

The data link layer uses the DL_PDCycle service to indicate the end of a Process Data cycle to the application layer. This service has no parameters. The service primitives are listed in Table 27.

Table 27 – DL_PDCycle

Parameter name	.ind
<none>	

7.2.1.13 DL_SetMode

The DL_SetMode service is used by System Management to set up the data link layer's state machines and to send the characteristic values required for operation to the data link layer. The parameters of the service primitives are listed in Table 28.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 28 – DL_SetMode

Parameter name	.req	.cnf

Argument
  Mode
  ValueList
	M
	M
	U	

Result (+)		S
Result (−) ErrorInfo		S M

Argument

The service-specific parameters are transmitted in the argument.

Mode

This parameter indicates the requested mode of the Master's DL on an individual Port.

Permitted values:
INACTIVE (handler shall change to the INACTIVE state),
STARTUP (handler shall change to the STARTUP state),
PREOPERATE (handler shall change to the PREOPERATE state),
OPERATE (handler shall change to the OPERATE state).

ValueList

This parameter contains the relevant operating parameters.

Data structure: record.

M-sequenceTime: (to be propagated to message handler)

M-sequenceType: (to be propagated to message handler)

Permitted values:
TYPE_0,
TYPE_1_1, TYPE_1_2, TYPE_1_V,
TYPE_2_1, TYPE_2_2, TYPE_2_3, TYPE_2_4, TYPE_2_5, TYPE_2_V
(TYPE_1_1 forces interleave mode of Process and On-request Data transmission, see 7.3.4.2).

PDInputLength: (to be propagated to message handler)

PDOutputLength: (to be propagated to message handler)

OnReqDataLengthPerMessage: (to be propagated to message handler)

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (service unavailable within current state),
PARAMETER_CONFLICT (consistency of parameter set violated).

7.2.1.14 DL_Mode

The DL uses the DL_Mode service to report to System Management that a certain operating status has been reached. The parameters of the service primitives are listed in Table 29.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 29 – DL_Mode

Parameter name	.ind
Argument	M
RealMode	M

Argument

The service-specific parameters are transmitted in the argument.

RealMode

This parameter indicates the status of the DL-mode handler.

Permitted values:

INACTIVE	(Handler changed to the INACTIVE state),
COM1	(COM1 mode established),
COM2	(COM2 mode established),
COM3	(COM3 mode established),
COMLOST	(Lost communication),
ESTABCOM	(Handler changed to the EstablishCom state),
STARTUP	(Handler changed to the STARTUP state),
PREOPERATE	(Handler changed to the PREOPERATE state),
OPERATE	(Handler changed to the OPERATE state).

7.2.1.15 DL_Event

The service DL_Event indicates a pending status or error information. The cause for an Event is located in a Device and the Device application triggers the Event transfer. The parameters of the service primitives are listed in Table 30.

Table 30 – DL_Event

Parameter name	.req	.ind
Argument	M	M
Instance	M	M
Type	M	M
Mode	M	M
EventCode	M	M
EventsLeft		M

Argument

The service-specific parameters are transmitted in the argument.

Instance

This parameter indicates the Event source.

Permitted values: Application (see Table A.17).

Type

This parameter indicates the Event category.

Permitted values: ERROR, WARNING, NOTIFICATION (see Table A.19).

Mode

This parameter indicates the Event mode.

Permitted values: SINGLESHOT, APPEARS, DISAPPEARS (see Table A.20).

EventCode

This parameter contains a code identifying a certain Event (see Table D.1).

Parameter type: 16-bit unsigned integer.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

EventsLeft

This parameter indicates the number of unprocessed Events.

7.2.1.16 DL_EventConf

The DL_EventConf service confirms the transmitted Events via the Event handler. This service has no parameters. The service primitives are listed in Table 31.

Table 31 – DL_EventConf

Parameter name	.req	.cnf
<none>		

7.2.1.17 DL_EventTrigger

The DL_EventTrigger request starts the Event signaling (see Event flag in Figure A.3) and freezes the Event memory within the DL. The confirmation is returned after the activated Events have been processed. Additional DL_EventTrigger requests are ignored until the previous one has been confirmed (see 7.3.8, 8.3.3 and Figure 66). This service has no parameters. The service primitives are listed in Table 32.

Table 32 – DL_EventTrigger

Parameter name	.req	.cnf
<none>		

7.2.1.18 DL_Control

The Master uses the DL_Control service to convey control information via the MasterCommand mechanism to the corresponding Device application and to get control information via the PD status flag mechanism (see A.1.5) and the PDInStatus service (see 7.2.2.5). The parameters of the service primitives are listed in Table 33.

Table 33 – DL_Control

Parameter name	.req	.ind
Argument	M	M
ControlCode	M	M(=)

Argument

The service-specific parameters are transmitted in the argument.

ControlCode

This parameter indicates the qualifier status of the Process Data (PD).

Permitted values:
VALID (Input Process Data valid; see 7.2.2.5, 8.2.2.12),
INVALID (Input Process Data invalid),
PDOUTVALID (Output Process Data valid; see 7.3.7.1),
PDOUTINVALID (Output Process Data invalid or missing).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.2 DL-A services

7.2.2.1 Overview

According to 7.1 the data link layer is split into the upper layer DL-B and the lower layer DL-A. The layer DL-A comprises the message handler as shown in Figure 28 and Figure 29.

The Master message handler encodes commands and data into messages and sends these to the connected Device via the physical layer. It receives messages from the Device via the physical layer and forwards their content to the corresponding handlers in the form of a confirmation. When the "Event flag" is set in a Device message (see A.1.5), the Master message handler invokes an EventFlag service to prompt the Event handler.

The Master message handler shall employ a retry strategy following a corrupted message, i.e. upon receiving an incorrect checksum from a Device, or no checksum at all. In these cases, the Master shall repeat the Master message two times (see Table 102). If the retries are not successful, a negative confirmation shall be provided, and the Master shall re-initiate the communication via the Port-x handler beginning with a wake-up.

After a start-up phase the message handler performs cyclic operation with the M-sequence type and cycle time provided by the DL_SetMode service.

Table 34 lists the assignment of Master and Device to their roles as initiator (I) or receiver (R) in the context of the execution of their individual DL-A services.

Table 34 – DL-A services within Master and Device

Service name	Master	Device
OD	R	I
PD	R	I
EventFlag	I	R
PDInStatus	I	R
MHInfo	I	I
ODTrig	I	
PDTrig	I	

7.2.2.2 OD

The OD service is used to set up the On-request Data for the next message to be sent. In turn, the confirmation of the service contains the data from the receiver. The parameters of the service primitives are listed in Table 35.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 35 – OD

Parameter name	.req	.ind	.rsp	.cnf
Argument	M	M		
RWDirection	M	M		
ComChannel	M	M		
AddressCtrl	M	M		
Length	M	M		
Data	C	C		
Result (+)			S	S
Data			C	C(=)
Length			M	M
Result (−)			S	S
ErrorInfo			M	M(=)

Argument

The service-specific parameters are transmitted in the argument.

RWDirection

This parameter indicates the read or write direction.

Permitted values:
READ (Read operation),
WRITE (Write operation).

ComChannel

This parameter indicates the selected communication channel for the transmission.

Permitted values: DIAGNOSIS, PAGE, ISDU (see Table A.1).

AddressCtrl

This parameter contains the address or flow control value (see A.1.2).

Permitted values: 0 to 31.

Length

This parameter contains the length of data to transmit.

Permitted values: 0 to 32.

Data

This parameter contains the data to transmit.

Data type: Octet string.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Data

This parameter contains the read data values.

Length

This parameter contains the length of the received data package.

Permitted values: 0 to 32.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_COMM (no communication available),
STATE_CONFLICT (service unavailable within current state).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.2.3 PD

The PD service is used to set up the Process Data to be sent through the process communication channel. The confirmation of the service contains the data from the receiver. The parameters of the service primitives are listed in Table 36.

Table 36 – PD

Parameter name	.req	.ind	.rsp	.cnf
Argument	M	M		
PDInAddress	C	C(=)		
PDInLength	C	C(=)		
PDOut	C	C(=)		
PDOutAddress	C	C(=)		
PDOutLength	C	C(=)		
Result (+)			S	S
PDIn			C	C(=)
Result (−)			S	S
ErrorInfo			M	M(=)

Argument

The service-specific parameters are transmitted in the argument.

PDInAddress

This parameter contains the address of the requested input Process Data (see 7.3.4.2).

PDInLength

This parameter contains the length of the requested input Process Data.

Permitted values: 0 to 32.

PDOut

This parameter contains the Process Data to be transferred from Master to Device.

Data type: Octet string.

PDOutAddress

This parameter contains the address of the transmitted output Process Data (see 7.3.4.2).

PDOutLength

This parameter contains the length of the transmitted output Process Data.

Permitted values: 0 to 32.

Result (+)

This selection parameter indicates that the service has been executed successfully.

PDIn

This parameter contains the Process Data to be transferred from Device to Master.

Data type: Octet string.

Result (−)

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:

NO_COMM	(no communication available),
STATE_CONFLICT	(service unavailable within current state).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.2.4 EventFlag

The EventFlag service sets or signals the status of the "Event flag" (see A.1.5) during cyclic communication. The parameters of the service primitives are listed in Table 37.

Table 37 – EventFlag

Parameter name	.ind	.req
Argument Flag	M	M

Argument

The service-specific parameters are transmitted in the argument.

Flag

This parameter contains the value of the "Event flag".

Permitted values:
TRUE ("Event flag" = 1),
FALSE ("Event flag" = 0).

7.2.2.5 PDInStatus

The service PDInStatus sets and signals the validity qualifier of the input Process Data. The parameters of the service primitives are listed in Table 38.

Table 38 – PDInStatus

Parameter name	.req	.ind
Argument Status	M	M

Argument

The service-specific parameters are transmitted in the argument.

Status

This parameter contains the validity indication of the transmitted input Process Data.

Permitted values:
VALID (Input Process Data valid based on PD status flag (see A.1.5); see 7.2.1.18),
INVALID (Input Process Data invalid).

7.2.2.6 MHInfo

The service MHInfo signals an exceptional operation within the message handler. The parameters of the service are listed in Table 39.

Table 39 – MHInfo

Parameter name	.ind
Argument MHInfo	M

Argument

The service-specific parameters are transmitted in the argument.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

MHInfo

This parameter contains the exception indication of the message handler.

Permitted values:

COMLOST	(lost communication),
ILLEGAL_MESSAGETYPE	(unexpected M-sequence type detected),
CHECKSUM_MISMATCH	(Checksum error detected).

7.2.2.7 ODTrig

The service ODTrig is only available on the Master. The service triggers the On-request Data handler and the ISDU, Command, or Event handler currently in charge to provide the On-request Data (via the OD service) for the next Master message. The parameters of the service are listed in Table 40.

Table 40 – ODTrig

Parameter name	.ind
Argument DataLength	M

Argument

The service-specific parameters are transmitted in the argument.

DataLength

This parameter contains the available space for On-request Data (OD) per message.

7.2.2.8 PDTrig

The service PDTrig is only available on the Master. The service triggers the Process Data handler to provide the Process Data (PD) for the next Master message.

The parameters of the service are listed in Table 41.

Table 41 – PDTrig

Parameter name	.ind
Argument DataLength	M

Argument

The service-specific parameters are transmitted in the argument.

DataLength

This parameter contains the available space for Process Data (PD) per message.

7.3 Data link layer protocol

7.3.1 Overview

Figure 28 and Figure 29 show the structure of the data link layer and its components: a DL-mode handler, a message handler, a Process Data handler, and an On-request Data handler to provide the specified services. Subclauses 7.3.2 to 7.3.8 define the behavior (dynamics) of these handlers by means of UML state machines and transition tables.

The On-request Data handler supports three independent types of data: ISDU, command and Event. Therefore, three additional state machines are working together with the On-request Data handler state machine as shown in Figure 30.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

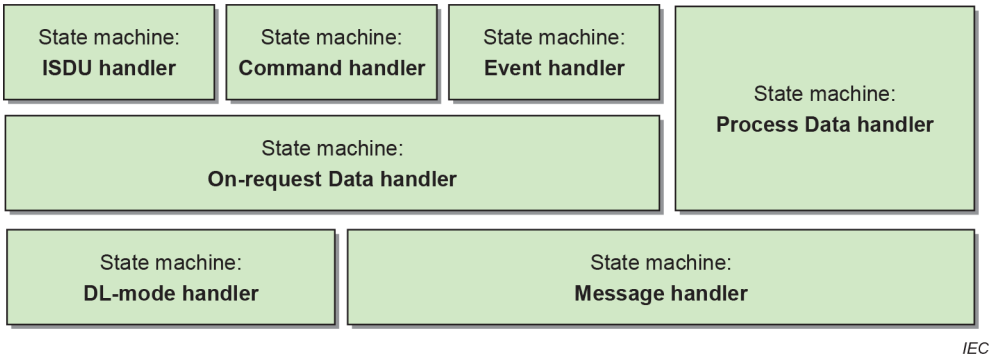


Figure 30 – State machines of the data link layer

Supplementary sequence or activity diagrams demonstrate certain use cases. See IEC TR 62390 [14] and ISO/IEC 19505 (all parts) [19].

The elements each handler is dealing with, such as messages, wake-up procedures, interleave mode, ISDU (Indexed Service Data Units), and Events are defined within the context of the respective handler.

7.3.2 DL-mode handler

7.3.2.1 General

The Master DL-mode handler shown in Figure 28 is responsible for setting up the SDCI communication using services of the Physical Layer (PL) and internal administrative calls to control and monitor the message handler as well as the states of other handlers.

The Device DL-mode handler shown in Figure 29 is responsible for detecting a wake-up request and for establishing communication. It receives MasterCommands to synchronize with the Master DL-mode handler states STARTUP, PREOPERATE, and OPERATE and manages the activation and de-activation of handlers as appropriate.

7.3.2.2 Wake-up procedures and Device conformity rules

System Management triggers the following actions on the data link layer with the help of the DL_SetMode service (requested mode = STARTUP).

The Master DL-mode handler tries to establish communication via a wake-up request (PL_WakeUp.req) followed by a test message with M-sequence TYPE_0 (read "MinCycleTime") according to the sequence shown in Figure 31.

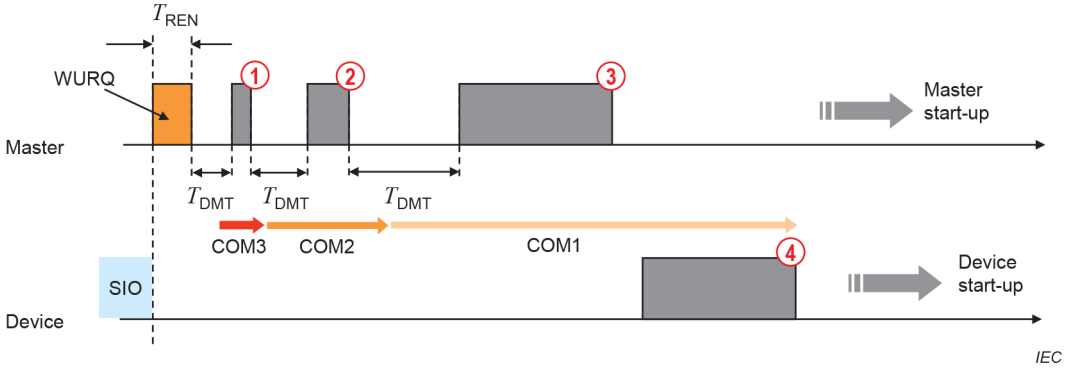


Figure 31 – Example of an attempt to establish communication

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

After the wake-up request (WURQ), specified in 5.3.3.3, the DL-mode handler requests the message handler to send the first test message after a time T_{REN} (see Table 10) and T_{DMT} (see Table 42). The specified transmission rates of COM1, COM2, and COM3 are used in descending order until a response is obtained, as shown in the example of Figure 31:

Step ①: Master message with transmission rate of COM3 (see Table 9).

Step ②: Master message with transmission rate of COM2 (see Table 9).

Step ③: Master message with transmission rate of COM1 (see Table 9).

Step ④: Device response message with transmission rate of COM1.

Before initiating a (new) message, the DL-mode handler shall wait at least for a time of T_{DMT}. T_{DMT} is specified in Table 42.

The following conformity rule applies for Devices regarding support of transmission rates:

- a Device shall support only one of the transmission rates of COM1, COM2, or COM3.

If an attempt to establish communication fails, the Master DL-mode handler shall not start a new retry wake-up procedure until after a time T_{DWU} as shown in Figure 32 and specified in Table 42.

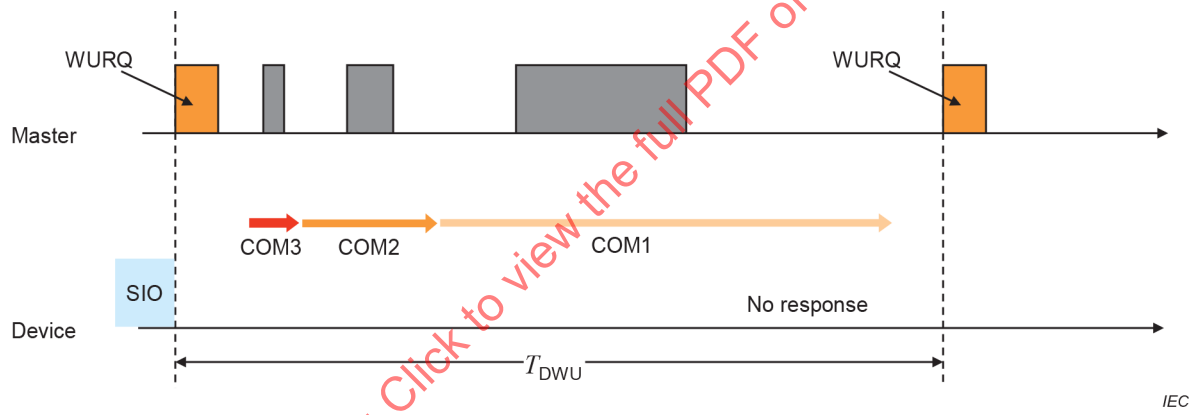


Figure 32 – Failed attempt to establish communication

The Master shall make up to n_{WU}+1 successive wake-up requests as shown in Figure 33. If this initial wake-up retry sequence fails, the Device shall reset its C/Q line to SIO mode after a time T_{DSIO} (T_{DSIO} is retrigged in the Device after each detected WURQ). The Master shall not trigger a new wake-up retry sequence until after a time T_{SD}.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

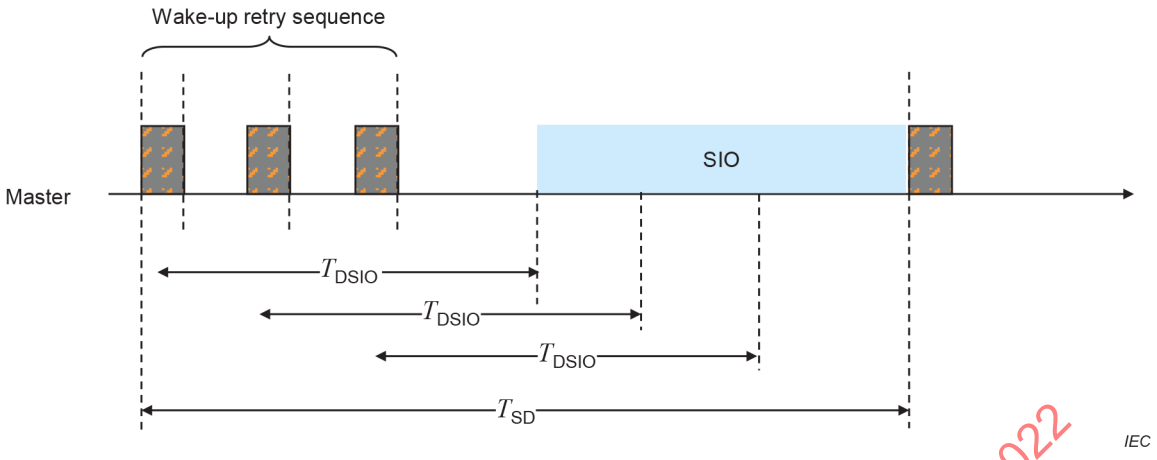


Figure 33 – Retry strategy to establish communication

The DL of the Master shall request the PL to go to SIO mode after a failed wake-up retry sequence.

The values for the timings of the wake-up procedures and retries are specified in Table 10 and Table 42. They are defined from a Master's point of view.

Table 42 – Wake-up procedure and retry characteristics

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
T_{DMT}	Master message delay	27	n/a	37	T_{BIT}	Bit time of subsequent data transmission rate
T_{DSIO}	Standard IO delay	60	n/a	300	ms	After T_{DSIO} the Device falls back to SIO mode (if supported)
T_{DWU}	Wake-up retry delay	30	n/a	50	ms	After T_{DWU} the Master repeats the wake-up request
n_{WU}	Wake-up retry count	2	2	2		Number of wake-up request retries
T_{SD}	Device detection time	0,5	n/a	1	s	Time between 2 wake-up request sequences (See NOTE)
NOTE Characteristic of the Master.						

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The Master's data link layer shall stop the establishing communication procedure once it finds a communicating Device and shall report the detected COMx-Mode to System Management using a DL_Mode indication. If the procedure fails, a corresponding error is reported using the same service.

7.3.2.3 Fallback procedure

System Management induces the following actions on the data link layer with the help of the DL_SetMode service (mode = INACTIVE):

– A MasterCommand "Fallback" (see Table B.2) forces the Device to change to the SIO mode.
– The Device shall accomplish the transition to the SIO mode after three MasterCycleTimes and/or within maximum T_{FBD} after the MasterCommand "Fallback". This allows for possible retries if the MasterCommand failed, indicated through a negative Device response.

– The Master shall ensure waiting at least maximum T_{FBD} before initiating the next start-up procedure.

Figure 34 shows the fallback procedure and its retry and timing constraints.

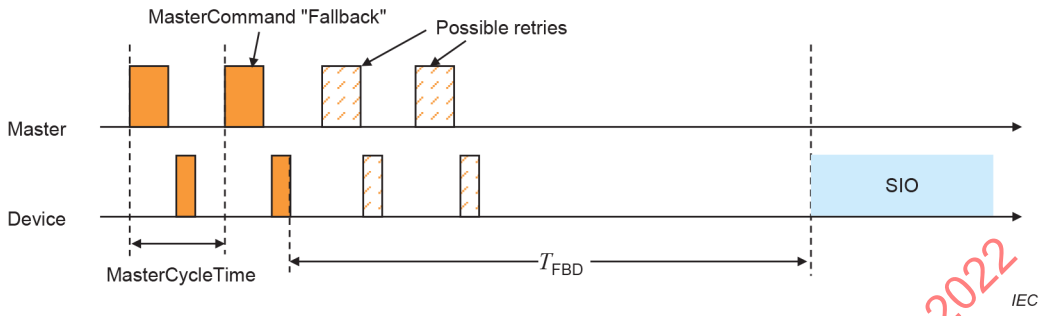


Figure 34 – Fallback procedure

Table 43 specifies the fallback timing characteristics. See A.2.6 for details.

Table 43 – Fallback timing characteristics

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
T_{FBD}	Fallback delay	3 MasterCycle-Times (OPERATE) or 3 $T_{initcyc}$ (PREOPERATE)	n/a	500	ms	After a time T_{FBD} the Device shall be switched to SIO mode (see Figure 34)

7.3.2.4 State machine of the Master DL-mode handler

Figure 35 shows the state machine of the Master DL-mode handler.

NOTE The conventions of the UML diagram types are defined in 3.3.8.

After reception of the service DL_SetMode_STARTUP from System Management, the DL-mode handler shall first create a wake-up current pulse via the PL_WakeUp service and then establish communication. This procedure is specified in submachine 1 in Figure 36.

The purpose of state "Startup_2" is to check a Device's identity via the data of the Direct Parameter page (see Figure 6). In state "PreOperate_3", the Master assigns parameters to the Device using ISDUs. Cyclic exchange of Process Data is performed in state "Operate". Within this state additional On-request Data such as ISDUs, commands, and Events can be transmitted using appropriate M-sequence types (see Figure 39).

In state PreOperate_3 and Operate_4 different sets of handlers within the Master are activated.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

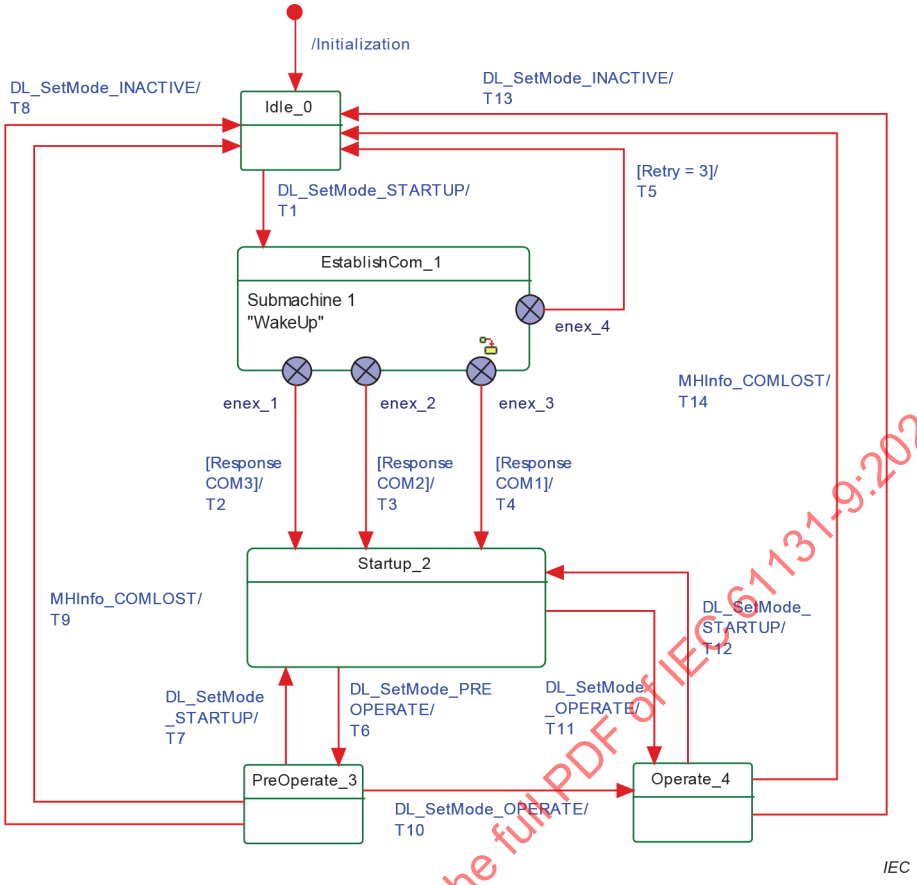


Figure 35 – State machine of the Master DL-mode handler

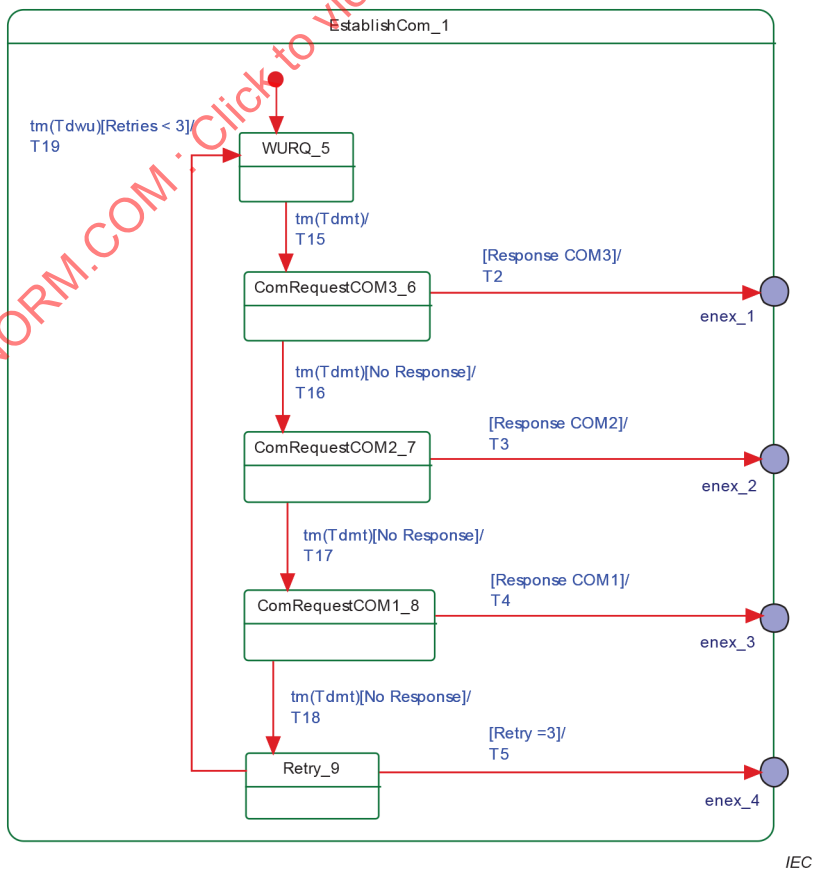


Figure 36 – Submachine 1 to establish communication

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 44 shows the state transition table of the Master DL-mode handler.

Table 44 – State transition table of the Master DL-mode handler

STATE NAME	STATE DESCRIPTION
Idle_0	Waiting on wakeup request from System Management (SM): DL_SetMode (STARTUP)
EstablishComm_1	Perform wake-up procedure (submachine 1)
Startup_2	System Management uses the STARTUP state for Device identification, check, and communication configuration (see Figure 71)
Preoperate_3	On-request Data exchange (parameter, commands, Events) without Process Data
Operate_4	Process Data and On-request Data exchange (parameter, commands, Events)
SM: WURQ_5	Create wake-up current pulse: Invoke service PL-Wake-Up (see Figure 12 and 5.3.3.3) and wait T_{DMT} (see Table 42).
SM: ComRequestCOM3_6	Try test message with transmission rate of COM3 via the message handler: Call MH_Conf_COMx (see Figure 40) and wait T_{DMT} (see Table 42).
SM: ComRequestCOM2_7	Try test message with transmission rate of COM2 via the message handler: Call MH_Conf_COMx (see Figure 40) and wait T_{DMT} (see Table 42).
SM: ComRequestCOM1_8	Try test message with transmission rate of COM1 via the message handler: Call MH_Conf_COMx (see Figure 40) and wait T_{DMT} (see Table 42).
SM: Retry_9	Check number of Retries

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	Set Retry = 0.
T2	1	2	Transmission rate of COM3 successful. Message handler activated and configured to COM3 (see Figure 40, Transition T2). Activate command handler (call CH_Conf_ACTIVE in Figure 53). Return DL_Mode.ind (STARTUP) and DL_Mode.ind (COM3) to SM.
T3	1	2	Transmission rate of COM2 successful. Message handler activated and configured to COM2 (see Figure 40, Transition T2). Activate command handler (call CH_Conf_ACTIVE in Figure 53). Return DL_Mode.ind (STARTUP) and DL_Mode.ind (COM2) to SM.
T4	1	2	Transmission rate of COM1 successful. Message handler activated and configured to COM1 (see Figure 40, Transition T2). Activate command handler (call CH_Conf_ACTIVE in Figure 53). Return DL_Mode.ind (STARTUP) and DL_Mode.ind (COM1) to SM.
T5	1	0	Return DL_Mode.ind (INACTIVE) to SM.
T6	2	3	SM requested the PREOPERATE state. Activate On-request Data (call OH_Conf_ACTIVE in Figure 48), ISDU (call IH_Conf_ACTIVE in Figure 51), and Event handler (call EH_Conf_ACTIVE in Figure 55). Change message handler state to PREOPERATE (call MH_Conf_PREOPERATE in Figure 40). Return DL_Mode.ind (PREOPERATE) to SM.
T7	3	2	SM requested the STARTUP state. Change message handler state to STARTUP (call MH_Conf_STARTUP in Figure 40). Deactivate On-request Data (call OH_Conf_INACTIVE in Figure 48), ISDU (call IH_Conf_INACTIVE in Figure 51), and Event handler (call EH_Conf_INACTIVE in Figure 55). Return DL_Mode.ind (STARTUP) to SM.
T8	3	0	SM requested the SIO mode. Deactivate all handlers (call xx_Conf_INACTIVE). Return DL_Mode.ind (INACTIVE) to SM. See 7.3.2.3.
T9	3	0	Message handler informs about lost communication via the DL-A service MHInfo (COMLOST). Deactivate all handlers (call xx_Conf_INACTIVE). Return DL_Mode.ind (COMLOST) to SM.
T10	3	4	SM requested the OPERATE state. Activate the Process Data handler (call PD_Conf_SINGLE if M-sequence type = TYPE_2_x, or PD_Conf_INTERLEAVE if M-sequence type = TYPE_1_1 in Figure 46). Change message handler state to OPERATE (call MH_Conf_OPERATE in Figure 40). Return DL_Mode.ind (OPERATE) to SM.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T11	2	4	SM requested the OPERATE state. Activate the Process Data handler (call PD_Conf_SINGLE or PD_Conf_INTERLEAVE in Figure 46 according to the Master Port configuration). Activate On-request Data (call OH_Conf_ACTIVE in Figure 48), ISDU (call IH_Conf_ACTIVE in Figure 51), and Event handler (call EH_Conf_ACTIVE in Figure 55). Change message handler state to OPERATE (call MH_Conf_OPERATE in Figure 40). Return DL_Mode.ind (OPERATE) to SM.
T12	4	2	SM requested the STARTUP state. Change message handler state to STARTUP (call MH_Conf_STARTUP in Figure 40). Deactivate Process Data (call PD_Conf_INACTIVE in Figure 46), On-request Data (call OH_Conf_INACTIVE in Figure 48), ISDU (call IH_Conf_INACTIVE in Figure 51), and Event handler (call EH_Conf_INACTIVE in Figure 55). Return DL_Mode.ind (STARTUP) to SM.
T13	4	0	SM requested the SIO state. Deactivate all handlers (call xx_Conf_INACTIVE). Return DL_Mode.ind (INACTIVE) to SM. See 7.3.2.3.
T14	4	0	Message handler informs about lost communication via the DL-A service MHInfo (COMLOST). Deactivate all handlers (call xx_Conf_INACTIVE). Return DL_Mode.ind (COMLOST) to SM.
T15	5	6	Set transmission rate of COM3 mode.
T16	6	7	Set transmission rate of COM2 mode.
T17	7	8	Set transmission rate of COM1 mode.
T18	8	9	Increment Retry
T19	9	5	-

INTERNAL ITEMS	TYPE	DEFINITION
MH_Conf_COMx	Call	This call causes the message handler to send a message with the requested transmission rate of COMx and with M-sequence TYPE_0 (see Table 46).
MH_Conf_STARTUP	Call	This call causes the message handler to switch to the STARTUP state (see Figure 40)
MH_Conf_PREOPERATE	Call	This call causes the message handler to switch to the PREOPERATE state (see Figure 40)
MH_Conf_OPERATE	Call	This call causes the message handler to switch to the OPERATE state (see Figure 40)
xx_Conf_ACTIVE	Call	These calls activate the respective handler. xx is substitute for MH (message handler), OH (On-request Data handler), IH (ISDU handler), CH (Command handler), and/or EH (Event handler)
xx_Conf_INACTIVE	Call	These calls deactivate the respective handler. xx is substitute for MH (message handler), OH (On-request Data handler), IH (ISDU handler), CH (Command handler), and/or EH (Eventhandler)
Retry	Variable	Number of retries to establish communication

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.2.5 State machine of the Device DL-mode handler

Figure 37 shows the state machine of the Device DL-mode handler.

In state PreOperate_3 and Operate_4 different sets of handlers within the Device are activated.

The Master uses MasterCommands (see Table 44) to change the Device to SIO, STARTUP, PREOPERATE, and OPERATE states.

Whenever the message handler detects illegal (unexpected) M-sequence types, it will cause the DL-mode handler to change to the STARTUP state and to indicate this state to its system management (see 9.3.3.2) for the purpose of synchronization of Master and Device.

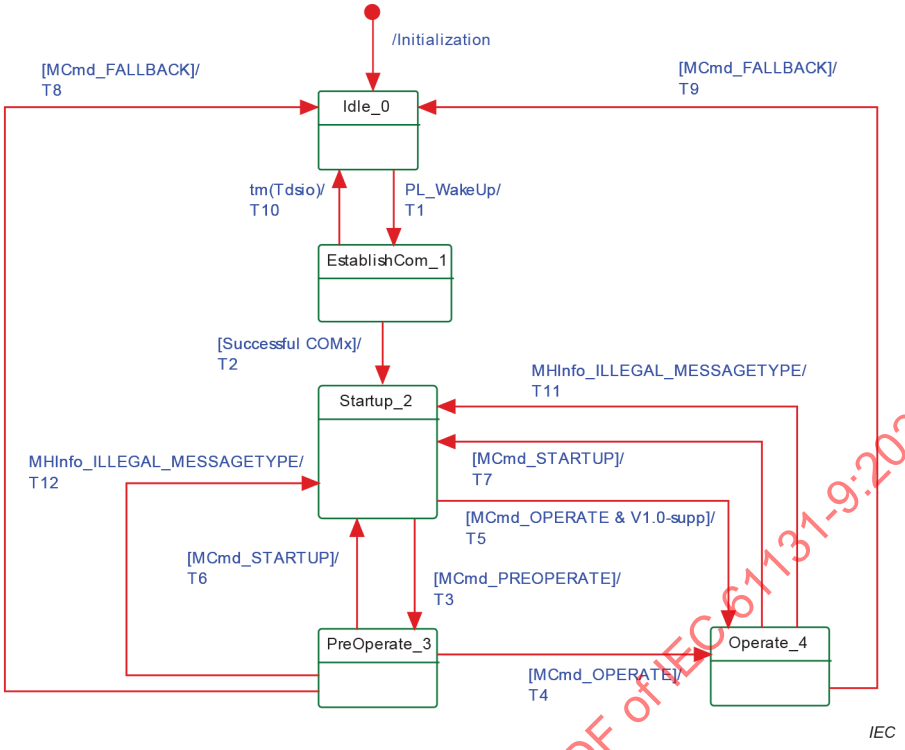


Figure 37 – State machine of the Device DL-mode handler

Table 45 shows the state transition table of the Device DL-mode handler.

Table 45 – State transition table of the Device DL-mode handler

STATE NAME	STATE DESCRIPTION
Idle_0	Waiting on a detected wakeup current pulse (PL_WakeUp.ind).
EstablishComm_1	Message handler activated and waiting for the COMx test messages (see Table 44)
Startup_2	Compatibility checks (see 9.2.3.3). Devices not supporting a Master according to [8] will remain in STARTUP thus supporting further identification but no process data exchange in this case.
Preoperate_3	On-request Data exchange (parameter, commands, Events) without Process Data
Operate_4	Process Data (PD) and On-request Data exchange (parameter, commands, Events)

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	Wake-up current pulse detected. Activate message handler (call MH_Conf_ACTIVE in Figure 44). Indicate state via service DL_Mode.ind (ESTABCOM) to SM.
T2	1	2	One out of the three transmission rates of COM3, COM2, or COM1 mode established. Activate On-request Data (call OH_Conf_ACTIVE in Figure 49) and command handler (call CH_Conf_ACTIVE in Figure 54). Indicate state via service DL_Mode.ind (COM1, COM2, or COM3) to SM.
T3	2	3	Device command handler received MasterCommand (MCmd_PREOPERATE). Activate ISDU (call IH_Conf_ACTIVE in Figure 52) and Event handler (call EH_Conf_ACTIVE in Figure 56). Indicate state via service DL_Mode.ind (PREOPERATE) to SM.
T4	3	4	Device command handler received MasterCommand (MCmd_OPERATE). Activate Process Data handler (call PD_Conf_ACTIVE in Figure 47). Indicate state via service DL_Mode.ind (OPERATE) to SM.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T5	2	4	Device command handler received MasterCommand (MCmd_OPERATE). Activate Process Data handler (call PD_Conf_ACTIVE in Figure 47), ISDU (call IH_Conf_ACTIVE in Figure 52), and Event handler (call EH_Conf_ACTIVE in Figure 56). Indicate state via service DL_Mode.ind (OPERATE) to SM.
T6	3	2	Device command handler received MasterCommand (MCmd_STARTUP). Deactivate ISDU (call IH_Conf_INACTIVE in Figure 52) and Event handler (call EH_Conf_INACTIVE in Figure 56). Indicate state via service DL_Mode.ind (STARTUP) to SM.
T7	4	2	Device command handler received MasterCommand (MCmd_STARTUP). Deactivate Process Data handler (call PD_Conf_INACTIVE in Figure 47), ISDU (call IH_Conf_INACTIVE in Figure 52), and Event handler (call EH_Conf_INACTIVE in Figure 56). Indicate state via service DL_Mode.ind (STARTUP) to SM.
T8	3	0	Device command handler received MasterCommand (MCmd_FALLBACK). Wait until T_{FBD} elapsed, and then deactivate all handlers (call xx_Conf_INACTIVE). Indicate state via service DL_Mode.ind (INACTIVE) to SM (see Figure 81 and Table 95).
T9	4	0	Device command handler received MasterCommand (MCmd_FALLBACK). Wait until T_{FBD} elapsed, and then deactivate all handlers (call xx_Conf_INACTIVE). Indicate state via service DL_Mode.ind (INACTIVE) to SM (see Figure 81 and Table 95).
T10	1	0	After unsuccessful wake-up procedures (see Figure 32) the Device establishes the configured SIO mode after an elapsed time T_{DSIO} (see Figure 33). Deactivate all handlers (call xx_Conf_INACTIVE). Indicate state via service DL_Mode.ind (INACTIVE) to SM.
T11	4	2	Message handler detected an illegal M-sequence type. Deactivate Process Data (call PD_Conf_INACTIVE in Figure 47), ISDU (call IH_Conf_INACTIVE in Figure 52), and Event handler (call EH_Conf_INACTIVE in Figure 56). Indicate state via service DL_Mode.ind (STARTUP) to SM (see Figure 81 and Table 95).
T12	3	2	Message handler detected an illegal M-sequence type. Deactivate ISDU (call IH_Conf_INACTIVE in Figure 52) and Event handler (call EH_Conf_INACTIVE in Figure 56). Indicate state via service DL_Mode.ind (STARTUP) to SM (see Figure 81 and Table 95).

INTERNAL ITEMS	TYPE	DEFINITION
T_{FBD}	Time	See Table 43
T_{DSIO}	Time	See Figure 33
MCmd_XXXXXXX	Call	Any MasterCommand received by the Device command handler (see Table 44 and Figure 54, state "CommandHandler_2")
V1.0-supp	Flag	Device supports V1.0 mode

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.3 Message handler

7.3.3.1 General

The role of the message handler is specified in 7.1 and 7.2.2.1. Subclause 7.3.3 specifies the structure and types of M-sequences and the behavior (dynamics) of the message handler.

7.3.3.2 M-sequences

A Master and its Device exchange data by means of a sequence of messages (M-sequence). An M-sequence comprises a message from the Master followed by a message from the Device as shown in Figure 38. Each message consists of UART frames.

All the multi-octet data types shall be transmitted as a big-endian sequence, i.e. the most significant octet (MSO) shall be sent first, followed by less significant octets in descending order, with the least significant octet (LSO) being sent last, as shown in Figure 2.

The Master message starts with the "M-sequence Control" (MC) octet, followed by the "CHECK/TYPE" (CKT) octet, and optionally followed by either "Process Data" (PD) and/or "On-request Data" (OD) octets. The Device message in turn starts optionally with "Process Data" (PD) octets and/or "On-request Data" (OD) octets, followed by the "CHECK/STAT" (CKS) octet.

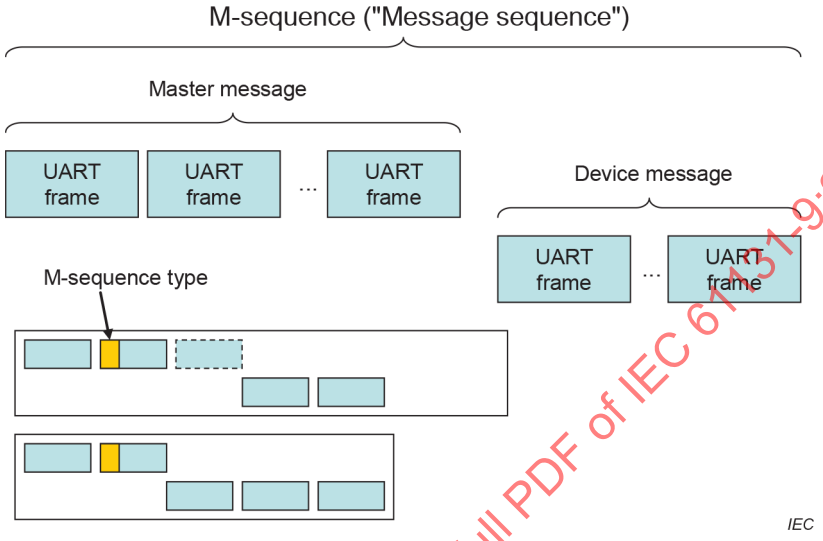


Figure 38 – SDCI message sequences

Various M-sequence types can be selected to meet the particular needs of an actuator or sensor (scan rate, amount of Process Data). The length of Master and Device messages can vary depending on the type of messages and the data transmission direction, see Figure 38.

Figure 39 presents an overview of the defined M-sequence types. Parts within dotted lines depend on the read or write direction within the M-sequence control octet.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

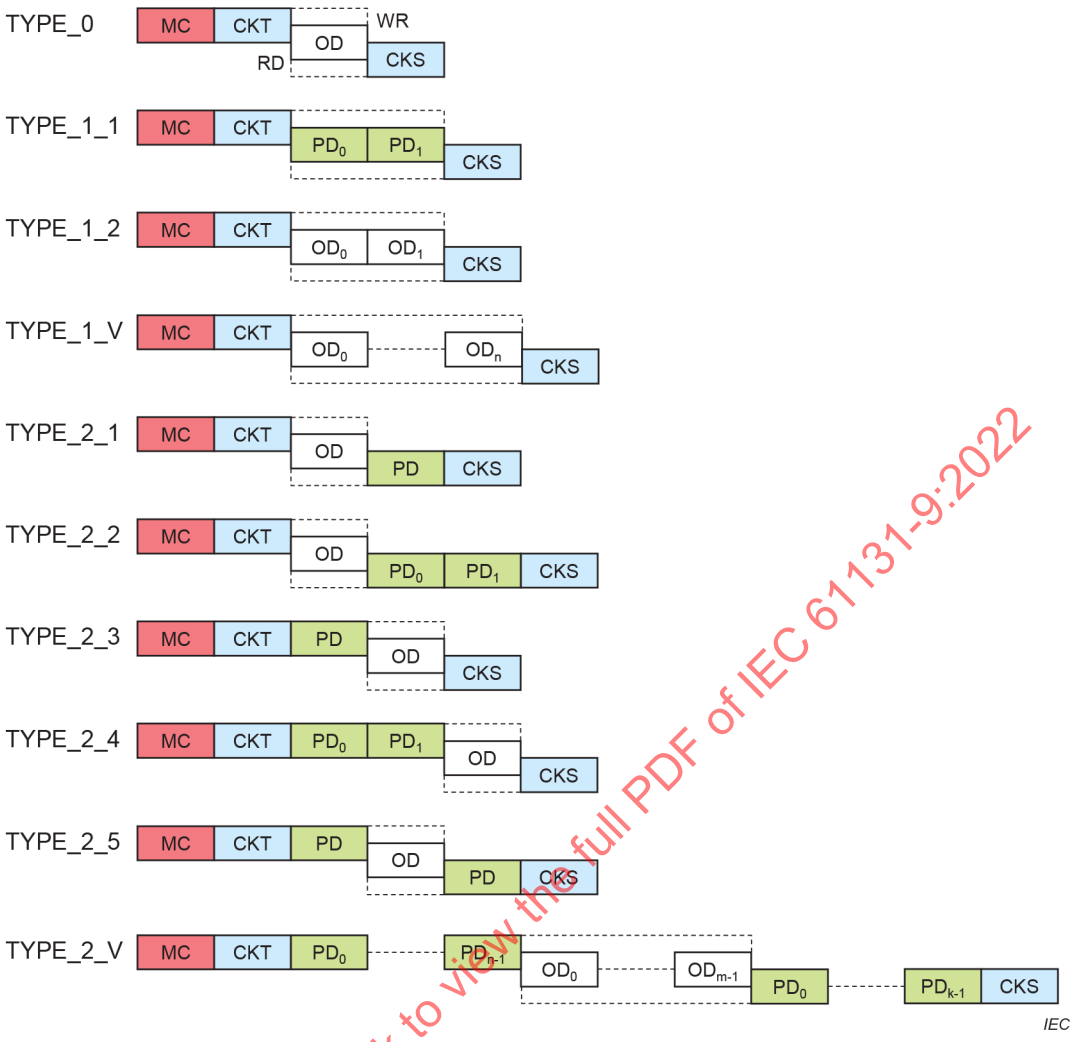


Figure 39 – Overview of M-sequence types

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The fixed M-sequence types consist of TYPE_0, TYPE_1_1, TYPE_1_2, and TYPE_2_1 through TYPE_2_5. Caution: The former TYPE_2_6 is no longer supported. The variable M-sequence types consist of TYPE_1_V and TYPE_2_V.

The different M-sequence types meet the various requirements of sensors and actuators regarding their Process Data width and respective conditions. See Clause A.2 for details of M-sequence types. See Clause A.3 for the timing constraints with M-sequences.

7.3.3.3 MasterCycleTime constraints

Within states STARTUP and PREOPERATE a Device is able to communicate in an acyclic manner. In order to detect the disconnecting of Devices it is highly recommended for the Master to perform from this point a periodic communication ("keep-alive message") via acyclic M-sequences through the data link layer. The minimum recovery times for acyclic communication specified in A.2.6 shall be considered.

After these phases, cyclic Process Data communication can be started by the Master via the DL_SetMode (OPERATE) service. M-sequence types for the cyclic data exchange shall be used in this communication phase to exchange Process Data (PD) and On-request Data with a Device (see Table A.9 and Table A.10).

The Master shall use for time t_{CYC} the value indicated in the Device parameter "MasterCycleTime" (see Table B.1) with a relative tolerance of −1 % to +10 % (including jitter).

In cases where a Device has to be switched back to SIO mode after parameterization, the Master shall send a command "Fallback" (see Table B.2), which is followed by a confirmation from the Device.

7.3.3.4 State machine of the Master message handler

Figure 40 shows the Master state machine of the Master message handler. Three submachines describing reactions on communication errors are shown in Figure 41, Figure 42, and Figure 43.

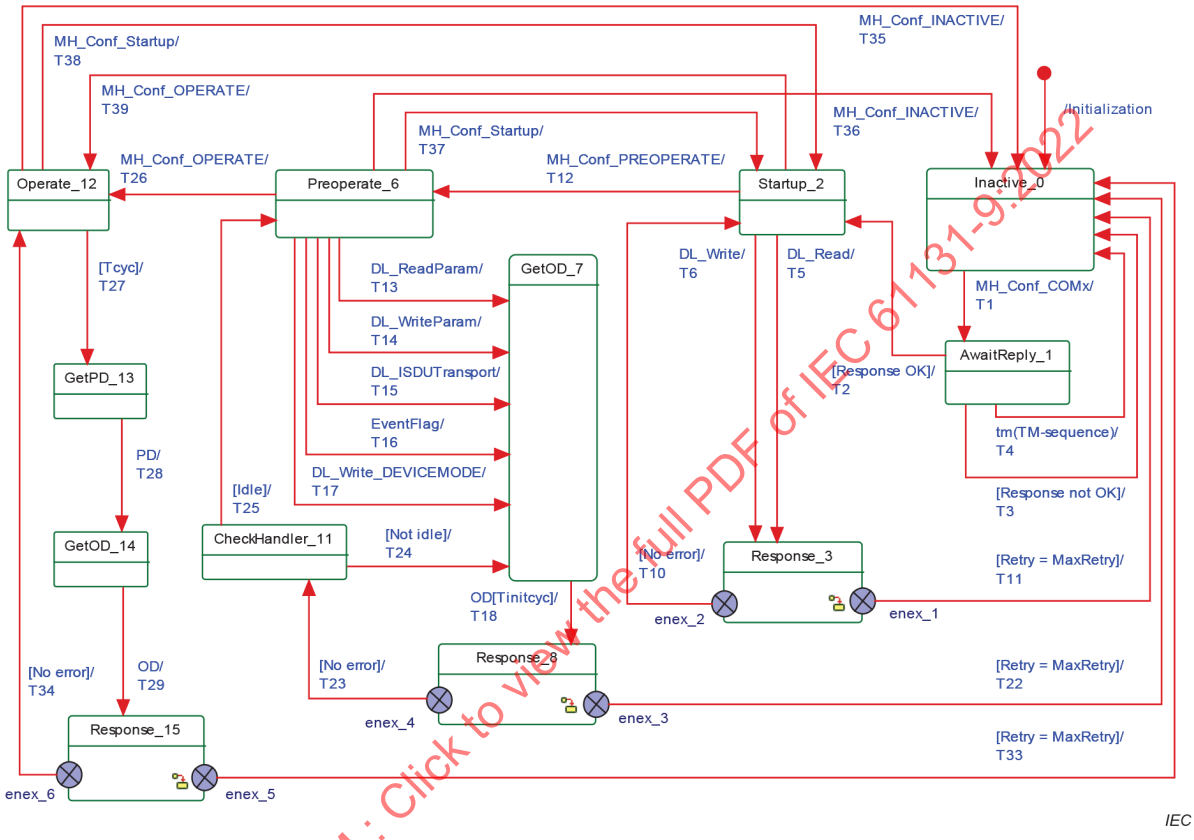


Figure 40 – State machine of the Master message handler

The message handler takes care of the special communication requirements within the states "EstablishCom", "Startup", "PreOperate", and "Operate" of the DL-Mode handler. An internal administrative call MH_Conf_COMx in state "Inactive_0" causes the message handler to send "test" messages with M-sequence TYPE_0 and different transmission rates of COM3, COM2, or COM1 during the establish communication sequence.

The state "Startup_2" provides all the communication means to support the identity checks of System Management with the help of DL_Read and DL_Write services. The message handler waits on the occurrence of these services to send and receive messages (acyclic communication). The state "Preoperate_6" is the checkpoint for all On-request Data activities such as ISDUs, commands, and Events for parameterization of the Device. The message handler waits on the occurrence of the services shown in Figure 40 to send and receive messages (acyclic communication). The state "Operate_12" is the checkpoint for cyclic Process Data exchange. Depending on the M-sequence type the message handler generates Master messages with Process Data acquired from the Process Data handler via the PD service and optionally On-request Data acquired from the On-request Data handler via the OD service.

Figure 41 shows the submachine of state "Response 3".

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

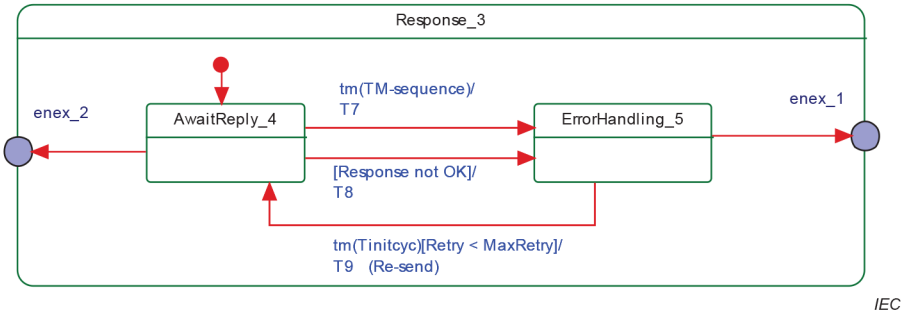


Figure 41 – Submachine "Response 3" of the message handler

Figure 42 shows the submachine of state "Response 8".

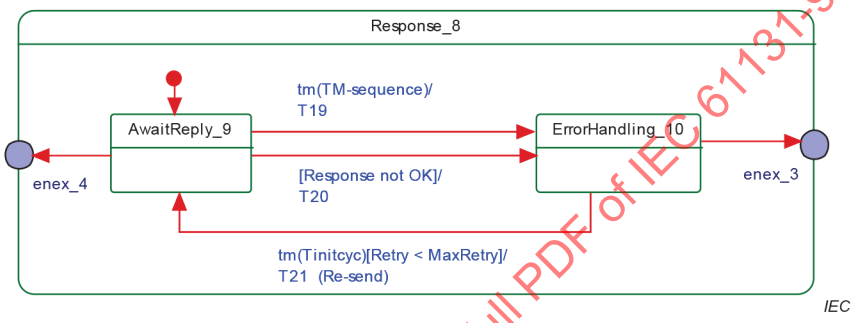


Figure 42 – Submachine "Response 8" of the message handler

Figure 43 shows the submachine of state "Response 15".

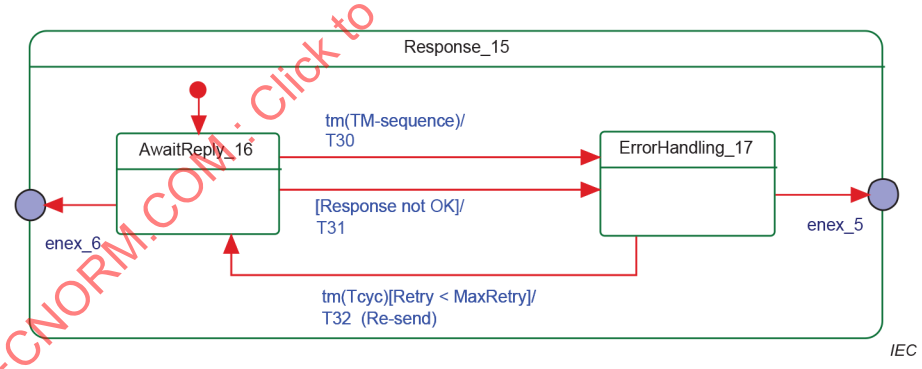


Figure 43 – Submachine "Response 15" of the message handler

Table 46 shows the state transition table of the Master message handler.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 46 – State transition table of the Master message handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting on demand for a "test" message via MH_Conf_COMx call (see Figure 36 and Table 44) from DL-mode handler.
AwaitReply_1	Waiting on response from the Device to the "test" message. Return to Inactive_0 state whenever the time $T_{\text{M-sequence}}$ elapsed without response from the Device or the response to the "test" message could not be decoded. In case of a correct response from the Device, the message handler changes to the Startup_2 state.
Startup_2	When entered via transition T2, this state is responsible to control acyclic On-request Data exchange according to conditions specified in Table A.7. Any service DL_Write or DL_Read from System Management causes a transition.
Response_3	The OD service caused the message handler to send a corresponding message. The submachine in this pseudo state waits on the response and checks its correctness.
SM: AwaitReply_4	This state checks whether the time $T_{\text{M-sequence}}$ elapsed and the response is correct.
SM: ErrorHandling_5	In case of an incorrect response the message handler will re-send the message after a waiting time T_{initcyc}. After too many retries the message handler will change to the Inactive_0 state.
Preoperate_6	Upon reception of a call MH_Conf_PREOPERATE the message handler changed to this state. The message handler is now responsible to control acyclic On-request Data exchange according to conditions specified in Table A.8. Any service DL_ReadParam, DL_WriteParam, DL_ISDUTransport, DL_Write, or EventFlag causes a transition.
GetOD_7	The message handler used the ODTrig service to aquire OD from the On-request Data handler. The message handler waits on the OD service to send a message after a time T_{initcyc}.
Response_8	The OD service caused the message handler to send a corresponding message. The submachine in this pseudo state waits on the response and checks its correctness.
SM: AwaitReply_9	This state checks whether the time $T_{\text{M-sequence}}$ elapsed and the response is correct.
SM: ErrorHandling_10	In case of an incorrect response the message handler will re-send the message after a waiting time T_{initcyc}. After too many retries the message handler will change to the Inactive_0 state.
CheckHandler_11	Some services require several OD acquisition cycles to exchange the OD. Whenever the affected OD, ISDU, or Event handler returned to the idle state, the message handler can leave the OD acquisition loop.
Operate_12	Upon reception of a call MH_Conf_OPERATE the message handler changed to this state and after an initial time T_{initcyc}, it is responsible to control cyclic Process Data and On-request Data exchange according to conditions specified in Table A.9 and Table A.10. The message handler restarts on its own a new message cycle after the time t_{CYC} elapsed.
GetPD_13	The message handler used the PDTrig service to aquire PD from the Process Data handler. The message handler waits on the PD service and then changes to state GetOD_14.
GetOD_14	The message handler used the ODTrig service to aquire OD from the On-request Data handler. The message handler waits on the OD service to complement the already acquired PD and to send a message with the acquired PD/OD.
Response_15	The message handler sent a message with the acquired PD/OD. The submachine in this pseudo state waits on the response and checks its correctness.
SM: AwaitReply_16	This state checks whether the time $T_{\text{M-sequence}}$ elapsed and the response is correct.
SM: ErrorHandling_17	In case of an incorrect response the message handler will re-send the message after a waiting time t_{CYC}. After too many retries the message handler will change to the Inactive_0 state.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	Send a message with the requested transmission rate of COMx and with M-sequence TYPE_0: Read Direct Parameter page 1, address 0x02 ("MinCycleTime"), compiling into an M-sequence control MC = 0xA2 (see A.1.2). Start timer with $T_{\text{M-sequence}}$.
T2	1	2	Return value of "MinCycleTime" via DL_Read service confirmation.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T3	1	0	Reset timer ($T_{M\text{-}sequence}$).
T4	1	0	Reset timer ($T_{M\text{-}sequence}$).
T5	2	3	Send message using the established transmission rate, the page communication channel, and the read access option (see A.1.2). Start timer with $T_{M\text{-}sequence}$.
T6	2	3	Send message using the established transmission rate, the page communication channel, and the write access option (see A.1.2). Start timer with $T_{M\text{-}sequence}$.
T7	4	5	Reset timer ($T_{M\text{-}sequence}$).
T8	4	5	Reset timer ($T_{M\text{-}sequence}$).
T9	5	4	Re-send message after a time $T_{initcyc}$. Restart timer with $T_{M\text{-}sequence}$.
T10	3	2	Return DL_Read or DL_Write service confirmation respectively to System Management.
T11	3	0	Message handler returns MH_Info (COMLOST) to DL-mode handler.
T12	2	6	-
T13	6	7	The Message handler invokes the ODTrig service for the On-request handler (see Figure 48), which is in state "ISDU_1". In this state it causes the ISDU handler to provide the OD service in correspondence to the DL_ReadParam service (see Figure 51, Transition T13).
T14	6	7	The Message handler invokes the ODTrig service for the On-request handler (see Figure 48), which is in state "ISDU_1". In this state it causes the ISDU handler to provide the OD service in correspondence to the DL_WriteParam service (see Figure 51, Transition T13).
T15	6	7	The Message handler invokes the ODTrig service for the On-request handler (see Figure 48), which is in state "ISDU_1". In this state it causes the ISDU handler to provide the OD service in correspondence to the DL_ISDUTransort service (see Figure 51, Transition T2). The message handler can need several cycles until the ISDU handler returns to the "idle" state.
T16	6	7	The Message handler invokes the ODTrig service for the On-request handler (see Figure 48), which is in state "Event_3". In this state it causes the Event handler to provide the OD service in correspondence to the EventFlag service (see Figure 55, Transition T2). The message handler can need several cycles until the Event handler returns to the "idle" state.
T17	6	7	The Message handler invokes the ODTrig service for the On-request handler (see Figure 48), which is in state "ISDU_1". In this state it causes the ISDU handler to provide the OD service in correspondence to the DL_Write service (see Figure 51, Transition T13).
T18	7	8	Send message after a recovery time $T_{initcyc}$ caused by the OD.req service. Start timer with $T_{M\text{-}sequence}$.
T19	9	10	Reset timer ($T_{M\text{-}sequence}$).
T20	9	10	Reset timer ($T_{M\text{-}sequence}$).
T21	10	9	Re-send message after a time $T_{initcyc}$. Restart timer with $T_{M\text{-}sequence}$.
T22	8	0	Message handler changes to state Inactive_0 and returns MH_Info (COMLOST) to DL-mode handler.
T23	8	11	-
T24	11	7	Acquire OD through invocation of the ODTrig service to the On-request Data handler, which in turn triggers the current handler in charge via the ISDU or EventTrig call.
T25	11	6	Return result via service primitive OD.cnf
T26	6	12	Message handler changes to state Operate_12.
T27	12	13	Start the t_{CYC} -timer. Acquire PD through invocation of the PDTrig service to the Process Data handler (see Figure 46).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T28	13	14	Acquire OD through invocation of the ODTrig service to the On-request Data handler (see Figure 48).
T29	14	15	PD and OD ready through PD.req service from PD handler and OD.req service via the OD handler. Message handler sends message. Start timer with $T_{M\text{-}sequence}$.
T30	16	17	Reset timer ($T_{M\text{-}sequence}$).
T31	16	17	Reset timer ($T_{M\text{-}sequence}$).
T32	17	16	Re-send message after a time t_{CYC}. Restart timer with $T_{M\text{-}sequence}$.
T33	15	0	Message handler changes to state Inactive_0 and returns MH_Info (COMLOST) to DL-mode handler.
T34	15	12	Device response message is correct. Return PD via service PD.cnf and via call PDTrig to the PD handler (see Table 48). Return OD via service OD.cnf and via call ODTrig to the On-request Data handler, which redirects it to the ISDU (see Table 53), Command (see Table 56), or Event handler (see Table 59) in charge.
T35	12	0	Message handler changes to state Inactive_0 and returns MH_Info (COMLOST) to the DL-mode handler.
T36	6	0	Message handler changes to state Inactive_0 and returns MH_Info (COMLOST) to the DL-mode handler.
T37	6	2	-
T38	12	2	-
T39	2	12	-

INTERNAL ITEMS	TYPE	DEFINITION
Retry	Variable	Retry counter
MaxRetry	Constant	MaxRetry = 2, see Table 102
$t_{M\text{-}sequence}$	Time	See Equation (A.6)
t_{CYC}	Time	The DL_SetMode service provides this value with its parameter "M-sequenceTime". See Equation (A.7)
$t_{initcyc}$	Time	See A.2.6
MH_Conf_xxx	Call	See Table 44

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.3.5 State machine of the Device message handler

Figure 44 shows the state machine of the Device message handler.

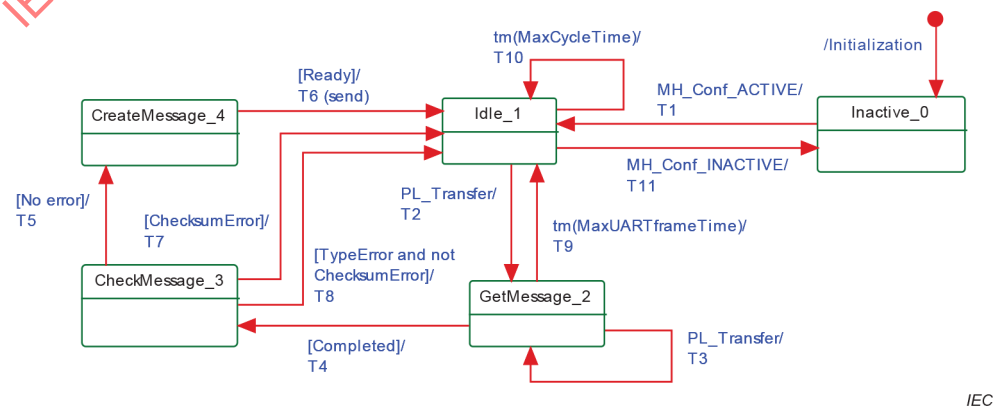


Figure 44 – State machine of the Device message handler

Table 47 shows the state transition table of the Device message handler.

Table 47 – State transition table of the Device message handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting for activation by the Device DL-mode handler through MH_Conf_ACTIVE (see Table 45, Transition T1).
Idle_1	Waiting on first UART frame of the Master message through PL_Transfer service indication. Check whether time "MaxCycleTime" elapsed.
GetMessage_2	Receive a Master message UART frame. Check number of received UART frames (Device detects M-sequence type by means of the first two received octets depending on the current communication state and thus knows the number of the UART frames). Check whether the time "MaxUARTframeTime" elapsed.
CheckMessage_3	Check M-sequence type and checksum of received message.
CreateMessage_4	Compile message from OD.rsp, PD.rsp, EventFlag, and PDStatus services.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	2	Start "MaxUARTframeTime" and "MaxCycleTime" when in OPERATE.
T3	2	2	Restart timer "MaxUARTframeTime".
T4	2	3	Reset timer "MaxUARTframeTime".
T5	3	4	Invoke OD.ind and PD.ind service indications
T6	4	1	Compile and invoke PL_Transfer.rsp service response (Device sends response message)
T7	3	1	–
T8	3	1	Indicate error to DL-mode handler via MHInfo (ILLEGAL_MESSAGETYPE)
T9	2	1	Reset both timers "MaxUARTframeTime" and "MaxCycleTime".
T10	1	1	Indicate error to actuator technology that shall observe this information and take corresponding actions (see 10.2 and 10.8.3).
T11	1	0	Device message handler changes state to Inactive_0.

INTERNAL ITEMS	TYPE	DEFINITION
MaxUARTFrameTime	Time	Time for the transmission of a UART frame (11 T_{BIT}) plus maximum of t_1 (1 T_{BIT}) = 11 T_{BIT}.
MaxCycleTime	Time	The purpose of the timer "MaxCycleTime" is to check whether cyclic Process Data exchange took too much time or has been interrupted. MaxCycleTime shall be > MasterCycleTime (see A.3.7).
TypeError	Guard	One of the possible errors detected: ILLEGAL_MESSAGETYPE, or COMLOST
ChecksumError	Guard	Checksum error of message detected

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.4 Process Data handler

7.3.4.1 General

The transport of output Process Data is performed using the DL_OutputUpdate services and for input Process Data using the DL_InputTransport services (see Figure 28). A Process Data cycle is completed when the entire set of Process Data has been transferred between Master and Device in the requested direction. Such a cycle can last for more than one M-sequence.

All Process Data are transmitted within one M-sequence when using M-sequences of TYPE_2_x (see Figure 39). In this case the execution time of a Process Data cycle is equal to the cycle time t_{CYC}.

7.3.4.2 Interleave mode

All Process Data and On-request Data are transmitted in this case with multiple alternating M-sequences TYPE_1_1 (Process Data) and TYPE_1_2 (On-request Data) as shown in Figure 45. It demonstrates the Master messages writing output Process Data to a Device. The service parameter PDOutAddress indicates the partition of the output PD to be transmitted (see 7.2.2.3). For input Process Data the service parameter PDInAddress correspondingly indicates the partition of the input PD. Within a Process Data cycle all input PD shall be read first followed by all output PD to be written. A Process Data cycle comprises all cycle times required to transmit the complete Process Data.

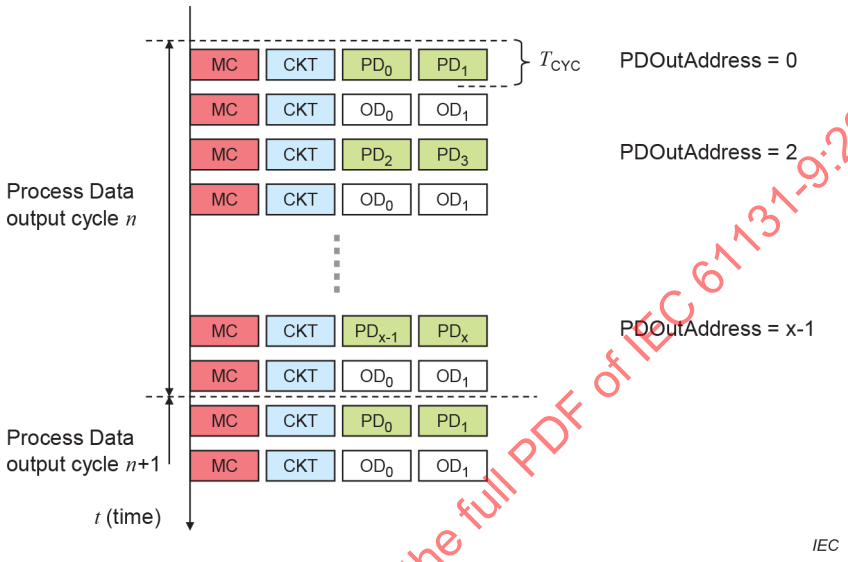


Figure 45 – Interleave mode for the segmented transmission of Process Data

Interleave mode is for legacy Devices only.

7.3.4.3 State machine of the Master Process Data handler

Figure 46 shows the state machine of the Master Process Data handler.

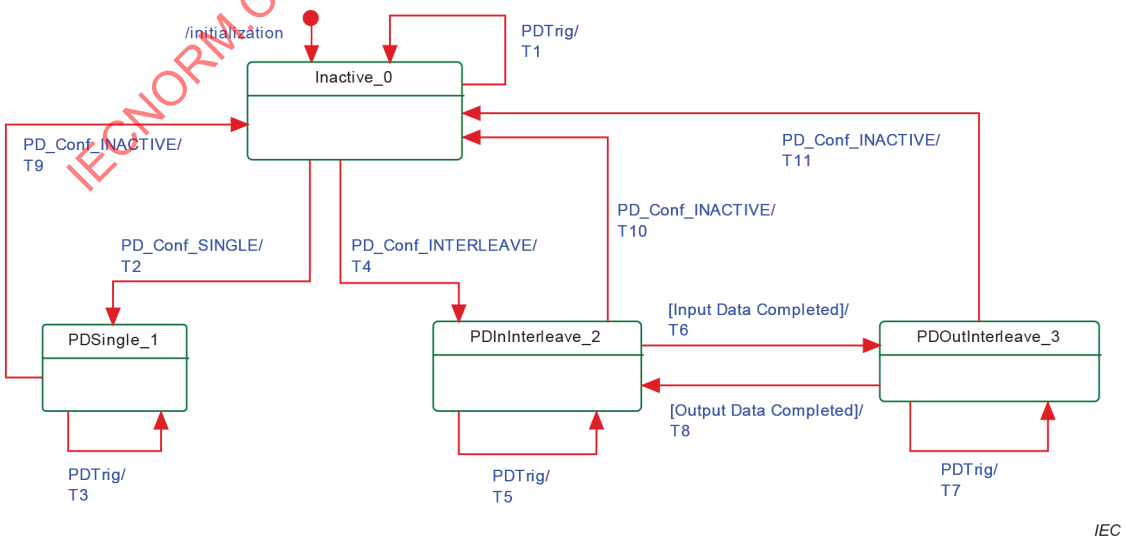


Figure 46 – State machine of the Master Process Data handler

Table 48 shows the state transition table of the Master Process Data handler.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 48 – State transition table of the Master Process Data handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting for activation
PDSingle_1	Process Data communication within one single M-sequence
PDInInterleave_2	Input Process Data communication in interleave mode
PDOutInterleave_3	Output Process Data communication in interleave mode

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	0	Invoke PD.req with no Process Data
T2	0	1	NOTE The DL-mode handler configured the Process Data handler for single PD transmission (see Table 44, T10 or T11).
T3	1	1	Take data from DL_PDOutputUpdate service and invoke PD.req to propagate output PD to the message handler. Take data from PD.cnf and invoke DL_PDInputTransport.ind and DL_PDCycle.ind to propagate input PD to the AL.
T4	0	2	NOTE Configured for interleave PD transmission (see Table 44, T10 or T11).
T5	2	2	Invoke PD.req and use PD.cnf to prepare DL_PDInputTransport.ind.
T6	2	3	Invoke DL_PDInputTransport.ind and DL_PDCycle.ind to propagate input PD to the AL (see 7.2.1.11).
T7	3	3	Take data from DL_PDOutputUpdate service and invoke PD.req to propagate output PD to the message handler.
T8	3	2	Invoke DL_PDCycle.ind to indicate end of Process Data cycle to the AL (see 7.2.1.12).
T9	1	0	-
T10	2	0	-
T11	3	0	-

INTERNAL ITEMS	TYPE	DEFINITION
<None>		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.4.4 State machine of the Device Process Data handler

Figure 47 shows the state machine of the Device Process Data handler.

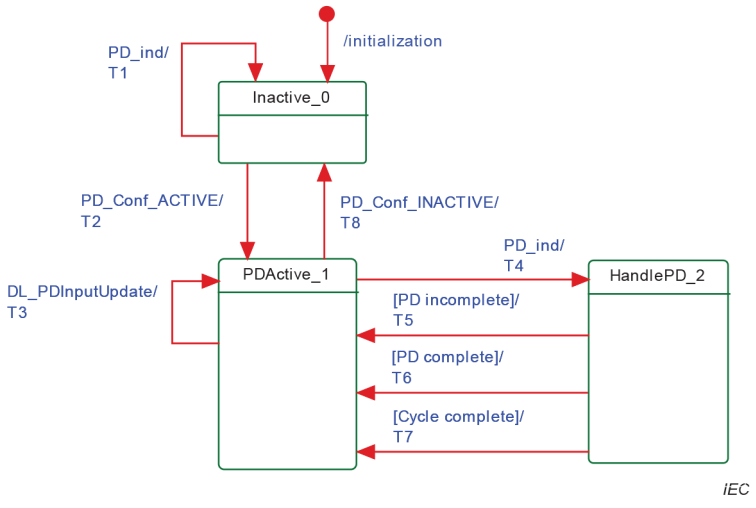


Figure 47 – State machine of the Device Process Data handler

See sequence diagrams in Figure 67 and Figure 68 for context.

Table 49 shows the state transition table of the Device Process Data handler

Table 49 – State transition table of the Device Process Data handler

STATE NAME		STATE DESCRIPTION	
Inactive_0		Waiting on activation	
PDActive_1		Handler active and waiting on next message handler demand via PD service or DL_PDInputUpdate service from AL.	
HandlePD_2		Check Process Data for completeness in interleave mode	
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T1	0	0	Ignore Process Data
T2	0	1	-
T3	1	1	Prepare input Process Data for PD.rsp for next message handler demand
T4	1	2	Message handler demands input PD via a PD.ind service and delivers output PD or segment of output PD. Invoke PD.rsp with input Process Data when in non-interleave mode (see 7.2.2.3).
T5	2	1	-
T6	2	1	Invoke DL_PDOutputTransport.ind (see 7.2.1.9)
T7	2	1	Invoke DL_PDCycle.ind (see 7.2.1.12)
T8	1	0	-
INTERNAL ITEMS		**TYPE**	**DEFINITION**
PD_ind		Label	Invocation of service PD.ind occurred from message handler

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.5 On-request Data handler

7.3.5.1 General

The Master On-request Data handler is a subordinate state machine active in the "Startup_2", "PreOperate_3", and "Operate_4" state of the DL-mode handler (see Figure 35). It controls three other state machines, the so-called ISDU handler, the command handler, and the Event handler. It always starts with the ISDU handler by default.

Whenever an EventFlag.ind is received, the state machine will change to the Event handler. After the complete readout of the Event information, it will return to the ISDU handler state.

Whenever a DL_Control.req or PDInStatus.ind service is received while in the ISDU handler or in the Event handler, the state machine will change to the command handler. Once the command has been served, the state machine will return to the previously active state (ISDU or Event).

7.3.5.2 State machine of the Master On-request Data handler

Figure 48 shows the Master state machine of the On-request Data handler.

The On-request Data handler redirects the ODTrig.ind service primitive for the next message content to the currently active subsidiary handler (ISDU, command, or Event). This is performed through one of the ISDUTrig, CommandTrig, or EventTrig calls.

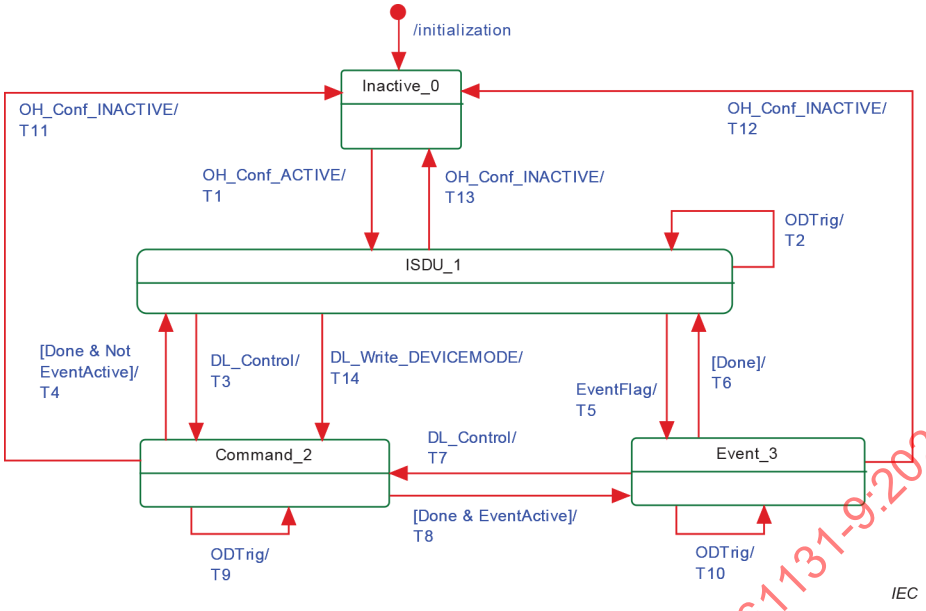


Figure 48 – State machine of the Master On-request Data handler

Table 50 shows the state transition table of the Master On-request Data handler.

Table 50 – State transition table of the Master On-request Data handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting on activation
ISDU_1	Default state of the On-request Data handler (lowest priority)
Command_2	State to control the Device via commands with highest priority
Event_3	State to convey Event information (errors, warnings, notifications) with higher priority

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	1	On-request Data handler propagates the ODTrig.ind service now named ISDUTrig to the ISDU handler (see Figure 51). In case of DL_Read, DL_Write, DL_ReadParam, or DL_WriteParam services, the ISDU handler will use a separate transition (see Figure 51, T13).
T3	1	2	–
T4	2	1	–
T5	1	3	EventActive = TRUE
T6	3	1	EventActive = FALSE
T7	3	2	–
T8	2	3	–
T9	2	2	On-request Data handler propagates the ODTrig.ind service now named CommandTrig to the command handler (see Figure 53)
T10	3	3	On-request Data handler propagates the ODTrig.ind service now named EventTrig to the Event handler (see Figure 55)
T11	2	0	–
T12	3	0	–
T13	1	0	–
T14	1	2	–

INTERNAL ITEMS	TYPE	DEFINITION
EventActive	Bool	Flag to indicate return direction after interruption of Event processing by a high priority command request

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.5.3 State machine of the Device On-request Data handler

Figure 49 shows the state machine of the Device On-request Data handler.

The Device On-request Data handler obtains information on the communication channel and the parameter or FlowCTRL address via the OD.ind service. The communication channels are totally independent. In case of a valid access, the corresponding ISDU, command or Event state machine is addressed via the associated communication channel.

The Device shall respond to read requests to not implemented address ranges with the value "0". It shall ignore write requests to not implemented address ranges.

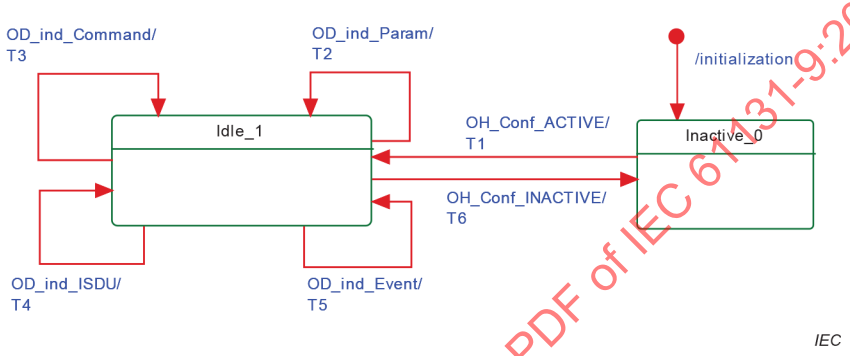


Figure 49 – State machine of the Device On-request Data handler

In case of an ISDU access in a Device without ISDU support, the Device shall respond with "No Service" (see Table A.12). An error message is not created.

NOTE OD.ind (R, ISDU, FlowCTRL = IDLE) is the default message if there are no On-request Data pending for transmission.

Table 51 shows the state transition table of the Device On-request Data handler.

Table 51 – State transition table of the Device On-request Data handler

STATE NAME		STATE DESCRIPTION	
Inactive_0		Waiting on activation	
Idle_1		Waiting on messages with On-request Data via service OD indication. Decomposition and analysis.	
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T1	0	1	–
T2	1	1	Provide data content of requested parameter or perform appropriate write action
T3	1	1	Redirect to command handler
T4	1	1	Redirect to ISDU handler
T5	1	1	Redirect to Event handler
T6	1	0	–
INTERNAL ITEMS		**TYPE**	**DEFINITION**
OD_ind_Param		Service	Alias for Service OD.ind (R/W, PAGE, 1 to 31, Data) in case of DL_ReadParam or DL_WriteParam
OD_ind_Command		Service	Alias for Service OD.ind (W, PAGE, 0, MasterCommand)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

INTERNAL ITEMS	TYPE	DEFINITION
OD_ind_ISDU	Service	Alias for Service OD.ind (R/W, ISDU, FlowCtrl, Data)
OD_ind_Event	Service	Alias for Service OD.ind (R/W, DIAGNOSIS, n, Data)

7.3.6 ISDU handler

7.3.6.1 Indexed Service Data Unit (ISDU)

The general structure of an ISDU is demonstrated in Figure 50 and specified in detail in Clause A.5.

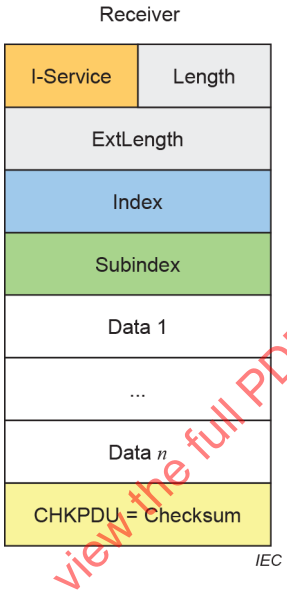


Figure 50 – Structure of the ISDU

The sequence of the elements corresponds to the transmission sequence. The elements of an ISDU can take various forms depending on the type of I-Service (see A.5.2 and Table A.12).

The ISDU allows accessing data objects (parameters and commands) to be transmitted (see Figure 6). The data objects shall be addressed by the “Index” element.

All multi-octet data types shall be transmitted as a big-endian sequence, i.e. the most significant octet (MSO) shall be sent first, followed by less significant octets in descending order, with the least significant octet (LSO) being sent last, as shown in Figure 2.

7.3.6.2 Transmission of ISDUs

An ISDU is transmitted via the ISDU communication channel (see Figure 8 and A.1.2). A number of messages are typically required to perform this transmission (segmentation). The Master transfers an ISDU by sending an I-Service (Read/Write) request to the Device via the ISDU communication channel. It then receives the Device’s response via the same channel.

In the ISDU communication channel, the "Address" element within the M-sequence control octet accommodates a counter (= FlowCTRL). FlowCTRL controls the segmented data flow (see A.1.2) by counting the M-sequences necessary to transmit an ISDU.

The receiver of an ISDU expects a FlowCTRL + 1 in the next message in case of undisturbed communication. If FlowCTRL is unchanged, the previously transmitted message is repeated. In any other case the ISDU structure is violated.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The Master uses the "Length" element of the ISDU and FlowCTRL to check the accomplishment of the complete transmission.

Permissible values for FlowCTRL are specified in Table 52.

Table 52 – FlowCTRL definitions

FlowCTRL	Definition
0x00 to 0x0F	COUNT M-sequence counter within an ISDU. Increments beginning with 1 after an ISDU START. Jumps back from 15 to 0 in the Event of an overflow.
0x10	START Start of an ISDU I-Service, i.e., start of a request or a response. For the start of a request, any previously incomplete services can be rejected. For a start request associated with a response, a Device shall send “No Service” until its application returns response data (see Table A.12).
0x11	IDLE 1 No request for ISDU transmission.
0x12	IDLE 2: Reserved for future use No request for ISDU transmission.
0x13 to 0x1E	Reserved
0x1F	ABORT Abort entire service. The Master responds by rejecting received response data. The Device responds by rejecting received request data and can generate an abort.

In state Idle_1, values 0x12 to 0x1F shall not lead to a communication error.

7.3.6.3 State machine of the Master ISDU handler

Figure 51 shows the state machine of the Master ISDU handler.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

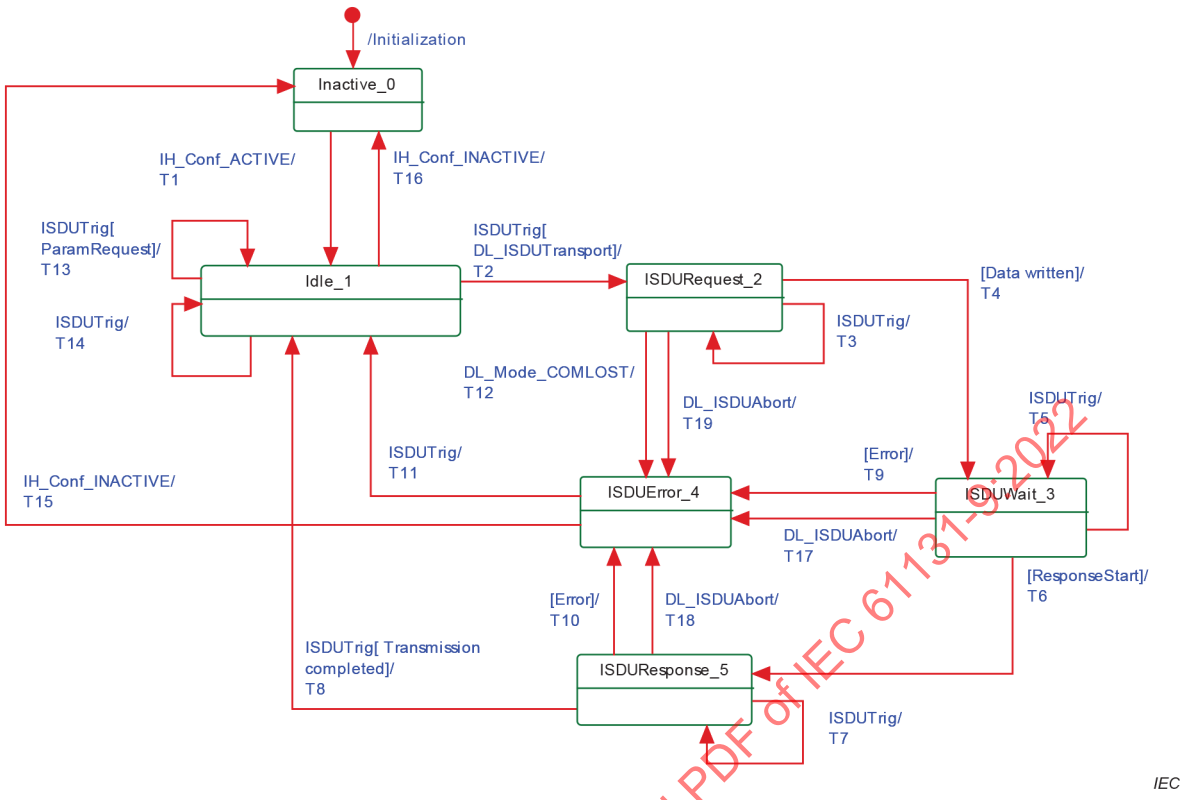


Figure 51 – State machine of the Master ISDU handler

Table 53 shows the state transition table of the Master ISDU handler.

Table 53 – State transition table of the Master ISDU handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting on activation
Idle_1	Waiting on transmission of next On-request Data
ISDURequest_2	Transmission of ISDU request data
ISDUWait_3	Waiting on response from Device. Observe ISDUTime
ISDUError_4	Error handling after detected errors: Invoke negative DL_ISDU_Transport response with ISDUTransportErrorInfo
ISDUResponse_5	Get response data from Device

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	2	Invoke OD.req with ISDU write start condition: OD.req (W, ISDU, flowCtrl = START, data)
T3	2	2	Invoke OD.req with ISDU data write and FlowCTRL under conditions of Table 52
T4	2	3	Start timer (ISDUTime)
T5	3	3	Invoke OD.req with ISDU read start condition: OD.req (R, ISDU, flowCtrl = START)
T6	3	5	Stop timer (ISDUTime)
T7	5	5	Invoke OD.req with ISDU data read and FlowCTRL under conditions of Table 52
T8	5	1	OD.req (R, ISDU, flowCtrl = IDLE) Invoke positive DL_ISDUTransport confirmation

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T9	3	4	–
T10	5	4	–
T11	4	1	Invoke OD.req with ISDU abortion: OD.req (R, ISDU, flowCtrl = ABORT). Invoke negative DL_ISDUTransport confirmation
T12	2	4	–
T13	1	1	Invoke OD.req with appropriate data. Invoke positive DL_ReadParam/DL_WriteParam confirmation
T14	1	1	Invoke OD.req with idle message: OD.req (R, ISDU, flowCtrl = IDLE)
T15	4	1	In case of lost communication, the message handler informs the DL_Mode handler which in turn uses the administrative call IH_Conf_INACTIVE. No actions during this transition required.
T16	1	0	–
T17	3	4	–
T18	5	4	–
T19	2	4	–

INTERNAL ITEMS	TYPE	DEFINITION
ISDUTime	Time	Measurement of Device response time (watchdog, see Table 102)
ResponseStart	Service	OD.cnf without "busy" indication (see Table A.14)
ParamRequest	Service	DL_ReadParam or DL_WriteParam
Error	Variable	Any detectable error within the ISDU transmission or DL_ISDUAbort requests, or any violation of the ISDU acknowledgment time (see Table 102)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.6.4 State machine of the Device ISDU handler

Figure 52 shows the state machine of the Device ISDU handler.

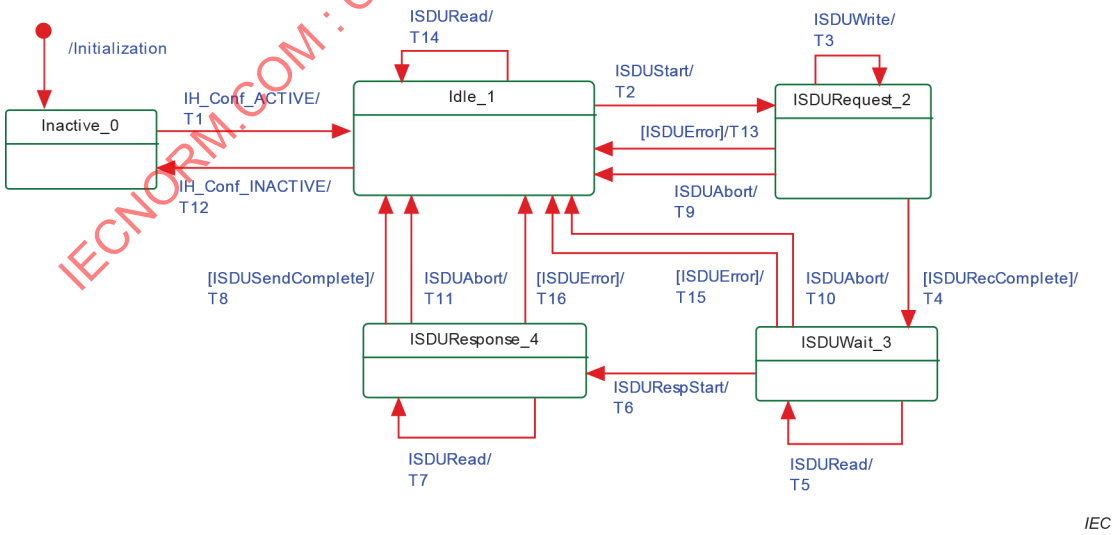


Figure 52 – State machine of the Device ISDU handler

Table 54 shows the state transition table of the Device ISDU handler.

Table 54 – State transition table of the Device ISDU handler

STATE NAME		STATE DESCRIPTION	
Inactive_0		Waiting on activation	
Idle_1		Waiting on next ISDU transmission	
ISDURequest_2		Reception of ISDU request	
ISDUWait_3		Waiting on data from application layer to transmit (see DL_ISDUTransport)	
ISDUResponse_4		Transmission of ISDU response data	
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T1	0	1	–
T2	1	2	Start receiving of ISDU request data
T3	2	2	Receive ISDU request data
T4	2	3	Invoke DL_ISDUTransport.ind to AL (see 7.2.1.6)
T5	3	3	Invoke OD.rsp with "busy" indication (see Table A.14)
T6	3	4	–
T7	4	4	Invoke OD.rsp with ISDU response data
T8	4	1	–
T9	2	1	–
T10	3	1	Invoke DL_ISDUAbort
T11	4	1	Invoke DL_ISDUAbort
T12	1	0	–
T13	2	1	Invoke DL_ISDUAbort
T14	1	1	Invoke OD.rsp with "no service" indication (see Table A.12 and Table A.14)
T15	3	1	Invoke DL_ISDUAbort
T16	4	1	Invoke DL_ISDUAbort
INTERNAL ITEMS		**TYPE**	**DEFINITION**
ISDUStart		Service	OD.ind(W, ISDU, Start, Data)
ISDUWrite		Service	OD.ind(W, ISDU, FlowCtrl, Data)
ISDURecComplete		Guard	If OD.ind(R, ISDU, Start, ...) received
ISDURespStart		Service	DL_ISDUTransport.rsp()
ISDURead		Service	OD.ind(R, ISDU, Start or FlowCtrl, ...)
ISDUSendComplete		Guard	If OD.ind(R, ISDU, IDLE, ...) received
ISDUAbort		Service	OD.ind(R/W, ISDU, Abort, ...)
ISDUError		Guard	If ISDU structure is incorrect or FlowCTRL error detected

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.7 Command handler

7.3.7.1 General

The command handler passes the control code (PDOUTVALID or PDOUTINVALID) contained in the DL_Control.req service primitive to the cyclically operating message handler via the OD.req service and MasterCommands. The message handler uses the page communication channel.

The permissible control codes for output Process Data are listed in Table 55.

Table 55 – Control codes

Control code	MasterCommand	Description
PDOUTVALID	ProcessDataOutputOperate	Output Process Data valid
PDOUTINVALID	DeviceOperate	Output Process Data invalid or missing

The command handler receives input Process Data status information via the PDInStatus service and propagates it within a DL_Control.ind service primitive.

In addition, the command handler translates Device mode change requests from System Management into corresponding MasterCommands (see Table B.2).

7.3.7.2 State machine of the Master command handler

Figure 53 shows the state machine of the Master command handler.

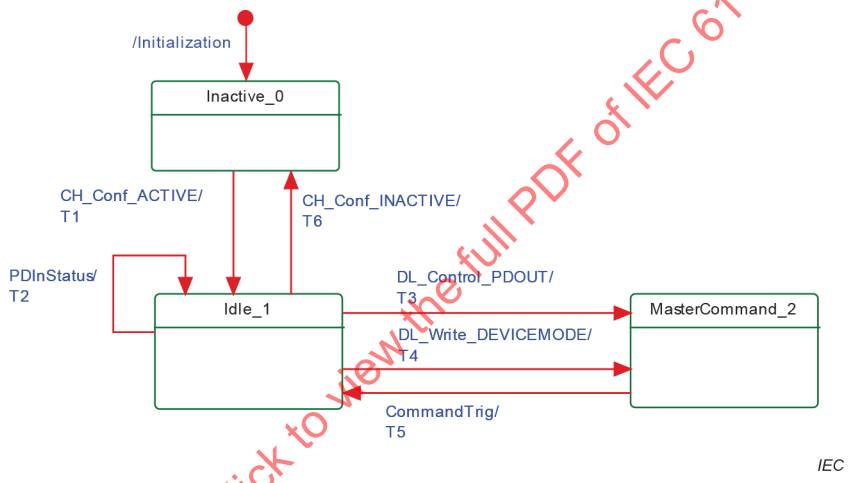


Figure 53 – State machine of the Master command handler

Table 56 shows the state transition table of the Master command handler.

Table 56 – State transition table of the Master command handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting on activation by DL-mode handler
Idle_1	Waiting on new command from AL: DL_Control (status of output PD) or from SM: DL_Write (change Device mode, for example to OPERATE), or waiting on PDInStatus.ind service primitive.
MasterCommand_2	Prepare data for OD.req service primitive. Waiting on demand from OD handler (CommandTrig).

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	1	If service PDInStatus.ind = VALID invoke DL_Control.ind (VALID) to signal valid input Process Data to AL. If service PDInStatus.ind = INVALID invoke DL_Control.ind (INVALID) to signal invalid input Process Data to AL.
T3	1	1	If service DL_Control.req = PDOUTVALID invoke OD.req (WRITE, PAGE, 0, 1, MasterCommand = 0x98). If service DL_Control.req = PDOUTINVALID invoke OD.req (WRITE, PAGE, 0, 1, MasterCommand = 0x99). See Table B.2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T4	1	2	The services DL_Write_DEVICEMODE translate into: INACTIVE: OD.req (WRITE, PAGE, 0, 1, MasterCommand = 0x5A) STARTUP: OD.req (WRITE, PAGE, 0, 1, MasterCommand = 0x97) PREOPERATE: OD.req (WRITE, PAGE, 0, 1, MasterCommand = 0x9A) OPERATE: OD.req (WRITE, PAGE, 0, 1, MasterCommand = 0x99)
T5	2	1	A call CommandTrig from the OD handler causes the command handler to invoke the OD.req service primitive and subsequently the message handler to send the appropriate MasterCommand to the Device.
T6	1	0	–

INTERNAL ITEMS	TYPE	DEFINITION
DEVICEMODE	Label	Any of the Device modes: INACTIVE, STARTUP, PREOPERATE, or OPERATE
PDOUT	Label	Any of the two output control codes: PDOUTVALID or PDOUTINVALID (see Table 55)

7.3.7.3 State machine of the Device command handler

Figure 54 shows the Device state machine of the command handler. It is mainly driven by MasterCommands from the Master's command handler to control the Device modes and the status of output Process Data. It also controls the status of input Process Data via the PDInStatus service.

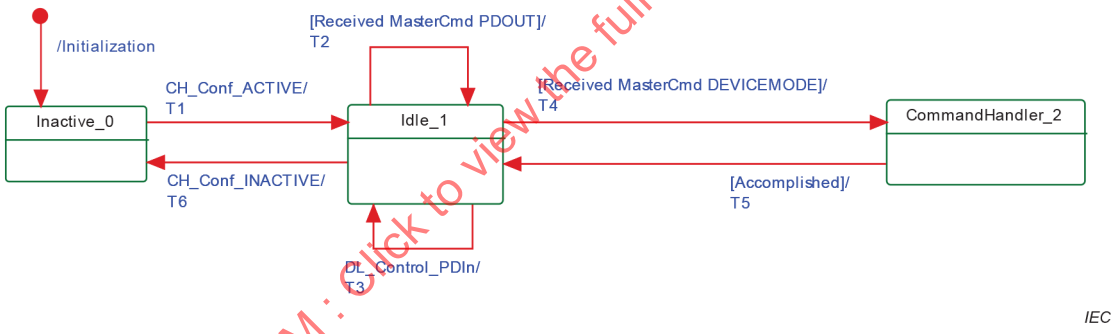


Figure 54 – State machine of the Device command handler

Table 57 shows the state transition table of the Device command handler.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 57 – State transition table of the Device command handler

STATE NAME			STATE DESCRIPTION
Inactive_0			Waiting on activation
Idle_1			Waiting on next MasterCommand
CommandHandler_2			Decompose MasterCommand and invoke specific actions (see B.1.2): If MasterCommand = 0x5A then change Device state to INACTIVE. If MasterCommand = 0x97 then change Device state to STARTUP. If MasterCommand = 0x9A then change Device state to PREOPERATE. If MasterCommand = 0x99 then change Device state to OPERATE.
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T1	0	1	–
T2	1	1	Invoke DL_Control.ind (PDOUTVALID) if received MasterCommand = 0x98. Invoke DL_Control.ind (PDOUTINVALID) if received MasterCommand = 0x99.
T3	1	1	If service DL_Control.req (VALID) then invoke PDInStatus.req (VALID). If service DL_Control.req (INVALID) then invoke PDInStatus.req (INVALID). Message handler uses PDInStatus service to set/reset the PD status flag (see A.1.5)
T4	1	2	–
T5	2	1	–
T6	1	0	–
INTERNAL ITEMS		**TYPE**	**DEFINITION**
<none>			

7.3.8 Event handler

7.3.8.1 Events

There are two types of Events, one without details, and another one with details. Events without details may have been implemented in legacy Devices, but they shall not be used for Devices in accordance with this document. However, all Masters shall support processing of both Events with details and Events without details.

The general structure and coding of Events is specified in Clause A.6. Event codes without details are specified in Table A.16. EventCodes with details are specified in Annex D. The structure of the Event memory for EventCodes with details within a Device is specified in Table 58.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 58 – Event memory

Address	Event slot number	Parameter Name	Description
0x00		StatusCode	Summary of status and error information. Also used to control read access for individual messages.
0x01	1	EventQualifier 1	Type, mode and source of the Event
0x02		EventCode 1	16-bit EventCode of the Event
0x03			
0x04	2	EventQualifier 2	Type, mode and source of the Event
0x05		EventCode 2	16-bit EventCode of the Event
0x06			
...			

Address	Event slot number	Parameter Name	Description
0x10	6	EventQualifier 6	Type, mode and source of the Event
0x11		EventCode 6	16-bit EventCode of the Event
0x12			
0x13 to 0x1F			Reserved for future use

7.3.8.2 Event processing

The Device AL writes an Event to the Event memory and then sets the "Event flag" bit, which is sent to the Master in the next message within the CKS octet (see 7.3.3.2 and A.1.5).

Upon reception of a Device reply message with the "Event flag" bit = 1, the Master shall switch from the ISDU handler to the Event handler. The Event handler starts reading the StatusCode.

If the "Event Details" bit is set (see Figure A.22), the Master shall read the Event details of the Events indicated in the StatusCode from the Event memory. Once it has read an Event detail, it shall invoke the service DL_Event.ind. After reception of the service DL_EventConf, the Master shall write any data to the StatusCode to reset the "Event flag" bit. The Event handling on the Master shall be completed regardless of the contents of the Event data received (EventQualifier, EventCode).

If the "Event Details" bit is not set (see Figure A.21) the Master Event handler shall generate the standardized Events according to Table A.16 beginning with the most significant bit in the EventCode.

Write access to the StatusCode indicates the end of Event processing to the Device. The Device shall ignore the data of this Master Write access. The Device then resets the "Event flag" bit and can now change the content of the fields in the Event memory.

7.3.8.3 State machine of the Master Event handler

Figure 55 shows the Master state machine of the Event handler.

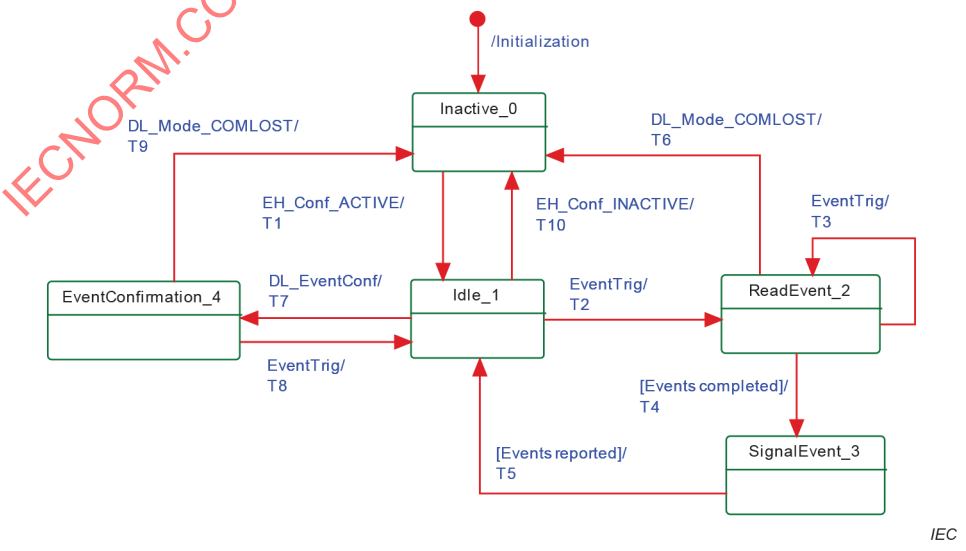


Figure 55 – State machine of the Master Event handler

Table 59 shows the state transition table of the Master Event handler.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 59 – State transition table of the Master Event handler

STATE NAME	STATE DESCRIPTION
Inactive_0	Waiting on activation
Idle_1	Waiting on next Event indication ("EventTrig" through On-request Data handler) or Event confirmation through service DL_EventConf from Master AL.
ReadEvent_2	Read Event data set from Device message by message through Event memory address. Check StatusCode for number of activated Events (see Table 58).
SignalEvent_3	Analyze Event data and invoke DL_Event indication to Master AL (see 7.2.1.15) for each available Event.
EventConfirmation_4	Waiting on Event confirmation transmission via service OD.req to the Device

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	2	Read Event StatusCode octet via service OD.req (R, DIAGNOSIS, Event memory address = 0, 1)
T3	2	2	Read octets from Event memory via service OD.req (R, DIAGNOSIS, incremented Event memory address, 1)
T4	2	3	–
T5	3	1	–
T6	2	0	–
T7	1	4	–
T8	4	1	Invoke OD.req (W, DIAGNOSIS, 0, 1, any data) with Write access to "StatusCode" (see Table 58) to confirm Event readout to Device
T9	4	0	–
T10	1	0	–

INTERNAL ITEMS	TYPE	DEFINITION
<None>		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.8.4 State machine of the Device Event handler

Figure 56 shows the state machine of the Device Event handler.

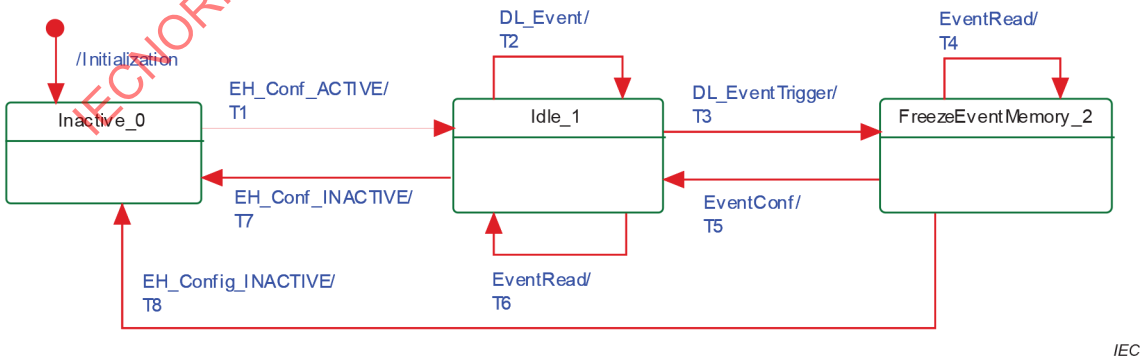


Figure 56 – State machine of the Device Event handler

Table 60 shows the state transition table of the Device Event handler.

Table 60 – State transition table of the Device Event handler

STATE NAME			STATE DESCRIPTION
Inactive_0			Waiting on activation
Idle_1			Waiting on DL-Event service from AL providing Event data and the DL_EventTrigger service to fire the "Event flag" bit (see A.1.5)
FreezeEventMemory_2			Waiting on readout of the Event memory and on Event memory readout confirmation through write access to the StatusCode
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T1	0	1	–
T2	1	1	Change Event memory entries with new Event data (see Table 58)
T3	1	2	Invoke service EventFlag.req (Flag = TRUE) to indicate Event activation to the Master via the "Event flag" bit. Mark all Event slots in memory as not changeable.
T4	2	2	Master requests Event memory data via EventRead (= OD.ind). Send Event data by invoking OD.rsp with Event data of the requested Event memory address.
T5	2	1	Invoke service EventFlag.req (Flag = FALSE) to indicate Event deactivation to the Master via the "Event flag" bit. Mark all Event slots in memory as invalid according to A.6.3.
T6	1	1	Send contents of Event memory by invoking OD.rsp with Event data
T7	1	0	–
T8	2	0	Discard Event memory data
INTERNAL ITEMS		**TYPE**	**DEFINITION**
EventRead		Service	OD.ind (R, DIAGNOSIS, Event memory address, length, data)
EventConf		Service	OD.ind (W, DIAGNOSIS, address = 0, data = don't care)

8 Application layer (AL)

8.1 General

Figure 57 shows an overview of the structure and services of the Master application layer (AL).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

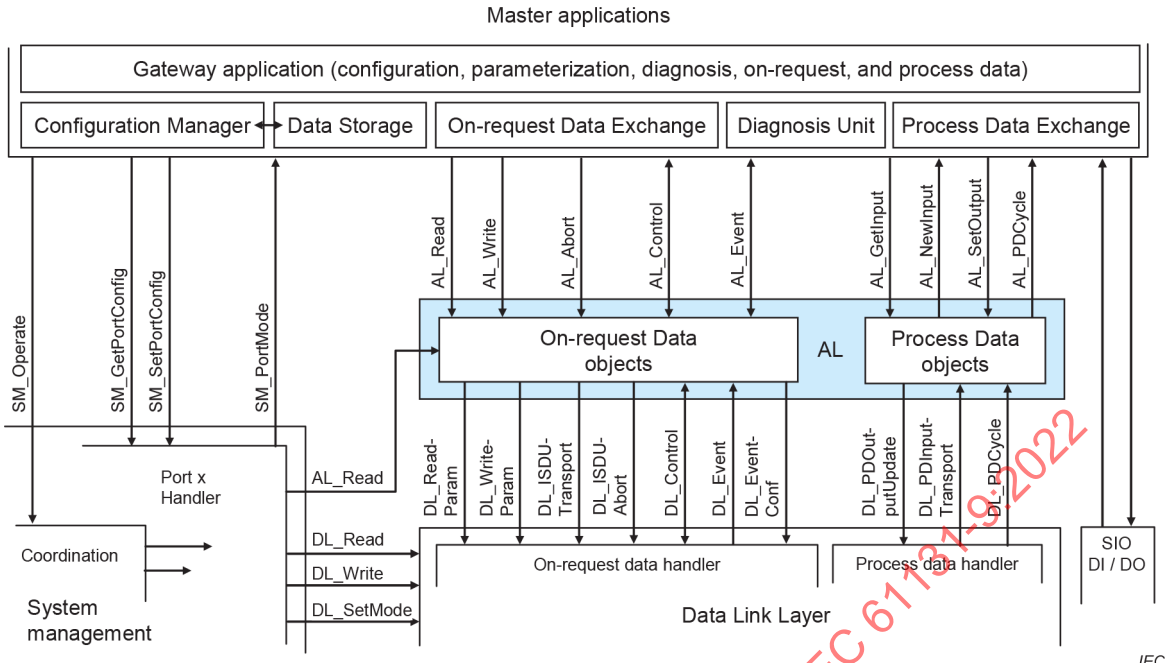
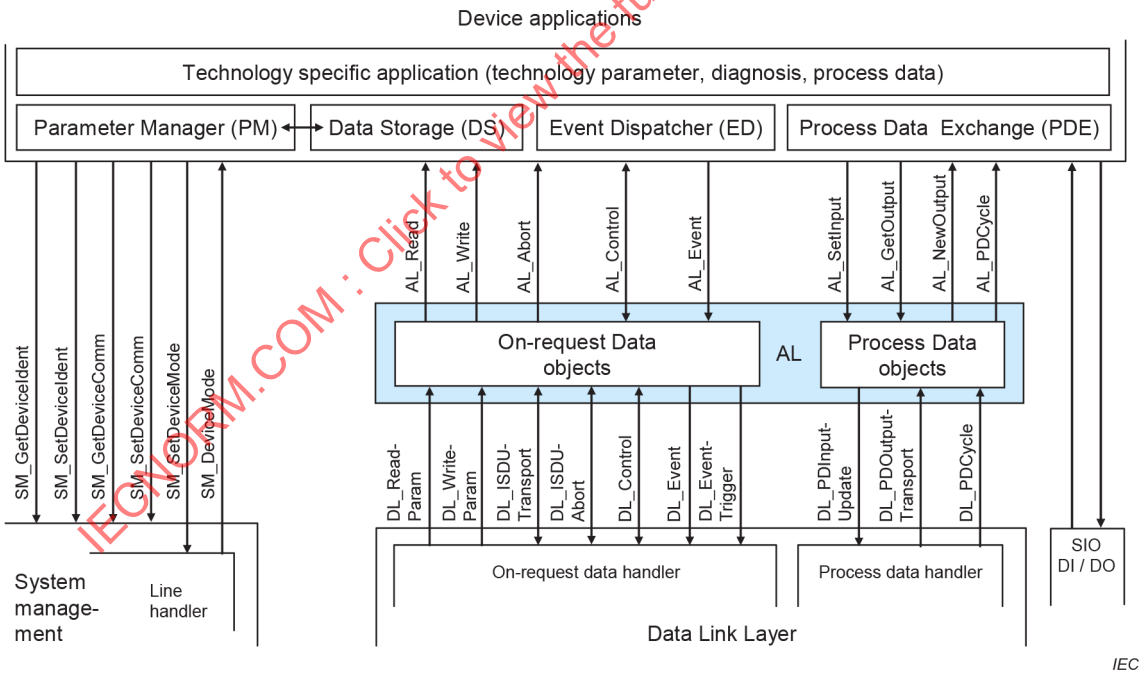


Figure 57 – Structure and services of the application layer (Master)

Figure 58 shows an overview of the structure and services of the Device application layer (AL).



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 58 – Structure and services of the application layer (Device)

8.2 Application layer services

8.2.1 AL services within Master and Device

This Clause 8.2 defines the services of the application layer (AL) to be provided to the Master and Device applications and System Management via its external interfaces. Table 61 lists the assignments of Master and Device to their roles as initiator or receiver for the individual AL services. Empty fields indicate no availability of this service on Master or Device.

Table 61 – AL services within Master and Device

Service name	Master	Device
AL_Read	R	I
AL_Write	R	I
AL_Abort	R	I
AL_GetInput	R	
AL_NewInput	I	
AL_SetInput		R
AL_PDCycle	I	I
AL_GetOutput		R
AL_NewOutput		I
AL_SetOutput	R	
AL_Event	I / R	R
AL_Control	I / R	R / I
Key (see 3.3.4) I Initiator of service R Receiver (Responder) of service		

8.2.2 AL Services

8.2.2.1 AL_Read

The AL_Read service is used to read On-request Data from a Device connected to a specific Port. The parameters of the service primitives are listed in Table 62.

Table 62 – AL_Read

Parameter name	.req	.ind	.rsp	.cnf
Argument	M	M		
Port	M			
Index	M	M		
Subindex	M	M		
Result (+)			S	S(=)
Port				M
Data			M	M(=)
Result (−)			S	S(=)
Port				M
ErrorInfo				M(=)

Argument

The service-specific parameters are transmitted in the argument.

Port

This parameter contains the Port number for the On-request Data to be read.

Parameter type: Unsigned8.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Index

This parameter indicates the address of On-request Data objects to be read from the Device. Index 0 in conjunction with Subindex 0 addresses the entire set of Direct Parameters from 0 to 15 (see Direct Parameter page 1 in Table B.1) or in conjunction with Subindices 1 to 16 the individual parameters from 0 to 15. Index 1 in conjunction with Subindex 0 addresses the entire set of Direct Parameters from addresses 16 to 31 (see Direct Parameter page 2 in Table B.1) or in conjunction with Subindices 1 to 16 the individual parameters from 16 to 31. It uses the page communication channel (see Figure 7) for both and always returns a positive result. For all the other Indices (see Clause B.2) the ISDU communication channel is used.

Permitted values: 0 to 65 535 (See B.2.1 for constraints).

Subindex

This parameter indicates the element number within a structured On-request Data object. A value of 0 indicates the entire set of elements.

Permitted values: 0 to 255.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Port

This parameter contains the Port number of the requested On-request Data.

Data

This parameter contains the read values of the On-request Data.

Parameter type: Octet string.

Result (−):

This selection parameter indicates that the service failed.

Port

This parameter contains the Port number for the requested On-request Data.

ErrorInfo

This parameter contains error information.

Permitted values: see Annex C.

NOTE The AL maps DL ErrorInfos into its own AL ErrorInfos using Annex C.

8.2.2.2 AL_Write

The AL_Write service is used to write On-request Data to a Device connected to a specific Port. The parameters of the service primitives are listed in Table 63.

Table 63 – AL_Write

Parameter name	.req	.ind	.rsp	.cnf
Argument	M	M		
Port	M			
Index	M	M		
Subindex	M	M		
Data	M	M(=)		
Result (+)			S	S(=)
Port				M
Result (−)			S	S(=)
Port				M
ErrorInfo			M	M(=)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The service-specific parameters are transmitted in the argument.

Port

This parameter contains the Port number for the On-request Data to be written.

Parameter type: Unsigned8.

Index

This parameter indicates the address of On-request Data objects to be written to the Device. Index 0 always returns a negative result except for use in conjunction with Subindex 16 at Devices without ISDU support. Index 1 in conjunction with Subindex 0 addresses the entire set of Direct Parameters from addresses 16 to 31 (see Direct Parameter page 2 in Table B.1) or in conjunction with Subindices 1 to 16 the individual parameters from 16 to 31. It uses the page communication channel (see Figure 7) in case of Index 1 and always returns a positive result. For all other Indices (see Clause B.2) the ISDU communication channel is used.

Permitted values: 1 to 65 535 (see Table 102).

Subindex

This parameter indicates the element number within a structured On-request Data object. A value of 0 indicates the entire set of elements.

Permitted values: 0 to 255.

Data

This parameter contains the values of the On-request Data.

Parameter type: Octet string.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Port

This parameter contains the Port number of the On-request Data.

Result (−):

This selection parameter indicates that the service failed.

Port

This parameter contains the Port number of the On-request Data.

ErrorInfo

This parameter contains error information.

Permitted values: see Annex C.

8.2.2.3 AL_Abort

The AL_Abort service is used to abort a current AL_Read or AL_Write service on a specific Port. Invocation of this service abandons the response to an AL_Read or AL_Write service in progress on the Master. The parameters of the service primitives are listed in Table 64.

Table 64 – AL_Abort

Parameter name	.req	.ind
Argument	M	M
Port	M	

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The service-specific parameter is transmitted in the argument.

Port

This parameter contains the Port number of the service to be abandoned.

8.2.2.4 AL_GetInput

The AL_GetInput service reads the input data within the Process Data provided by the data link layer of a Device connected to a specific Port. The parameters of the service primitives are listed in Table 65.

Table 65 – AL_GetInput

Parameter name	.req	.cnf
Argument	M	
Port	M	
Result (+)		S
Port		M
InputData		M
Result (−)		S
Port		M
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

Port

This parameter contains the Port number for the Process Data to be read.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Port

This parameter contains the Port number for the Process Data.

InputData

This parameter contains the values of the requested process input data of the specified Port.

Parameter type: Octet string.

Result (−):

This selection parameter indicates that the service failed.

Port

This parameter contains the Port number for the Process Data.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_DATA (DL did not provide Process Data).

8.2.2.5 AL_NewInput

The AL_NewInput local service indicates the receipt of updated input data within the Process Data of a Device connected to a specific Port. The parameters of the service primitives are listed in Table 66.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 66 – AL_NewInput

Parameter name	.ind
Argument	M
Port	M

Argument

The service-specific parameter is transmitted in the argument.

Port

This parameter specifies the Port number of the received Process Data.

8.2.2.6 AL_SetInput

The AL_SetInput local service updates the input data within the Process Data of a Device. The parameters of the service primitives are listed in Table 67.

Table 67 – AL_SetInput

Parameter name	.req	.cnf
Argument	M	
InputData	M	
Result (+)		S
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

InputData

This parameter contains the Process Data values of the input data to be transmitted.

Parameter type: Octet string.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (Service unavailable within current state).

8.2.2.7 AL_PDCycle

The AL_PDCycle local service indicates the end of a Process Data cycle. The Device application can use this service to transmit new input data to the application layer via AL_SetInput. The parameters of the service primitives are listed in Table 68.

Table 68 – AL_PDCycle

Parameter name	.ind
Argument	
Port	O

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The service-specific parameter is transmitted in the argument.

Port

This parameter contains the Port number of the received new Process Data (Master only).

8.2.2.8 AL_GetOutput

The AL_GetOutput service reads the output data within the Process Data provided by the data link layer of the Device. The parameters of the service primitives are listed in Table 69.

Table 69 – AL_GetOutput

Parameter name	.req	.cnf
Argument	M	
Result (+) OutputData		S M
Result (−) ErrorInfo		S M

Argument

The service-specific parameters are transmitted in the argument.

Result (+):

This selection parameter indicates that the service has been executed successfully.

OutputData

This parameter contains the Process Data values of the requested output data.

Parameter type: Octet string.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
NO_DATA (DL did not provide Process Data).

8.2.2.9 AL_NewOutput

The AL_NewOutput local service indicates the receipt of updated output data within the Process Data of a Device. This service has no parameters. The service primitives are shown in Table 70.

Table 70 – AL_NewOutput

Parameter name	.ind
<None>	

8.2.2.10 AL_SetOutput

The AL_SetOutput local service updates the output data within the Process Data of a Master. The parameters of the service primitives are listed in Table 71.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 71 – AL_SetOutput

Parameter name	.req	.cnf
Argument	M	
Port	M	
OutputData	M	
Result (+)		S
Port		M
Result (−)		S
Port		M
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

Port

This parameter contains the Port number of the Process Data to be written.

OutputData

This parameter contains the output data to be written at the specified Port.

Parameter type: Octet string.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Port

This parameter contains the Port number for the Process Data.

Result (−):

This selection parameter indicates that the service failed.

Port

This parameter contains the Port number for the Process Data.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (Service unavailable within current state).

8.2.2.11 AL_Event

The AL_Event service indicates up to six pending status or error messages. The source of one Event can be local (Master) or remote (Device). The Event can be triggered by a communication layer or by an application. The parameters of the service primitives are listed in Table 72.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 72 – AL_Event

Parameter name		.req	.ind	.rsp	.cnf
Argument		M	M	M	M
Port			M	M	M
EventCount		M	M		
Event (1)	Instance	M	M		
	Mode	M	M		
	Type	M	M		
	Origin		M		
	EventCode	M	M		
…					
Event (n)	Instance	M	M		
	Mode	M	M		
	Type	M	M		
	Origin		M		
	EventCode	M	M		

Argument

The service-specific parameters are transmitted in the argument.

Port

This parameter contains the Port number of the Event data.

EventCount

This parameter indicates the number n (1 to 6) of Events in the Event memory.

Event(x)

Depending on EventCount this parameter exists n times. Each instance contains the following elements.

Instance

This parameter indicates the Event source.

Permitted values: Application (see Table A.17).

Mode

This parameter indicates the Event mode.

Permitted values: SINGLESHOT, APPEARS, DISAPPEARS (see Table A.20).

Type

This parameter indicates the Event category.

Permitted values: ERROR, WARNING, NOTIFICATION (see Table A.19).

Origin

This parameter indicates whether the Event was generated in the local communication section or remotely (in the Device).

Permitted values: LOCAL, REMOTE.

EventCode

This parameter contains a code identifying a certain Event.

Permitted values: see Annex D.

8.2.2.12 AL_Control

The AL_Control service contains the Process Data qualifier status information transmitted to and from the Device application. This service shall be synchronized with AL_GetInput and AL_SetOutput respectively (see 11.7.2.1). The parameters of the service primitives are listed in Table 73.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 73 – AL_Control

Parameter name	.req	.ind

Argument
  Port
  ControlCode
	M
	C
	M
		M
		C
		M

Argument

The service-specific parameters are transmitted in the argument.

Port

This parameter contains the number of the related Port.

ControlCode

This parameter contains the qualifier status of the Process Data (PD).

Permitted values:
VALID (Input Process Data valid),
INVALID (Input Process Data invalid),
PDOUTVALID (Output Process Data valid, see Table 55),
PDOUTINVALID (Output Process Data invalid, see Table 55).

8.3 Application layer protocol

8.3.1 Overview

Figure 8 shows that the application layer offers services for data objects which are transformed into the special communication channels of the data link layer.

The application layer manages the data transfer with all its assigned Ports. This means that AL service calls need to identify the particular Port they are related to.

8.3.2 On-request Data transfer

8.3.2.1 OD state machine of the Master AL

Figure 59 shows the state machine for the handling of On-request Data (OD) within the application layer.

"AL_Service" represents any AL service in Table 61 related to OD. "Portx" indicates a particular Port number.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

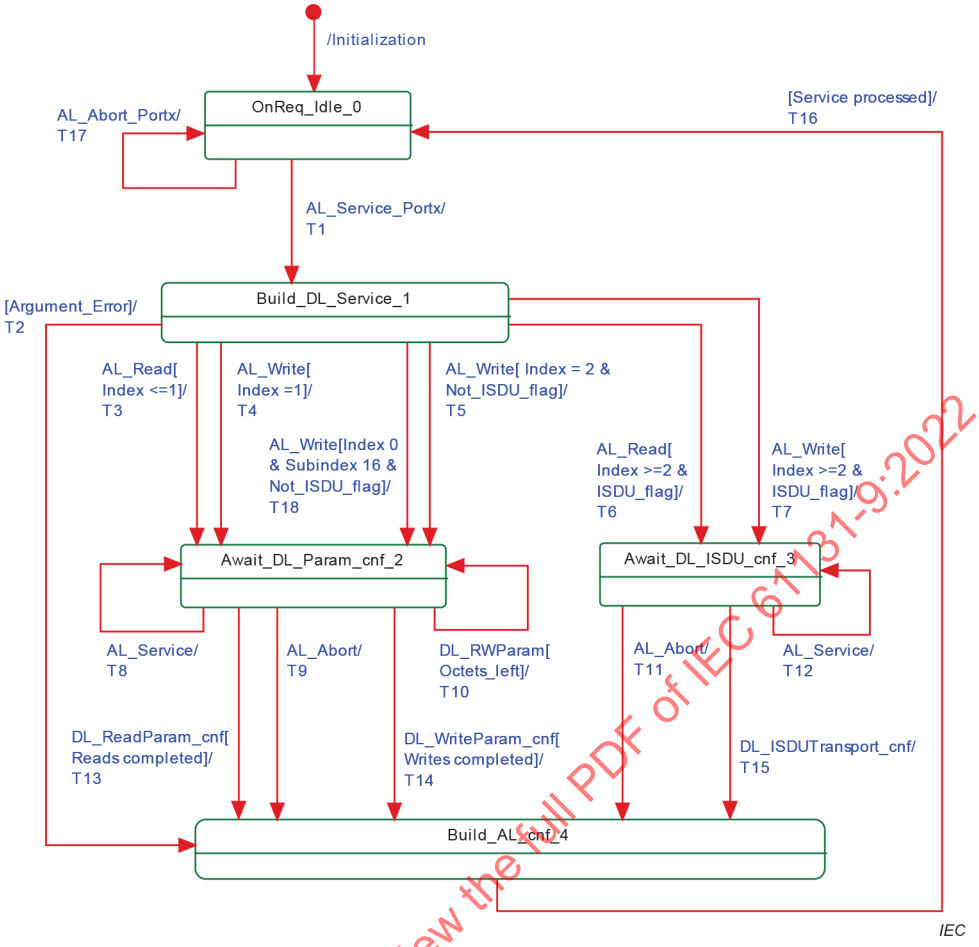


Figure 59 – OD state machine of the Master AL

Table 74 shows the states and transitions for the OD state machine of the Master AL.

Table 74 – States and transitions for the OD state machine of the Master AL

STATE NAME	STATE DESCRIPTION
OnReq_Idle_0	AL service invocations from the Master applications or from the SM Portx handler (see Figure 57) can be accepted within this state.
Build_DL_Service_1	Within this state AL service calls are checked, and corresponding DL services are created within the subsequent states. In case of an error in the arguments of the AL service a negative AL confirmation is created and returned.
Await_DL_Param_cnf_2	Within this state the AL service call is transformed in a sequence of as many DL_ReadParam or DL_WriteParam calls as needed (Direct Parameter page access; see page communication channel in Figure 7). All asynchronously occurred AL service invocations except AL_Abort are rejected (see 3.3.8).
Await_DL_ISDU_cnf_3	Within this state the AL service call is transformed in a DL_ISDUTransport service call (see ISDU communication channel in Figure 7). All asynchronously occurred AL service invocations except AL_Abort are rejected (see 3.3.8).
Build_AL_cnf_4	Within this state an AL service confirmation is created depending on an argument error, the DL service confirmation, or an AL_Abort.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	Memorize the Port number "Portx".
T2	1	4	Prepare negative AL service confirmation.
T3	1	2	Prepare DL_ReadParam for Index 0 or 1.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T4	1	2	Prepare DL_WriteParam for Index 1.
T5	1	2	Prepare DL_Write for Address 0x0F if the Device does not support ISDU.
T6	1	3	Prepare DL_ISDUTransport (read)
T7	1	3	Prepare DL_ISDUTransport (write)
T8	2	2	Return negative AL service confirmation on this asynchronous service call.
T9	2	4	All current DL service actions are abandoned, and a negative AL service confirmation is prepared.
T10	2	2	Call next DL_ReadParam or DL_WriteParam service if not all OD are transferred.
T11	3	4	All current DL service actions are abandoned, and a negative AL service confirmation is prepared.
T12	3	3	Return negative AL service confirmation on this asynchronous service call.
T13	2	4	Prepare positive AL service confirmation.
T14	2	4	Prepare positive AL service confirmation.
T15	3	4	Prepare positive AL service confirmation.
T16	4	0	Return positive AL service confirmation with Port number "Portx".
T17	0	0	Return negative AL service confirmation with Port number "Portx".
T18	1	2	Prepare DL_Write for Address 0x0F if the Device does not support ISDU.

INTERNAL ITEMS	TYPE	DEFINITION
Argument_Error	Bool	Illegal values within the service body, for example "Port number or Index out of range"
DL_RWParam	Label	"DL_RWParam": DL_WriteParam_cnf or DL_ReadParam_cnf
Completed	Bool	No more OD left for transfer
Octets_left	Bool	More OD for transfer
Portx	Variable	Service body variable indicating the Port number
ISDU_Flag	Bool	Device supports ISDU
AL_Service	Label	"AL_Service" represents any AL service in Table 61 related to OD

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.2.2 OD state machine of the Device AL

Figure 60 shows the state machine for the handling of On-request Data (OD) within the application layer of a Device.

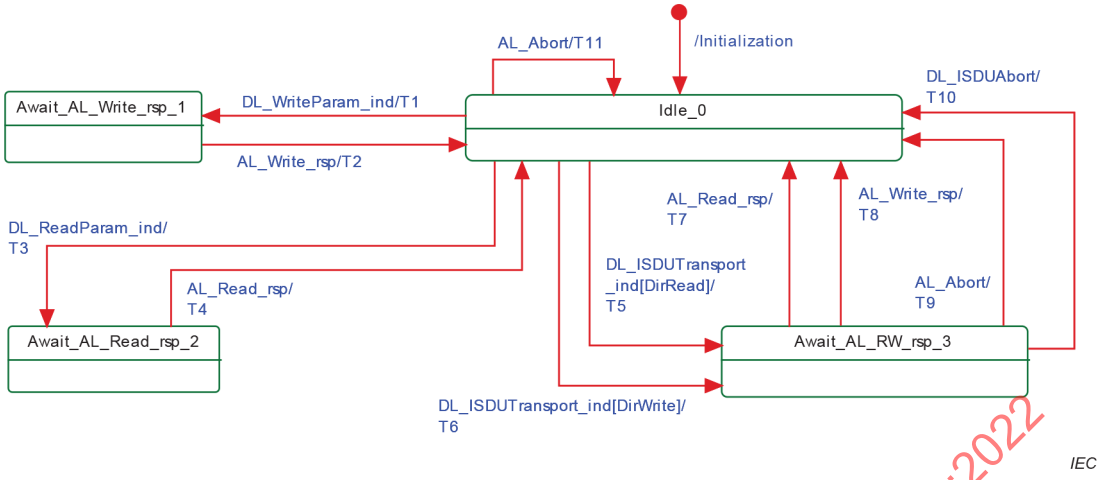


Figure 60 – OD state machine of the Device AL

Table 75 shows the states and transitions for the OD state machine of the Device AL.

Table 75 – States and transitions for the OD state machine of the Device AL

STATE NAME	STATE DESCRIPTION
Idle_0	The Device AL is waiting on subordinated DL service calls triggered by Master messages.
Await_AL_Write_rsp_1	The Device AL is waiting on a response from the technology specific application (write access to Direct Parameter page).
Await_AL_Read_rsp_2	The Device AL is waiting on a response from the technology specific application (read access to Direct Parameter page).
Await_AL_RW_rsp_3	The Device AL is waiting on a response from the technology specific application (read or write access via ISDU).

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	Invoke AL_Write.
T2	1	0	Invoke DL_WriteParam (16 to 31).
T3	0	2	Invoke AL_Read.
T4	2	0	Invoke DL_ReadParam (0 to 31).
T5	0	3	Invoke AL_Read.
T6	0	3	Invoke AL_Write.
T7	3	0	Invoke DL_ISDUTransport (read)
T8	3	0	Invoke DL_ISDUTransport (write)
T9	3	0	Current AL_Read or AL_Write abandoned upon this asynchronous AL_Abort service call. Return negative DL_ISDUTransport (see 3.3.8).
T10	3	0	Current waiting on AL_Read or AL_Write abandoned.
T11	0	0	Current DL_ISDUTransport abandoned. All OD are set to "0".

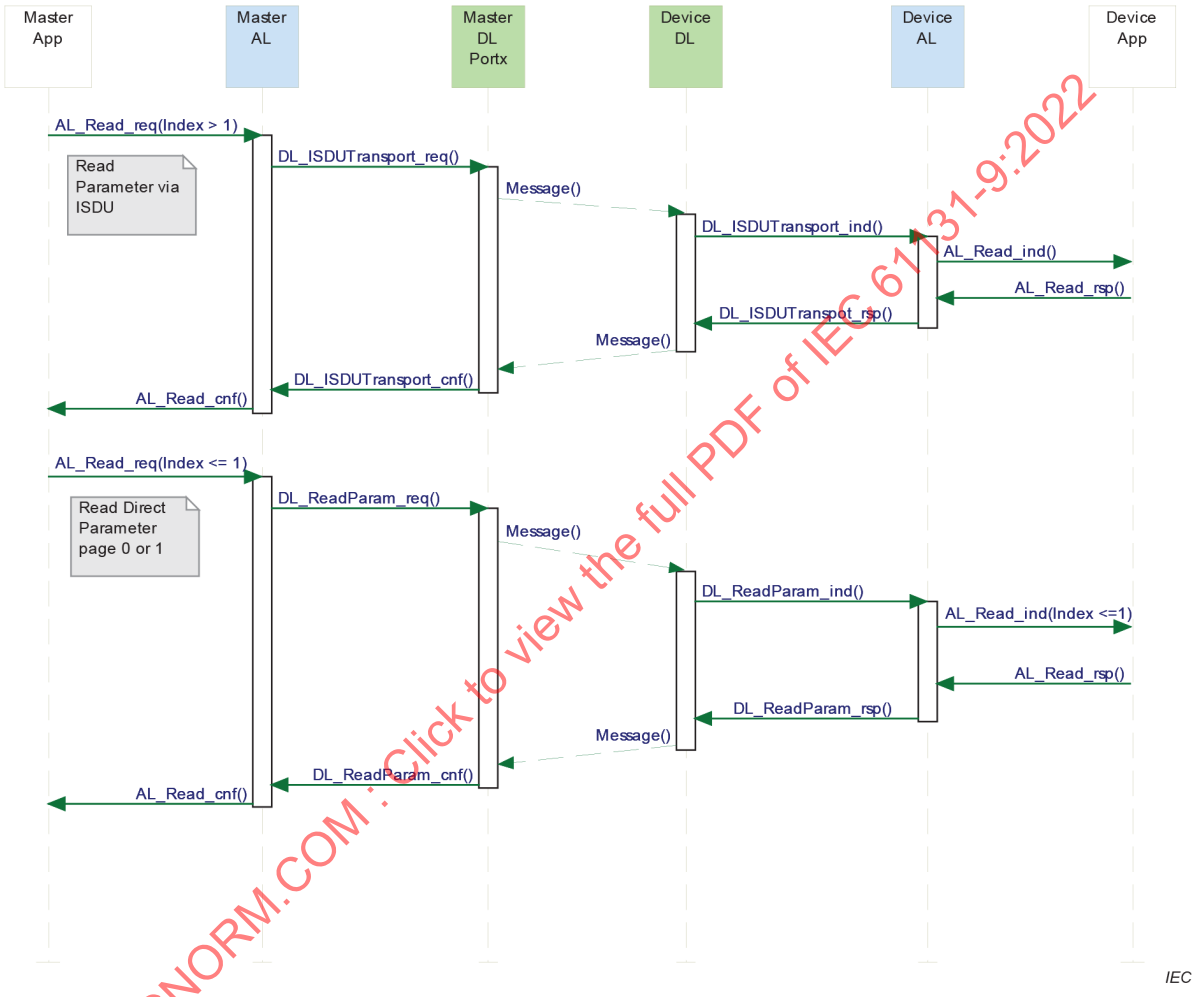
INTERNAL ITEMS	TYPE	DEFINITION
DirRead	Bool	Access direction: DL_ISDUTransport (read) causes an AL_Read
DirWrite	Bool	Access direction: DL_ISDUTransport (write) causes an AL_Read

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.2.3 Sequence diagrams for On-request Data

Figure 61 through Figure 63 demonstrate complete interactions between Master and Device for several On-request Data exchange use cases.

Figure 61 demonstrates two examples for the exchange of On-request Data. For Indices > 1 this is performed with the help of ISDUs and corresponding DL services (ISDU communication channel according to Figure 7). Access to Direct Parameter pages 0 and 1 uses different DL services (page communication channel according to Figure 7)



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 61 – Sequence diagram for the transmission of On-request Data

Figure 62 demonstrates the behavior of On-request Data exchange in case of an error such as "requested Index not available" (see Table C.1).

Another possible error occurs when the Master application (gateway) tries to read an Index > 1 from a Device, which does not support ISDU. The Master AL would respond immediately with "NO_ISDU_SUPPORTED" as the features of the Device are acquired during start-up through reading the Direct Parameter page 1 via the parameter "M-sequence Capability" (see Table B.1).

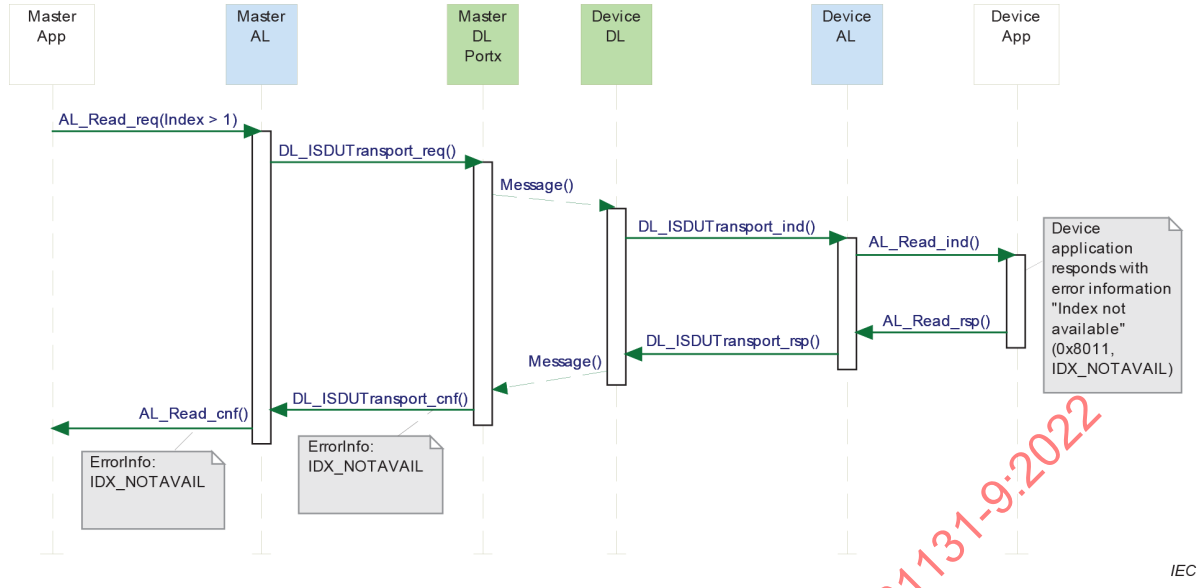


Figure 62 – Sequence diagram for On-request Data in case of errors

Figure 63 demonstrates the behavior of On-request Data exchange in case of an ISDU timeout (5 000 ms). A Device shall respond within less than the "ISDU acknowledgment time" (see 10.8.5).

NOTE See Table 102 for system constants such as "ISDU acknowledgment time".

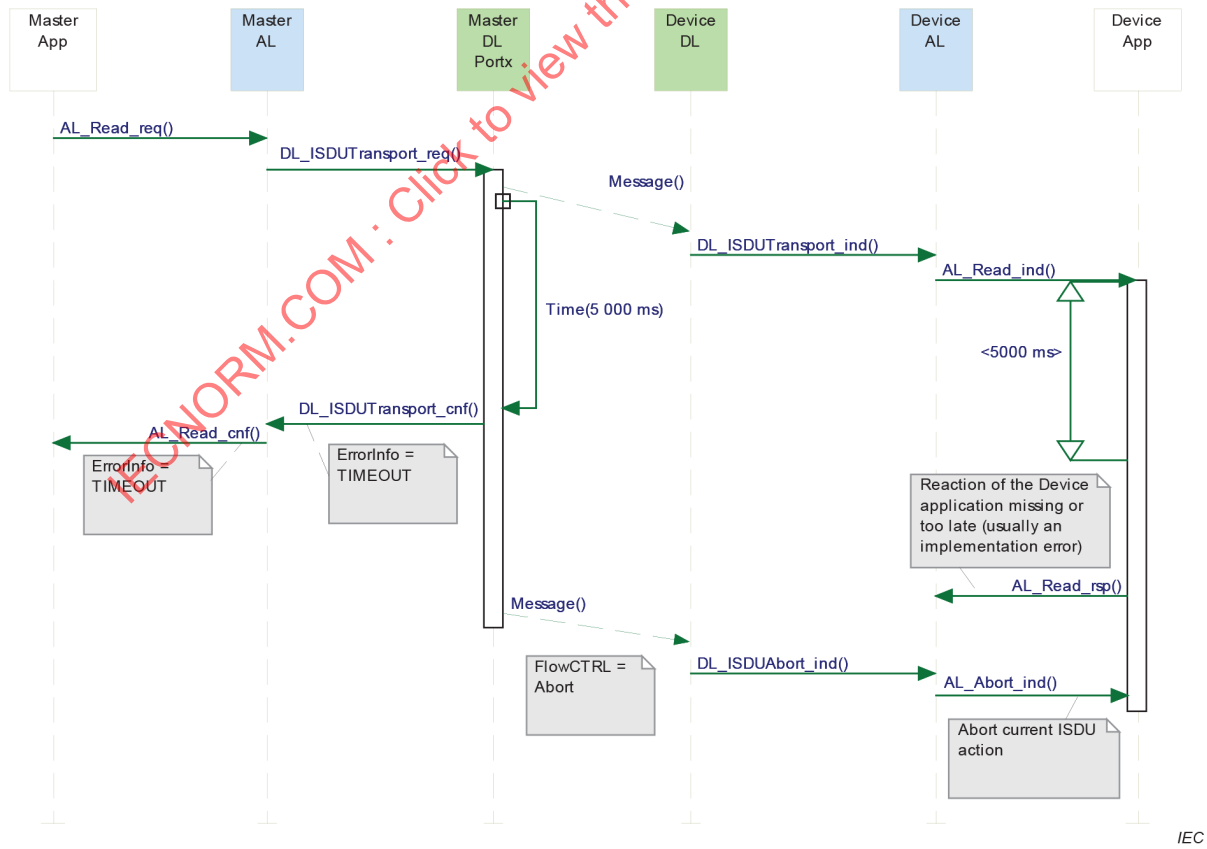


Figure 63 – Sequence diagram for On-request Data in case of timeout

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.3 Event processing

8.3.3.1 Event state machine of the Master AL

Figure 64 shows the Event state machine of the Master application layer.

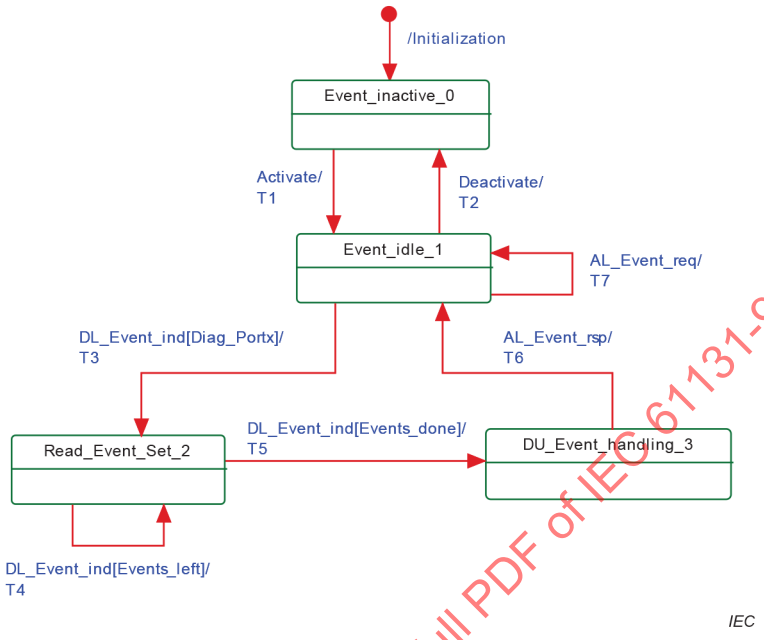


Figure 64 – Event state machine of the Master AL

Table 76 specifies the states and transitions of the Event state machine of the Master application layer.

Table 76 – State and transitions of the Event state machine of the Master AL

STATE NAME	STATE DESCRIPTION
Event_inactive_0	The AL Event handling of the Master is inactive.
Event_idle_1	The Master AL is ready to accept DL_Events (diagnosis information) from the DL.
Read_Event_Set_2	The Master AL received a DL_Event_ind with diagnosis information. After this first DL_Event.ind, the AL collects the complete set (1 to 6) of DL_Events of the current EventTrigger (see 11.6).
DU_Event_handling_3	The Master AL remains in this state as long as the Diagnosis Unit (see 11.6) did not acknowledge the AL_Event.ind.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	0	–
T3	1	2	–
T4	2	2	–
T5	2	3	AL_Event.ind
T6	3	1	DL_EventConf.req
T7	1	1	AL_Event.ind

INTERNAL ITEMS	TYPE	DEFINITION
Diag_Portx	Bool	Event set contains diagnosis information with details.
Events_done	Bool	Event set is processed.
Events_left	Bool	Event set not yet completed.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.3.2 Event state machine of the Device AL

Figure 65 shows the Event state machine of the Device application layer

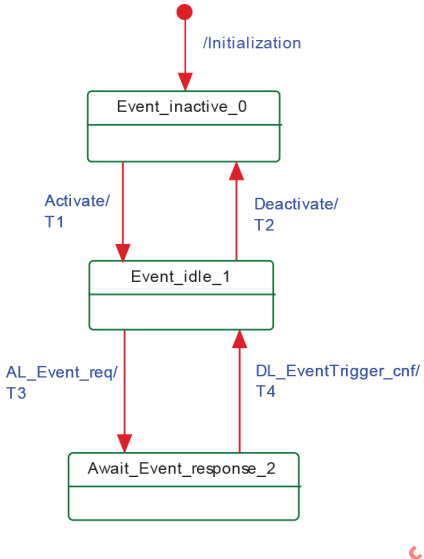


Figure 65 – Event state machine of the Device AL

Table 77 specifies the states and transitions of the Event state machine of the Device application layer.

Table 77 – State and transitions of the Event state machine of the Device AL

STATE NAME	STATE DESCRIPTION
Event_inactive_0	The AL Event handling of the Device is inactive.
Event_idle_1	The Device AL is ready to accept AL_Events (diagnosis information) from the technology specific Device applications for the transfer to the DL. The Device applications can create new Events during this time.
Await_event_response_2	The Device AL propagated an AL_Event with diagnosis information and waits on a DL_EventTrigger confirmation of the DL. The Device AL shall not accept any new AL_Event during this time.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	0	–
T3	1	2	An AL_Event request triggers a DL_Event and the corresponding DL_EventTrigger service. The DL_Event carries the diagnosis information from AL to DL. The DL_EventTrigger sets the Event flag within the cyclic data exchange (see A.1.5).
T4	2	1	A DL_EventTrigger confirmation triggers an AL_Event confirmation.

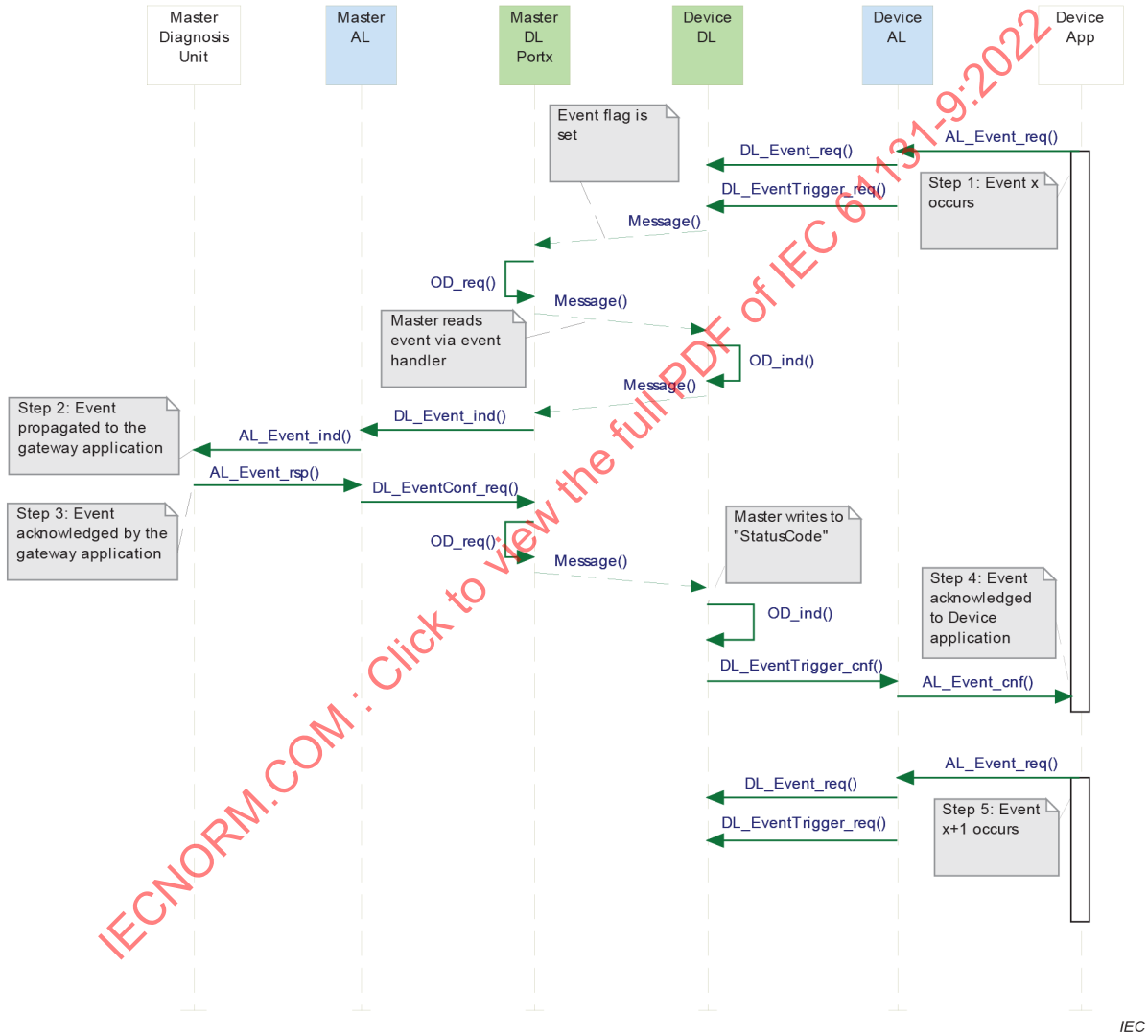
INTERNAL ITEMS	TYPE	DEFINITION
none		

8.3.3.3 Single Event scheduling

Figure 66 shows how a single Event from a Device is processed, in accordance with the relevant state machines.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

– The Device application creates an Event request (Step 1), which is passed from the AL to the DL and buffered within the Event memory (see Table 58).
– The Device AL activates the EventTrigger service to raise the Event flag, which causes the Master to read the Event from the Event memory.
– The Master then propagates this Event to the gateway application (Step 2), and waits for an Event acknowledgment.
– Once the Event acknowledgment is received (Step 3), it is indicated to the Device by writing to the StatusCode (Step 4).
– The Device confirms the original Event request to its application (Step 5), which may now initiate a new Event request.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 66 – Single Event scheduling

8.3.3.4 Multi Event transport (legacy Devices only)

Besides the method specified in 8.3.3.3 in which each single Event is conveyed through the layers and acknowledged by the gateway application, all Masters shall support a so-called "multi Event transport" which allows up to six Events to be transferred at a time. The Master AL transfers the Event set as a single diagnosis indication to the gateway application and returns a single acknowledgment for the entire set to the legacy Device application.

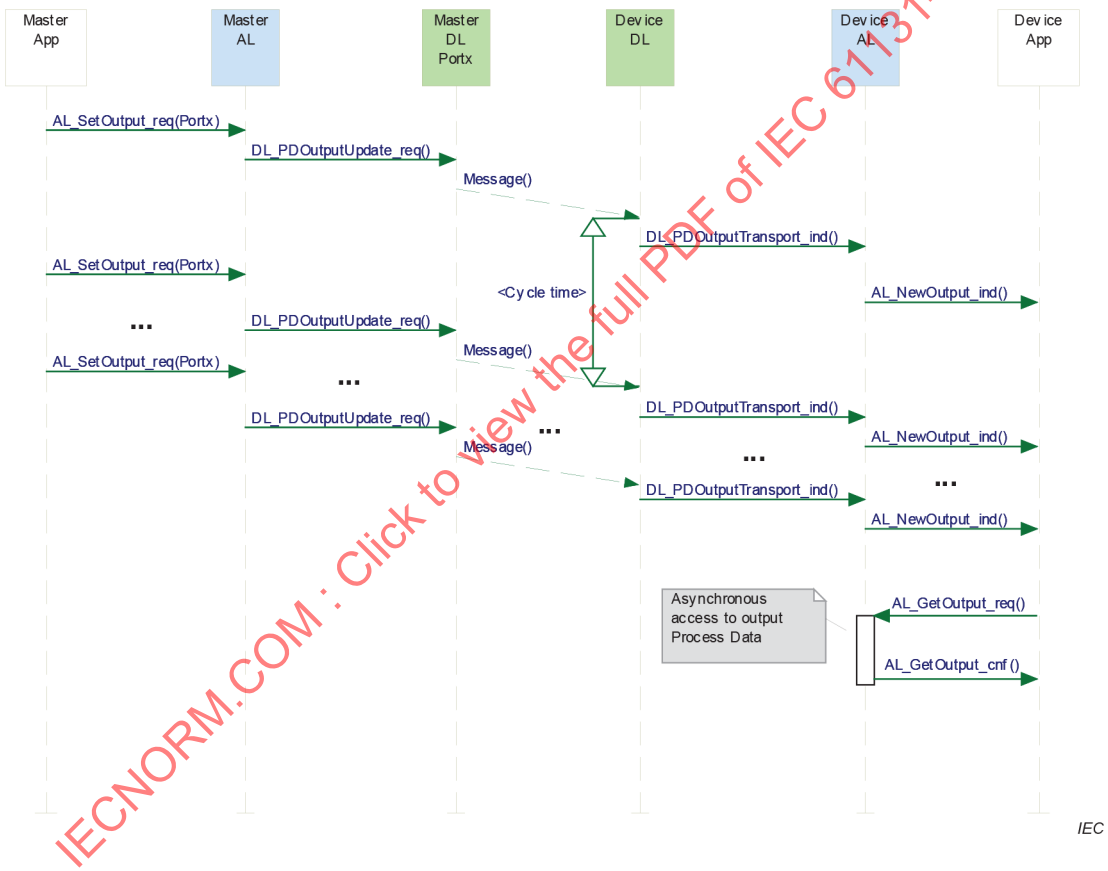
Figure 66 also applies for the multi-Event transport, except that this transport uses one DL_Event indication for each Event memory slot, and a single AL_Event indication for the entire Event set.

One AL_Event.req carries up to six Events and one AL_Event.ind indicates up to six pending Events. AL_Event.rsp and AL_Event.cnf refer to the indicated entire Event set.

8.3.4 Process Data cycles

Figure 67 and Figure 68 demonstrate complete interactions between Master and Device for output and input Process Data use cases.

Figure 67 demonstrates how the AL and DL services of Master and Device are involved in the cyclic exchange of output Process Data. The Device application is able to acquire the current values of output PD via the AL_GetOutput service.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 67 – Sequence diagram for output Process Data

Figure 68 demonstrates how the AL and DL services of Master and Device are involved in the cyclic exchange of input Process Data. The Master application is able to acquire the current values of input PD via the AL_GetInput service.

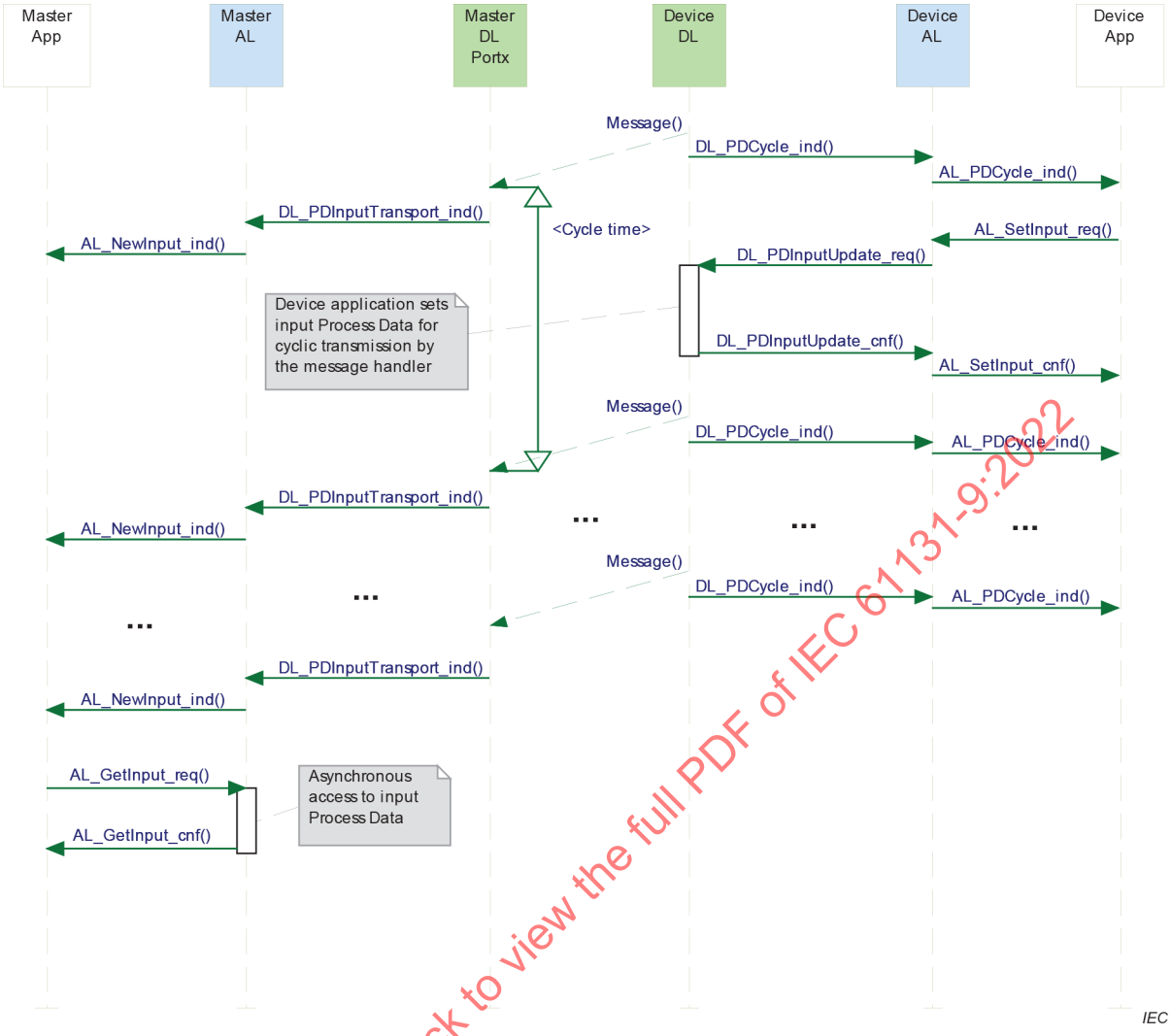


Figure 68 – Sequence diagram for input Process Data

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

9 System Management (SM)

9.1 General

The SDCI System Management is responsible for the coordinated startup of the Ports within the Master and the corresponding operations within the connected Devices. The difference between the SM of the Master and the Device is more significant than with the other layers. Consequently, the structure of this Clause 9 separates the services and protocols of Master and Device.

9.2 System Management of the Master

9.2.1 Overview

The Master System Management services are used to set up the Master Ports and the system for all possible operational modes.

The Master SM adjusts Ports through

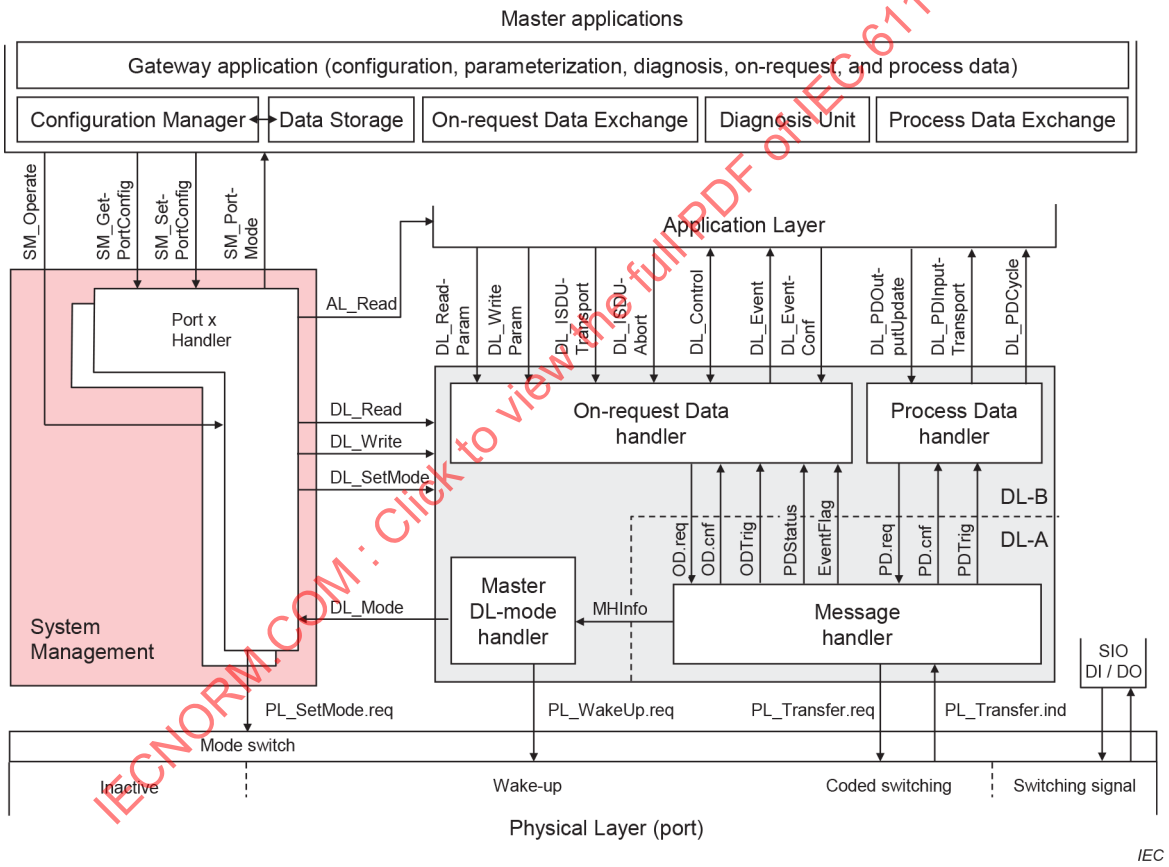
– establishing the required communication protocol revision,
– checking the Device compatibility (actual Device identifications match expected values),
– adjusting adequate Master M-sequence types and MasterCycleTimes.

For this it uses the following services shown in Figure 69:

– SM_SetPortConfig transfers the necessary Device parameters (configuration data) from Configuration Management (CM) to System Management (SM). The Port is then started implicitly.
– SM_PortMode reports the positive result of the Port setup back to CM in case of correct Port setup and inspection. It reports the negative result back to CM via corresponding "errors" in case of mismatching revisions and incompatible Devices.
– SM_GetPortConfig reads the actual and effective parameters.
– SM_Operate switches a single Port into the "OPERATE" mode.

Figure 69 provides an overview of the structure and services of the Master System Management.

The Master System Management needs one application layer service (AL_Read) to acquire data (identification parameter) from special Indices for inspection.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

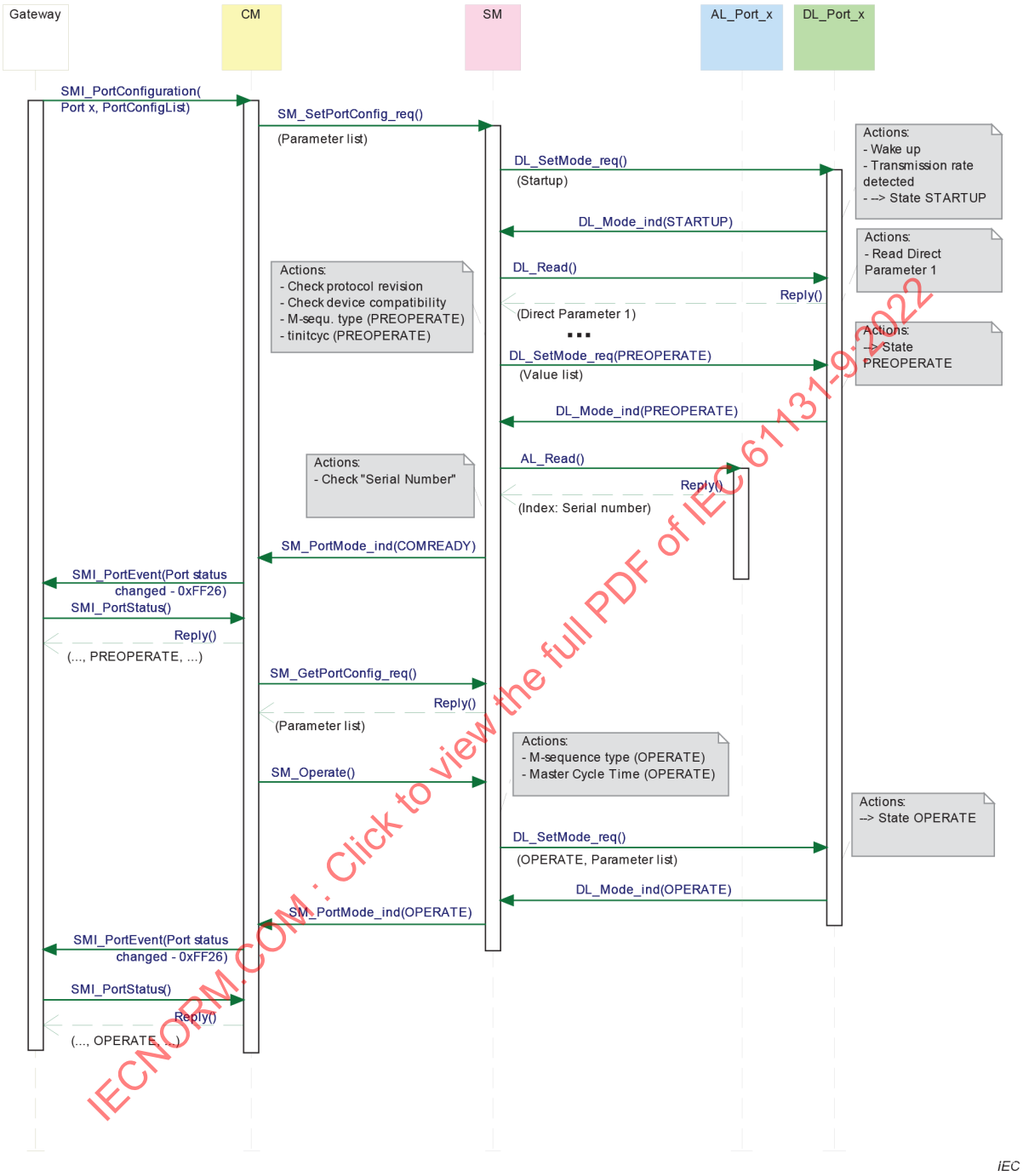
Figure 69 – Structure and services of the Master System Management

Figure 70 demonstrates the actions between the layers Master application (Master App), Configuration Management (CM), System Management (SM), Data Link (DL) and Application Layer (AL) for the startup use case of a particular Port.

This particular use case is characterized by the following statements:

– The Device for the available configuration is connected and inspection is successful.
– The Device uses the correct protocol version according to this document.
– The configured InspectionLevel is "type compatible" (SerialNumber is read out of the Device and not checked).

Dotted arrows in Figure 70 represent response services to an initial service.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 70 – Sequence chart of the use case "Port x setup"

9.2.2 SM Master services

9.2.2.1 Overview

System Management provides the SM Master services to the user via its upper interface. Table 78 lists the assignment of the Master to its role as initiator or receiver for the individual SM services.

Table 78 – SM services within the Master

Service name	Master
SM_SetPortConfig	R
SM_GetPortConfig	R
SM_PortMode	I
SM_Operate	R
Key (see 3.3.4) I Initiator of service R Receiver (Responder) of service	

9.2.2.2 SM_SetPortConfig

The SM_SetPortConfig service is used to set up the requested Device configuration. The parameters of the service primitives are listed in Table 79.

Table 79 – SM_SetPortConfig

Parameter name	.req	.cnf
Argument	M	
ParameterList	M	
Result (+)		S
Port Number		M
Result (−)		S
Port Number		M
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

ParameterList

This parameter contains the configured Port and Device parameters of a Master Port.

Parameter type: Record.

Record Elements:

Port Number

This parameter contains the Port number.

ConfiguredCycleTime

This parameter contains the requested cycle time for the OPERATE mode.

Permitted values:
0 (FreeRunning),
Time (see Table B.3).

TargetMode

This parameter indicates the requested operational mode of the Port.

Permitted values: INACTIVE, DI, DO, CFGCOM, AUTOCOM (see Table 81).

ConfiguredRevisionID (CRID):

Data length: 1 octet for the protocol version (see B.1.5).

InspectionLevel:

Permitted values: NO_CHECK, TYPE_COMP, IDENTICAL (see Table 80).

ConfiguredVendorID (CVID)

Data length: 2 octets.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

NOTE VendorIDs are assigned by the IO-Link community.

ConfiguredDeviceID (CDID)

Data length: 3 octets.

ConfiguredFunctionID (CFID)

Data length: 2 octets.

ConfiguredSerialNumber (CSN)

Data length: up to 16 octets (see Table 80).

Result (+):

This selection parameter indicates that the service has been executed successfully.

Port Number

This parameter contains the Port number.

Result (−):

This selection parameter indicates that the service failed.

Port Number

This parameter contains the Port number.

ErrorInfo

This parameter contains error information.

Permitted values:
PARAMETER_CONFLICT (consistency of parameter set violated).

Table 80 specifies the coding of the different inspection levels (values of the InspectionLevel parameter). See 9.2.3.2 and 11.3.2.

Table 80 – Definition of the InspectionLevel (IL)

Parameter	InspectionLevel (IL)		
	NO_CHECK	TYPE_COMP	IDENTICAL [a]
DeviceID (DID) (compatible)	–	Yes (RDID = CDID)	Yes (RDID = CDID)
VendorID (VID)	–	Yes (RVID = CVID)	Yes (RVID = CVID)
SerialNumber (SN)	–	–	Yes (RSN = CSN)
[a] "IDENTICAL" = optional (not recommended for new developments).			

Table 81 specifies the coding of the different Target Modes.

Table 81 – Definitions of the Target Modes

Target Mode	Definition
CFGCOM	Device communicating in mode CFGCOM after successful inspection
AUTOCOM	Device communicating in mode AUTOCOM without inspection
INACTIVE	Communication disabled, no DI, no DO
DI	Port in digital input mode (SIO)
DO	Port in digital output mode (SIO)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

CFGCOM is a Target Mode based on a user configuration (for example with the help of an IODD) and consistency checking of RID, VID, DID.

AUTOCOM is a Target Mode without configuration. This means that there is no checking of CVID and CDID. The CRID is set to the highest revision the Master is supporting. AUTOCOM should only be selectable together with Inspection Level "NO_CHECK" (see Table 80).

9.2.2.3 SM_GetPortConfig

The SM_GetPortConfig service is used to acquire the real (actual) Device configuration. The parameters of the service primitives are listed in Table 82.

Table 82 – SM_GetPortConfig

Parameter name	.req	.cnf
Argument	M	
Port Number	M	
Result (+)		S(=)
Parameterlist		M
Result (−)		S(=)
Port Number		M
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

Port Number

This parameter contains the Port number.

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ParameterList

This parameter contains the configured Port and Device parameter of a Master Port.

Parameter type: Record

Record Elements:

PortNumber

This parameter contains the Port number.

TargetMode

This parameter indicates the operational mode.

Permitted values: INACTIVE, DI, DO, CFGCOM, AUTOCOM (see Table 81).

RealBaudrate

This parameter indicates the actual transmission rate.

Permitted values:
COM1 (transmission rate of COM1),
COM2 (transmission rate of COM2),
COM3 (transmission rate of COM3).

RealCycleTime

This parameter contains the real (actual) cycle time.

RealRevision (RRID)

Data length: 1 octet for the protocol version (see B.1.5).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

RealVendorID (RVID)

Data length: 2 octets.

NOTE VendorIDs are assigned by the IO-Link community.

RealDeviceID (RDID)

Data length: 3 octets.

RealFunctionID (RFID)

Data length: 2 octets.

RealSerialNumber (RSN)

Data length: up to 16 octets.

Result (−):

This selection parameter indicates that the service failed.

Port Number

This parameter contains the Port number.

ErrorInfo

This parameter contains error information.

Permitted values:
PARAMETER_CONFLICT (consistency of parameter set violated).

All parameters shall be set to "0" if there is no information available.

9.2.2.4 SM_PortMode

The SM_PortMode service is used to indicate changes or faults of the local communication mode. These shall be reported to the Master application. The parameters of the service primitives are listed in Table 83.

Table 83 – SM_PortMode

Parameter name	.ind
Argument	M
Port Number	M
Mode	M

Argument

The service-specific parameters are transmitted in the argument.

Port Number

This parameter contains the Port number.

Mode

Permitted values:

INACTIVE	(Communication disabled, no DI, no DO),
DI	(Port in digital input mode (SIO)),
DO	(Port in digital output mode (SIO)),
COMREADY	(Communication established and inspection successful),
SM_OPERATE	(Port is ready to exchange Process Data),
COMLOST	(Communication failed, new wake-up procedure required),
REVISION_FAULT	(Incompatible protocol revision),
COMP_FAULT	(Incompatible Device or Legacy-Device according to the Inspection Level),
SERNUM_FAULT	(Mismatching SerialNumber according to the InspectionLevel),
CYCTIME_FAULT	(Device does not support the configured cycle time).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

9.2.2.5 SM_Operate

The SM_Operate service prompts System Management to calculate the MasterCycleTime for the Ports if the service is acknowledged positively with Result (+). This service is effective at the indicated Port. The parameters of the service primitives are listed in Table 84.

Table 84 – SM_Operate

Parameter name	.req	.cnf
Argument Port number	M M	
Result (+)		S

Result (−)
  Port Number
  ErrorInfo
		S
		M
		M

Argument

The service-specific parameters are transmitted in the argument.

Port Number

This parameter contains the Port number.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

Port Number

This parameter contains the Port number.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (service unavailable within current state, for example if Port is already in OPERATE state).

9.2.3 SM Master protocol

9.2.3.1 Overview

Owing to the comprehensive configuration, parameterization, and operational features of SDCI, the description of the behavior with the help of state diagrams becomes rather complex. Similar to the DL state machines 9.2.3 uses the possibility of submachines within the main state machines.

Comprehensive compatibility check methods are performed within the submachine states. These methods are indicated by "do *method*" fields within the state graphs, for example in Figure 72.

The corresponding decision logic is demonstrated via activity diagrams (see Figure 73, Figure 74, Figure 75, and Figure 78).

9.2.3.2 SM Master state machine

Figure 71 shows the main state machine of the System Management Master.

Two submachines for the compatibility and serial number check are specified in 9.2.3.3 and 9.2.3.4.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

In case of communication disruption, the System Management is informed via the service DL_Mode (COMLOST).

Only the SM_SetPortConfig service allows reconfiguration of a Port.

The service SM_Operate causes no effect in any state except in state "wait_4".

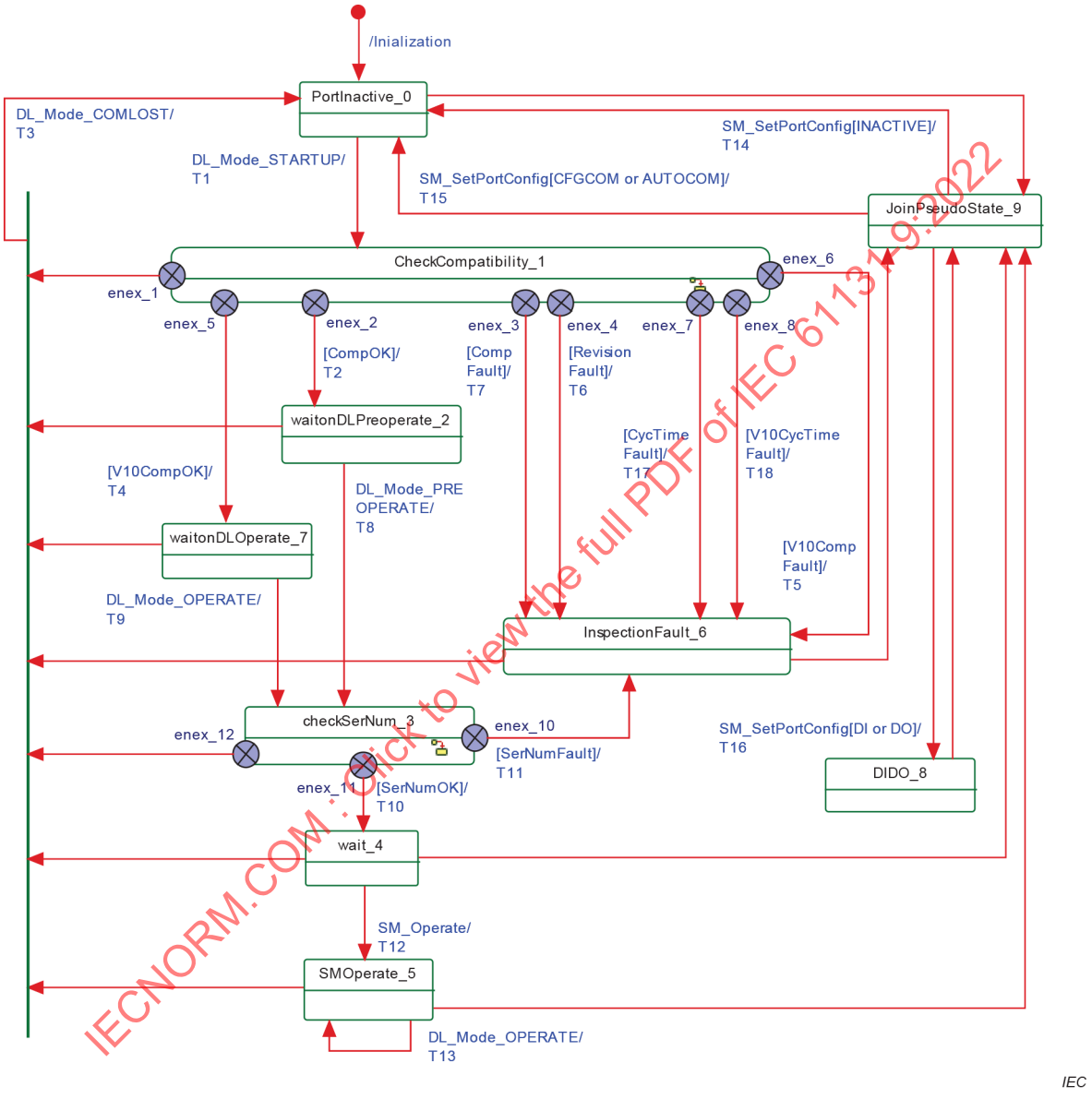


Figure 71 – Main state machine of the Master System Management

Table 85 shows the state transition table of the Master System Management.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 85 – State transition table of the Master System Management

STATE NAME	STATE DESCRIPTION
PortInactive_0	No communication
CheckCompatibility_1	Port is started and revision and Device compatibility is checked. See Figure 72.
waitonDLPreoperate_2	Wait until the PREOPERATE state is established and all the On-Request handlers are started. Port is ready to communicate.
checkSerNum_3	SerialNumber is checked depending on the InspectionLevel (IL). See Figure 77.
wait_4	Port is ready to communicate and waits on service SM_Operate from CM.
SM Operate_5	Port is in state OPERATE and performs cyclic Process Data exchange.
InspectionFault_6	Port is ready to communicate. However, cyclic Process Data exchange cannot be performed due to incompatibilities.
waitonDLOperate_7	Wait on the requested state OPERATE in case the Master is connected to a legacy Device. The SerialNumber can be read thereafter.
DIDO_8	Port will be switched into the DI or DO mode (SIO, no communication).
JoinPseudoState_9	This pseudo state is used instead of a UML join bar. It allows execution of individual SM_SetPortConfig services depending on the system status (INACTIVE, CFGCOM, AUTOCOM, DI, or DO)

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	CompRetry = 0
T2	1	2	DL_SetMode.req (PREOPERATE, ValueList)
T3	1,2,3,4,5, 6,7	0	DL_SetMode.req (INACTIVE) and SM_Mode.ind (COMLOST) due to communication fault
T4	1	7	DL_SetMode.req (OPERATE, ValueList)
T5	1	6	SM_PortMode.ind (COMP_FAULT), DL_SetMode.req (OPERATE, ValueList)
T6	1	6	SM_PortMode.ind (REVISION_FAULT), DL_SetMode.req (PREOPERATE, ValueList)
T7	1	6	SM_PortMode.ind (COMP_FAULT), DL_SetMode.req (PREOPERATE, ValueList)
T8	2	3	–
T9	7	3	–
T10	3	4	SM_PortMode.ind (COMREADY)
T11	3	6	SM_PortMode.ind (SERNUM_FAULT)
T12	4	5	DL_SetMode.req (OPERATE, ValueList)
T13	5	5	–
T14	0,4,5,6,8	0	SM_PortMode.ind (INACTIVE), DL_SetMode.req (INACTIVE)
T15	0,4,5,6,8	0	DL_SetMode.req (STARTUP, ValueList), PL_SetMode.req (SDCI)
T16	0,4,5,6,8	8	PL_SetMode.req (SIO), SM_Mode.ind (DI or DO), DL_SetMode.req (INACTIVE)
T17	1	6	SM_PortMode.ind (CYCTIME_FAULT), DL_SetMode.req (PREOPERATE, ValueList)
T18	1	6	SM_PortMode.ind (CYCTIME_FAULT), DL_SetMode.req (OPERATE, ValueList), ValueList.M-sequenceTime = MinCycleTime of Device

INTERNAL ITEMS	TYPE	DEFINITION
CompOK	Bool	See Figure 75
CompFault	Bool	See Figure 75; error variable COMP_FAULT
CycTimeFault	Bool	See Figure 75; error variable CYCTIME_FAULT
RevisionFault	Bool	See Figure 73; error variable REVISION_FAULT

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

INTERNAL ITEMS	TYPE	DEFINITION
SerNumFault	Bool	See Figure 78; error variable SERNUM_FAULT
SerNumOK	Bool	See Figure 78
V10CompFault	Bool	See Figure 74; error variable COMP_FAULT
V10CompOK	Bool	See Figure 74
V10CycTimeFault	Bool	See Figure 74; error variable CYCTIME_FAULT
INACTIVE	Variable	A target mode in service SM_SetPortConfig
CFGCOM, AUTOCOM	Variables	Target Modes in service SM_SetPortConfig

9.2.3.3 SM Master submachine "Check Compatibility"

Figure 72 shows the SM Master submachine checkCompatibility_1.

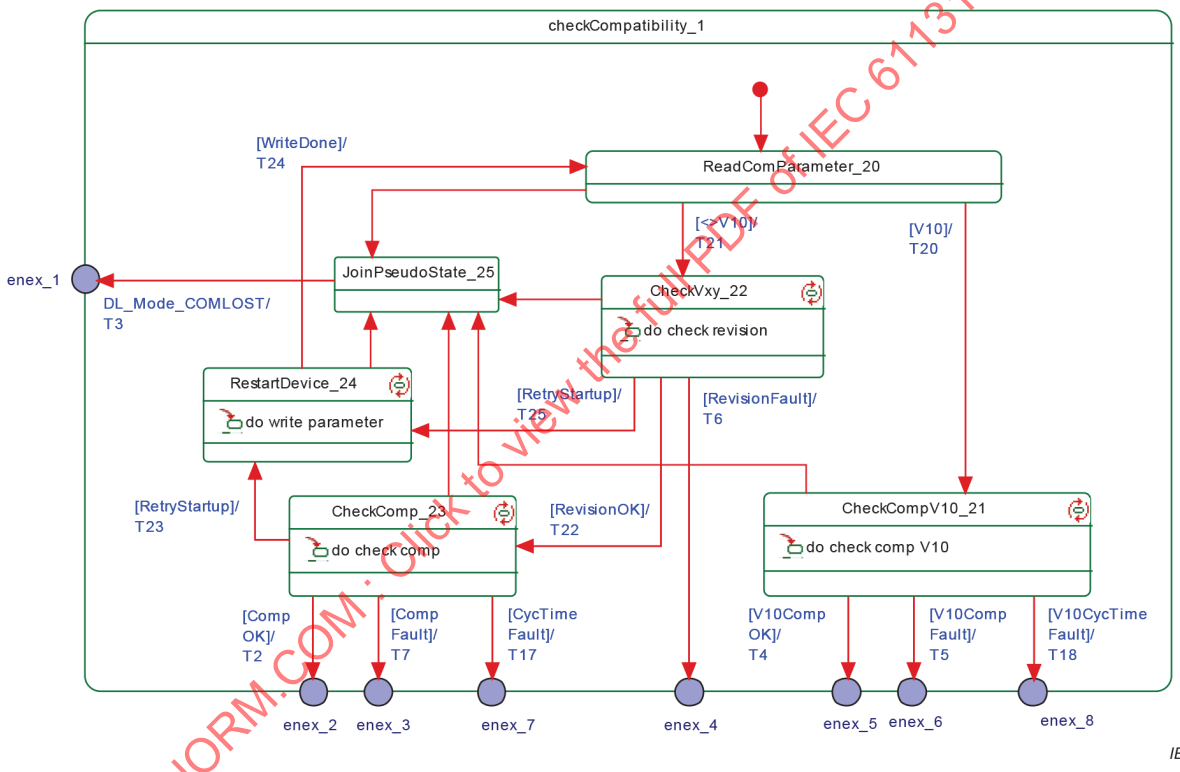


Figure 72 – SM Master submachine CheckCompatibility_1

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 86 shows the state transition table of the Master submachine checkCompatibility_1.

Table 86 – State transition table of the Master submachine CheckCompatibility_1

STATE NAME	STATE DESCRIPTION
ReadComParameter_20	Acquires communication parameters from Direct Parameter Page 1 (0x02 to 0x06) via service DL_Read (see Table B.1).
CheckCompV10_21	Acquires identification parameters from Direct Parameter Page 1 (0x07 to 0x0D) via service DL_Read (see Table B.1). The configured InspectionLevel (IL) defines the decision logic of the subsequent compatibility check "CheckCompV10" with parameters RVID, RDID, and RFID according to Figure 74.
CheckVxy_22	A check is performed to verify whether the configured revision (CRID) matches the real (actual) revision (RRID) according to Figure 73.
CheckComp_23	Acquires identification parameters from Direct Parameter Page 1 (0x07 to 0x0D) via service DL_Read (see Table B.1). The configured InspectionLevel (IL) defines the decision logic of the subsequent compatibility check "CheckComp" according to Figure 75.
RestartDevice_24	Writes the compatibility parameters configured protocol revision (CRID) and configured DeviceID (CDID) into the Device depending on the Target Mode of communication CFGCOM or AUTOCOM (see Table 81) according to Figure 76.
JoinPseudoState_25	This pseudo state is used instead of a UML join bar. No guards involved.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T20	20	21	–
T21	20	22	DL_Write (0x00, MCmd_MASTERIDENT), see Table B.2
T22	22	23	–
T23	23	24	–
T24	24	20	–
T25	22	24	CompRetry = CompRetry +1

INTERNAL ITEMS	TYPE	DEFINITION
CompOK	Bool	See Figure 75
CompFault	Bool	See Figure 75; error variable COMP_FAULT
RevisionFault	Bool	See Figure 73; error variable REVISION_FAULT
RevisionOK	Bool	See Figure 73
SerNumFault	Bool	See Figure 78; error variable SERNUM_FAULT
SerNumOK	Bool	See Figure 78
V10	Bool	Real protocol revision of connected Device is a legacy version (V1.0, see B.1.5)
<>V10	Bool	Real protocol revision of connected Device is in accordance with this document
V10CompFault	Bool	See Figure 74; error variable COMP_FAULT
V10CompOK	Bool	See Figure 74
RetryStartup	Bool	See Figure 73 and Figure 75
CompRetry	Variable	Internal counter
WriteDone	Bool	Finalization of the restart service sequence
MCmd_XXXXXXX	Call	See Table 45

Some states contain complex logic to deal with the compatibility and validity checks. Figure 73 to Figure 76 demonstrate the context.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 73 shows the decision logic for the protocol revision check in state "CheckVxy". In case of configured Devices, the following rule applies: if the configured revision (CRID) and the real revision (RRID) do not match, the CRID will be transmitted to the Device. If the Device does not accept, the Master returns an indication via the SM_Mode service with REV_FAULT.

In case of not configured Devices, the operational mode AUTOCOM shall be used. See 9.2.2.2 and 9.2.2.3 for the parameter name abbreviations.

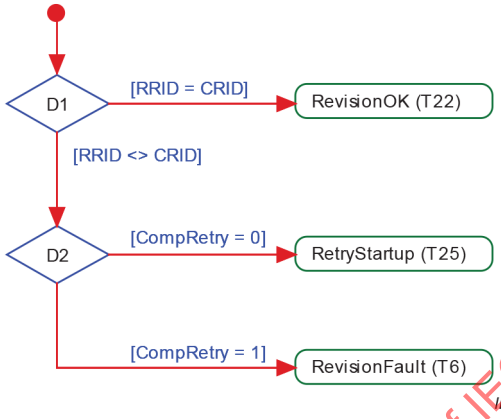
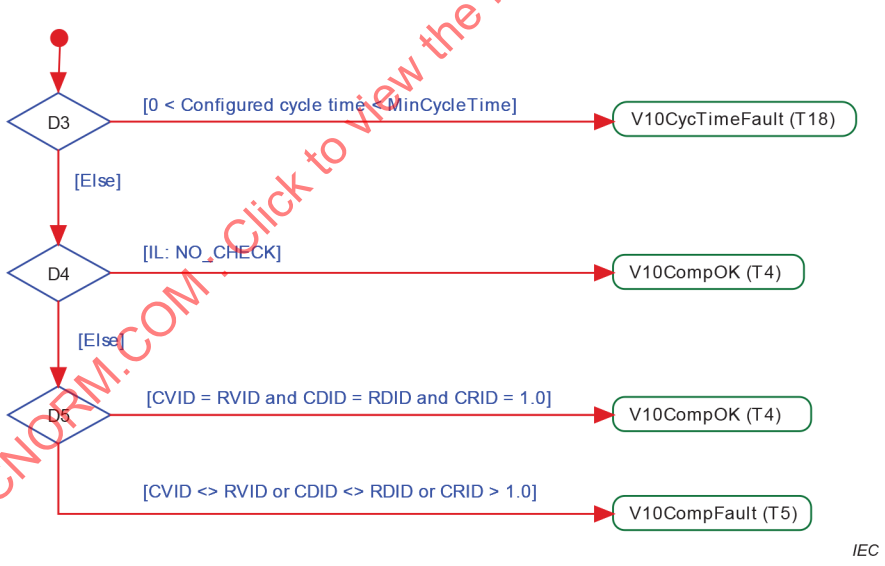


Figure 73 – Activity for state "CheckVxy"

Figure 74 shows the decision logic for the legacy compatibility check in state "CheckCompV10".



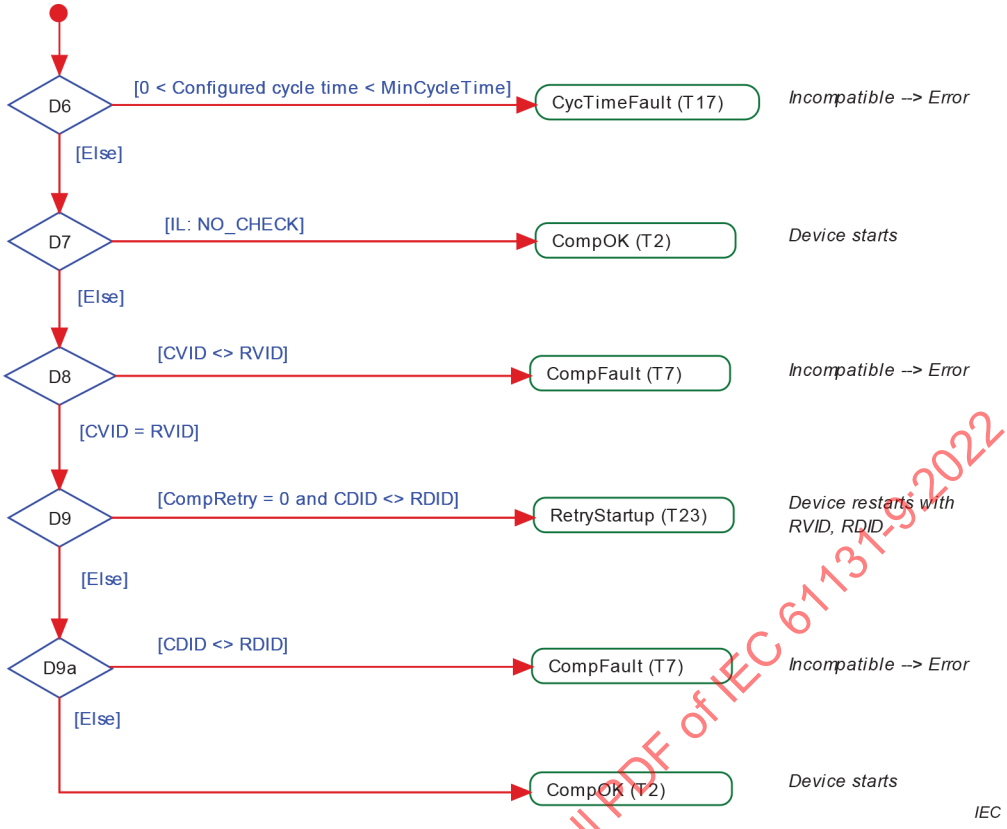
Key

IL = Inspection Level

Figure 74 – Activity for state "CheckCompV10"

Figure 75 shows the decision logic for the compatibility check in state "CheckComp".

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022



Key

IL = Inspection Level

Figure 75 – Activity for state "CheckComp"

Figure 76 shows the activity (write parameter) in state "RestartDevice".

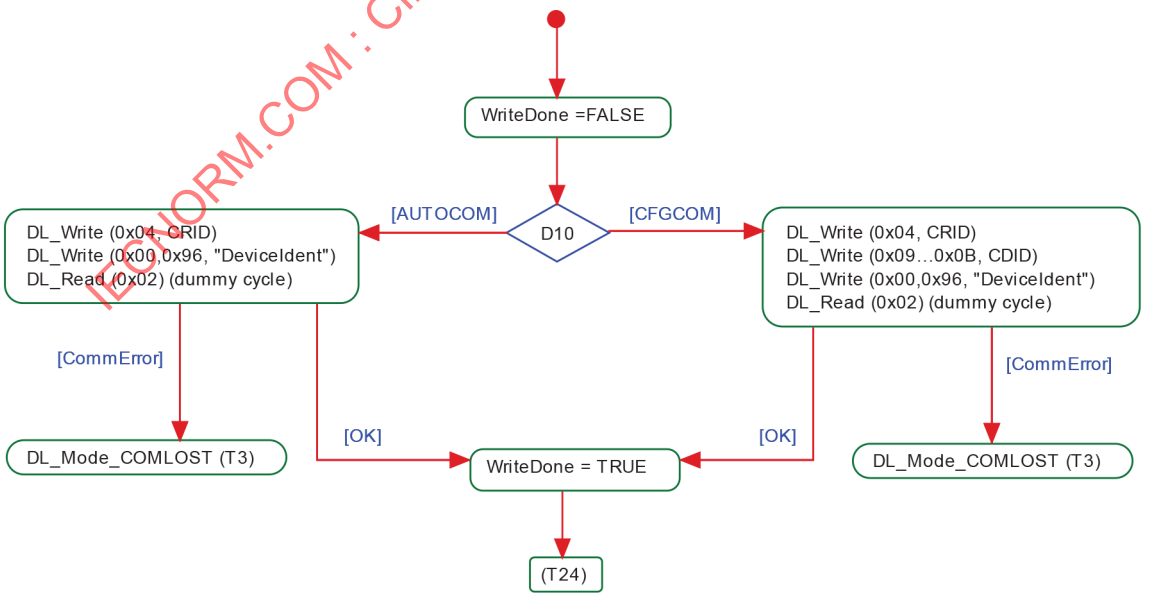


Figure 76 – Activity (write parameter) in state "RestartDevice"

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

9.2.3.4 SM Master submachine "Check serial number"

Figure 77 shows the SM Master submachine "checkSerNum_3". State CheckSerNum_31 can be skipped (option).

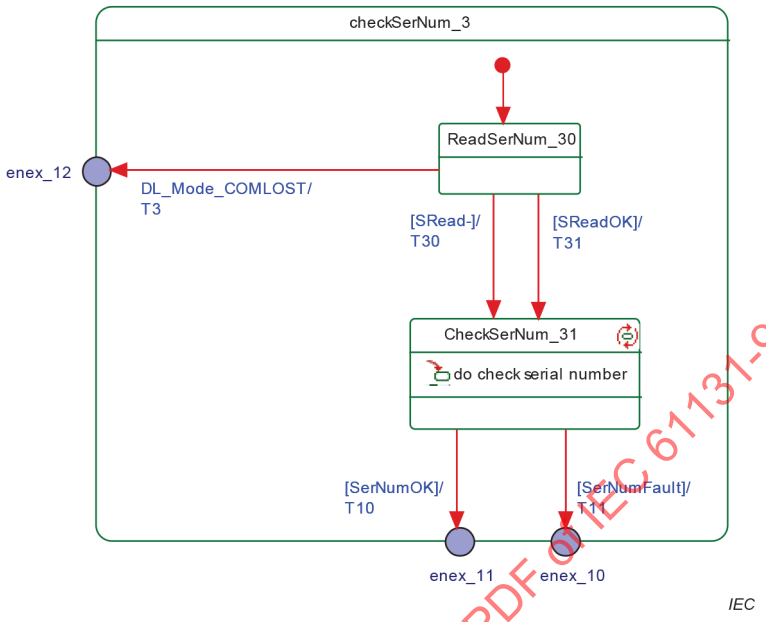


Figure 77 – SM Master submachine checkSerNum_3

Table 87 shows the state transition table of the Master submachine checkSerNum_3

Table 87 – State transition table of the Master submachine checkSerNum_3

STATE NAME		STATE DESCRIPTION	
ReadSerNum_30		Acquires the SerialNumber from the Device via AL_Read.req (Index: 0x0015). A positive response (AL_Read(+)) leads to SReadOK = true. A negative response (AL_Read(-)) leads to SRead- = true.	
CheckSerNum_31		Optional: SerialNumber checking skipped or checked correctly.	
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T30	40	41	–
T31	40	41	–
INTERNAL ITEMS		**TYPE**	**DEFINITION**
SRead-		Bool	Negative response of service AL_Read (Index 0x0015)
SReadOK		Bool	SerialNumber read correctly
SerNumOK		Bool	See Figure 78
SerNumFault		Bool	See Figure 78

Figure 78 shows the decision logic (activity) for the state CheckSerNum_31.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

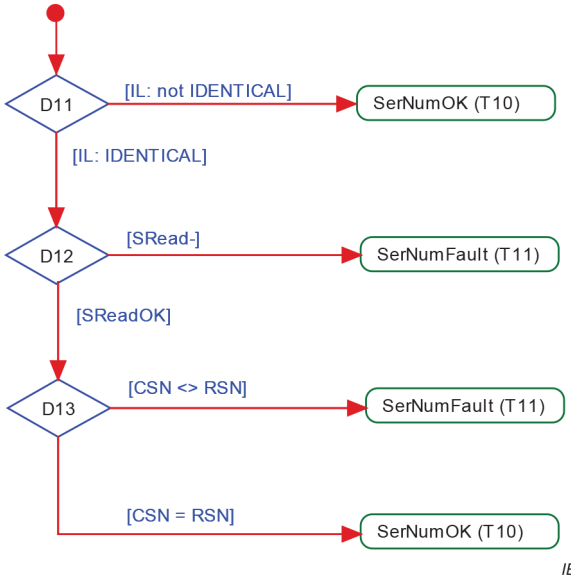


Figure 78 – Activity (check SerialNumber) for state CheckSerNum_31

9.2.3.5 Rules for the usage of M-sequence types

The System Management is responsible for setting up the correct M-sequence types. This occurs after the check compatibility actions (transition to PREOPERATE) and before the transition to OPERATE.

Different M-sequence types shall be used within the different operational states (see A.2.6). For example, when switching to the OPERATE state the M-sequence type relevant for cyclic operation shall be used. The M-sequence type to be used in operational state OPERATE is determined by the size of the input and output Process Data. The available M-sequence types in the three modes STARTUP, PREOPERATE, and OPERATE and the corresponding coding of the parameter M-sequenceCapability are specified in A.2.6. The input and output data formats shall be acquired from the connected Device in order to adjust the M-sequence type. It is mandatory for a Master to implement all the specified M-sequence types in A.2.6.

9.3 System Management of the Device

9.3.1 Overview

Figure 79 provides an overview of the structure and services of the Device System Management.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

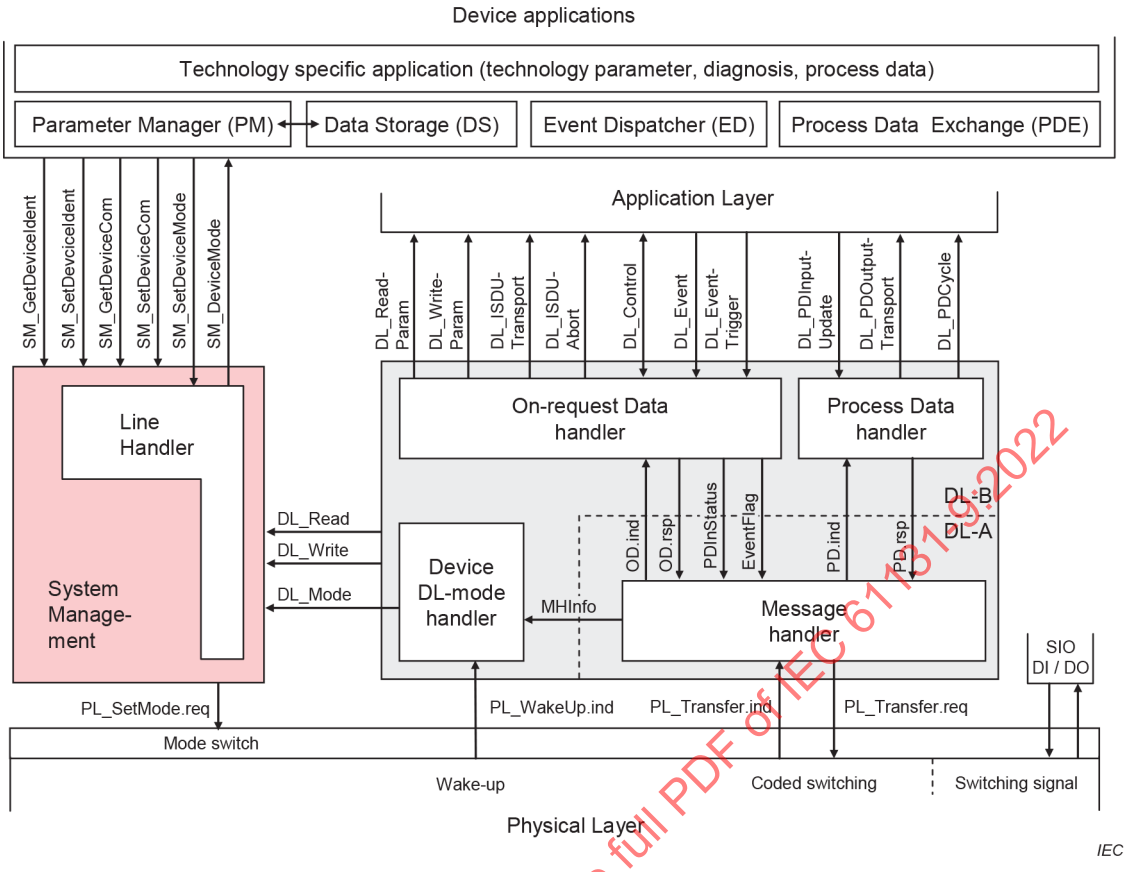


Figure 79 – Structure and services of the System Management (Device)

The System Management (SM) of the Device provides the central controlling instance via the line handler through all the phases of initialization, default state (SIO), communication startup, communication, and fallback to SIO mode.

The Device SM interacts with the PL to establish the necessary line driver and receiver adjustments (see Figure 16), with the DL to get the necessary information from the Master (wake-up, transmission rates, etc.) and with the Device applications to ensure the Device identity and compatibility (identification parameters).

The transitions between the line handler states (see Figure 81) are initiated by the Master Port activities (wake-up and communication) and triggered through the Device Data Link Layer via the DL_Mode indications and DL_Write requests (commands).

The SM provides the Device identification parameters through the Device applications interface.

The sequence chart in Figure 80 demonstrates a typical Device sequence from initialization to default SIO mode and via wake-up request from the Master to final communication. The sequence chart is complemented by the use case of a communication error such as T_{DSIO} expired, or communication fault, or a request from Master such as Fallback (caused by Event).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

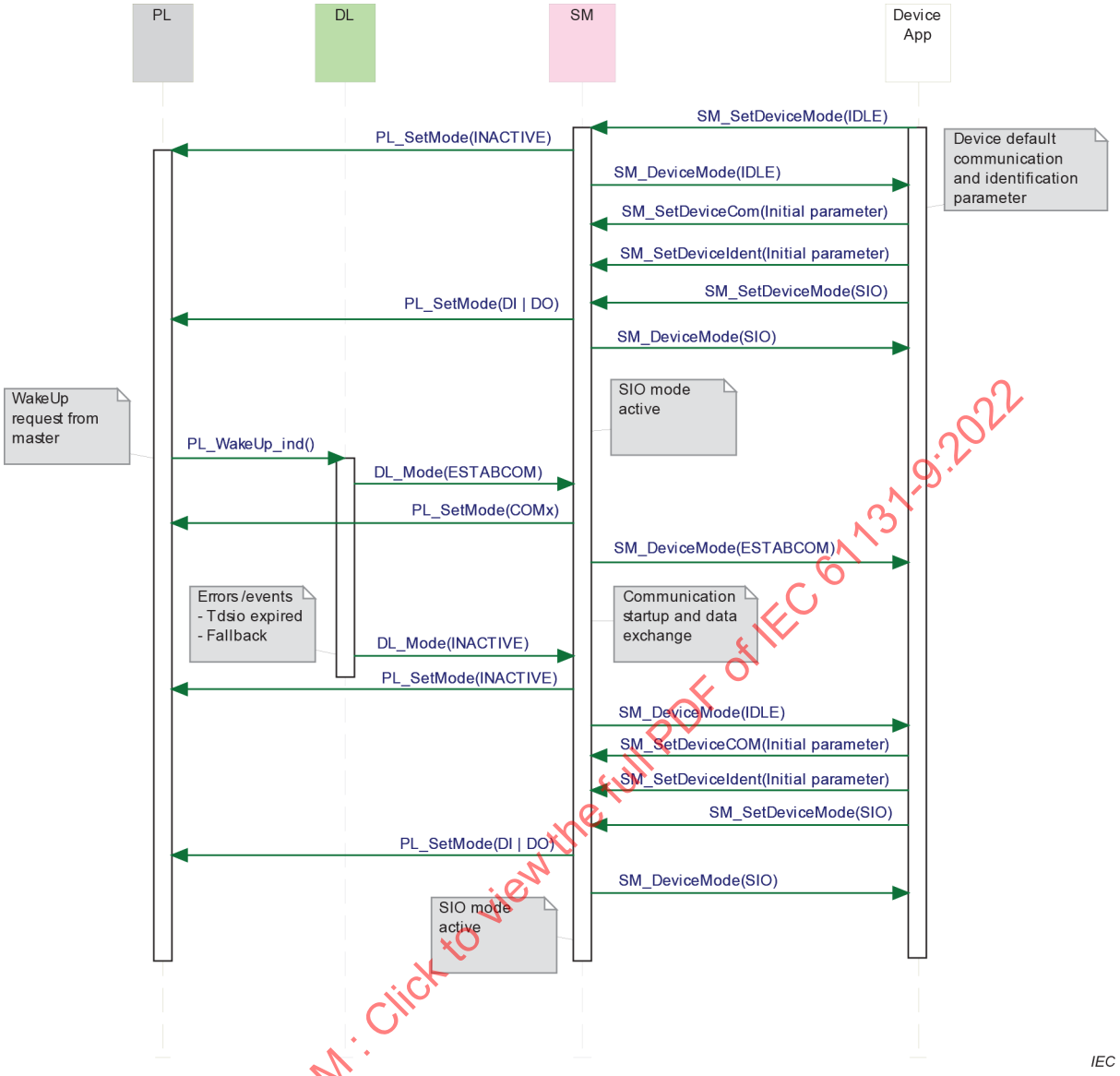


Figure 80 – Sequence chart of the use case "INACTIVE – SIO – SDCI – SIO"

The SM services shown in Figure 80 are specified in 9.3.2.

9.3.2 SM Device services

9.3.2.1 Overview

Subclause 9.3.2 describes the services the Device System Management provides to its applications as shown in Figure 79.

Table 88 lists the assignment of the Device to its role as initiator or receiver for the individual System Management service.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 88 – SM services within the Device

Service name	Device
SM_SetDeviceCom	R
SM_GetDeviceCom	R
SM_SetDeviceIdent	R
SM_GetDeviceIdent	R
SM_SetDeviceMode	R
SM_DeviceMode	I
Key (see 3.3.4) I Initiator of service R Receiver (Responder) of service	

9.3.2.2 SM_SetDeviceCom

The SM_SetDeviceCom service is used to configure the communication properties supported by the Device in the System Management. The parameters of the service primitives are listed in Table 89.

Table 89 – SM_SetDeviceCom

Parameter name	.req	.cnf
Argument ParameterList	M M	
Result (+)		S
Result (−) ErrorInfo		S M

Argument

The service-specific parameters are transmitted in the argument.

ParameterList

This parameter contains the configured communication parameters for a Device.

Parameter type: Record.

Record Elements:

SupportedSIOMode

This parameter indicates the SIO mode supported by the Device.

Permitted values:
INACTIVE (C/Q line in high impedance),
DI (C/Q line in digital input mode),
DO (C/Q line in digital output mode).

SupportedTransmissionrate

This parameter indicates the transmission rate supported by the Device.

Permitted values:
COM1 (transmission rate of COM1),
COM2 (transmission rate of COM2),
COM3 (transmission rate of COM3).

MinCycleTime

This parameter contains the minimum cycle time supported by the Device (see B.1.3).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

M-sequence Capability

This parameter indicates the capabilities supported by the Device (see B.1.4):
– ISDU support,
– OPERATE M-sequence types,
– PREOPERATE M-sequence types.

RevisionID (RID)

This parameter contains the protocol revision (see B.1.5) supported by the Device.

ProcessDataIn

This parameter contains the length of PD to be sent to the Master (see B.1.6).

ProcessDataOut

This parameter contains the length of PD to be sent by the Master (see B.1.7).

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
PARAMETER_CONFLICT (consistency of parameter set violated).

9.3.2.3 SM_GetDeviceCom

The SM_GetDeviceCom service is used to read the current communication properties from the System Management. The parameters of the service primitives are listed in Table 90.

Table 90 – SM_GetDeviceCom

Parameter name	.req	.cnf
Argument	M	
Result (+)		S
ParameterList		M
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

Result (+):

This selection parameter indicates that the service has been executed successfully.

ParameterList

This parameter contains the configured communication parameter for a Device.

Parameter type: Record.

Record Elements:

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

CurrentMode

This parameter indicates the current SIO or Communication Mode by the Device.

Permitted values:
INACTIVE (C/Q line in high impedance),
DI (C/Q line in digital input mode),
DO (C/Q line in digital output mode),
COM1 (transmission rate of COM1),
COM2 (transmission rate of COM2),
COM3 (transmission rate of COM3).

MasterCycleTime

This parameter contains the MasterCycleTime to be set by the Master System Management (see B.1.3). This parameter is only valid in the state SM_Operate.

M-sequence Capability

This parameter indicates the current M-sequence capabilities configured in the System Management of the Device (see B.1.4):
- ISDU support,
- OPERATE M-sequence types,
- PREOPERATE M-sequence types.

RevisionID (RID)

This parameter contains the current protocol revision (see B.1.5) within the System Management of the Device.

ProcessDataIn

This parameter contains the current length of PD to be sent to the Master (see B.1.6).

ProcessDataOut

This parameter contains the current length of PD to be sent by the Master (see B.1.7).

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (service unavailable within current state).

9.3.2.4 SM_SetDeviceIdent

The SM_SetDeviceIdent service is used to configure the Device identification data in the System Management. The parameters of the service primitives are listed in Table 91.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 91 – SM_SetDeviceIdent

Parameter name	.req	.cnf
Argument	M	
ParameterList	M	
Result (+)		S
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

ParameterList

This parameter contains the configured identification parameter for a Device.

Parameter type: Record.

Record Elements:

VendorID (VID)

This parameter contains the VendorID assigned to a Device (see B.1.8).

Data length: 2 octets.

DeviceID (DID)

This parameter contains one of the assigned DeviceIDs (see B.1.9).

Data length: 3 octets.

FunctionID (FID)

This parameter contains one of the assigned FunctionIDs (see B.1.10).

Data length: 2 octets.

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (service unavailable within current state),
PARAMETER_CONFLICT (consistency of parameter set violated).

9.3.2.5 SM_GetDeviceIdent

The SM_GetDeviceIdent service is used to read the Device identification parameter from the System Management. The parameters of the service primitives are listed in Table 92.

Table 92 – SM_GetDeviceIdent

Parameter name	.req	.cnf
Argument	M	
Result (+)		S
ParameterList		M
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

Result (+):

This selection parameter indicates that the service has been executed successfully.

ParameterList

This parameter contains the configured communication parameters of the Device.

Parameter type: Record.

Record Elements:

VendorID (VID)

This parameter contains the actual VendorID of the Device (see B.1.8).

Data length: 2 octets.

DeviceID (DID)

This parameter contains the actual DeviceID of the Device (see B.1.9).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Data length: 3 octets.

FunctionID (FID)

This parameter contains the actual FunctionID of the Device (see B.1.10).

Data length: 2 octets.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (service unavailable within current state).

9.3.2.6 SM_SetDeviceMode

The SM_SetDeviceMode service is used to set the Device into a defined operational state during initialization. The parameters of the service primitives are listed in Table 93.

Table 93 – SM_SetDeviceMode

Parameter name	.req	.cnf
Argument	M	
Mode	M	
Result (+)		S
Result (−)		S
ErrorInfo		M

Argument

The service-specific parameters are transmitted in the argument.

Mode

Permitted values:
IDLE (Device changes to waiting for configuration),
SIO (Device changes to the mode defined in service "SM_SetDeviceCom").

Result (+):

This selection parameter indicates that the service has been executed successfully.

Result (−):

This selection parameter indicates that the service failed.

ErrorInfo

This parameter contains error information.

Permitted values:
STATE_CONFLICT (service unavailable within current state).

9.3.2.7 SM_DeviceMode

The SM_DeviceMode service is used to indicate changes of communication states to the Device application. The parameters of the service primitives are listed in Table 94.

Table 94 – SM_DeviceMode

Parameter name	.ind
Argument	M
Mode	M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The service-specific parameters are transmitted in the argument.

Mode

Permitted values:

IDLE	(Device changed to waiting for configuration),
SIO	(Device changed to the mode defined in service "SM_SetDeviceCom"),
ESTABCOM	(Device changed to the SM mode "SM_ComEstablish"),
COM1	(Device changed to the COM1 mode),
COM2	(Device changed to the COM2 mode),
COM3	(Device changed to the COM3 mode),
STARTUP	(Device changed to the STARTUP mode),
IDENT_STARTUP	(Device changed to the SM mode "SM_IdentStartup"),
IDENT_CHANGE	(Device changed to the SM mode "SM_IdentCheck"),
PREOPERATE	(Device changed to the PREOPERATE mode),
OPERATE	(Device changed to the OPERATE mode).

9.3.3 SM Device protocol

9.3.3.1 Overview

The behavior of the Device is mainly driven by Master messages.

9.3.3.2 SM Device state machine

Figure 81 shows the SM line handler state machine of the Device. It is triggered by the DL_Mode handler and the Device application. It evaluates the different communication phases during startup and controls the line state of the Device.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

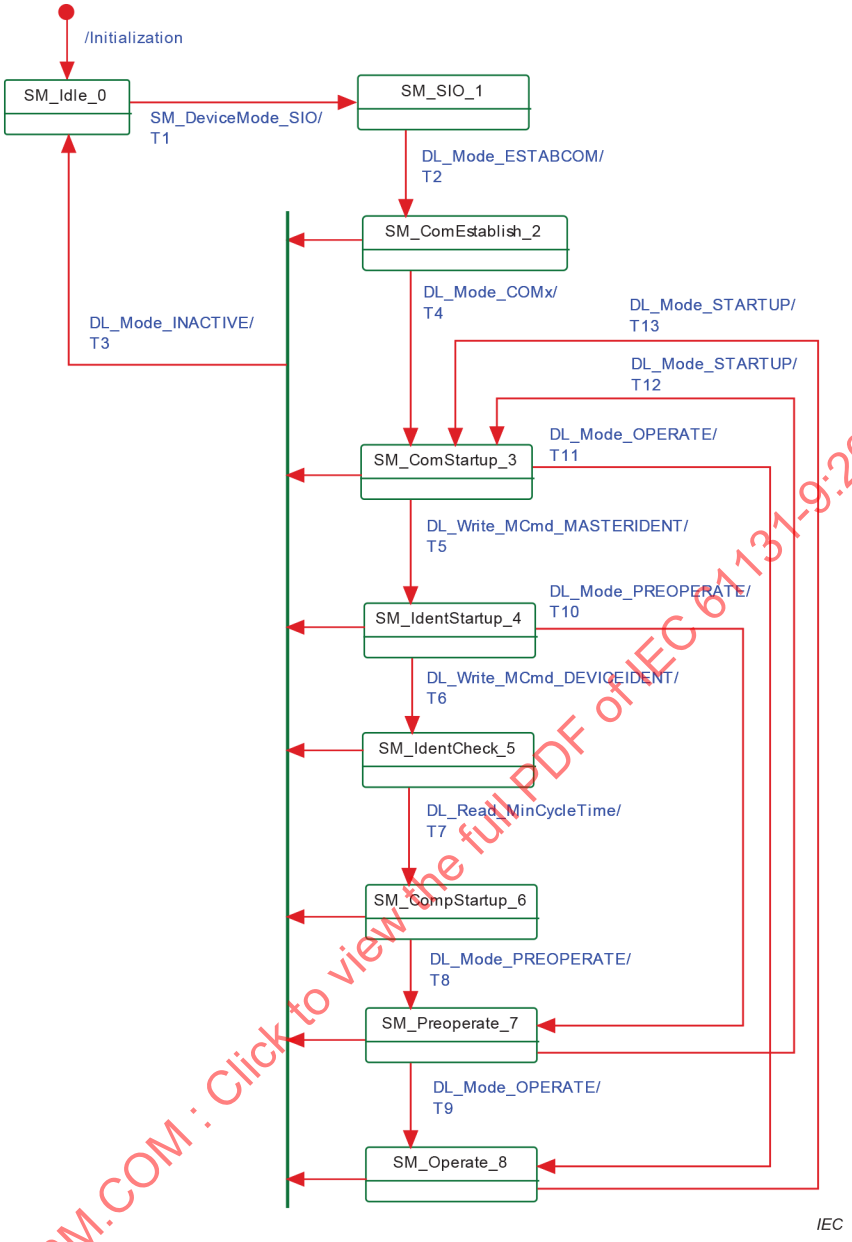


Figure 81 – State machine of the Device System Management

Table 95 specifies the individual states and the actions within the transitions.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 95 – State transition table of the Device System Management

STATE NAME	STATE DESCRIPTION
SM_Idle_0	In SM_Idle the SM is waiting for configuration by the Device application and to be set to SIO mode. The state is left on receiving a SM_SetDeviceMode(SIO) request from the Device application The following sequence of services shall be executed between Device application and SM. Invoke SM_SetDeviceCom(initial parameter list) Invoke SM_SetDeviceIdent(VID, initial DID, FID)
SM_SIO_1	In SM_SIO the SM line handler is remaining in the default SIO mode. The Physical Layer is set to the SIO mode characteristics defined by the Device application via the SetDeviceMode service. The state is left on receiving a DL_Mode(ESTABCOM) indication.
SM_ComEstablish_2	In SM_ComEstablish the SM is waiting for the communication to be established in the Data Link Layer. The state is left on receiving a DL_Mode(INACTIVE) or a DL_Mode(COMx) indication, where COMx may be any of COM1, COM2 or COM3.
SM_ComStartup_3	In SM_ComStartup the communication parameter (Direct Parameter page 1, addresses 0x02 to 0x06) are read by the Master SM via DL_Read requests. The state is left upon reception of a DL_Mode(INACTIVE), a DL_Mode(OPERATE) indication (legacy Master only), or a DL_Write(MCmd_MASTERIDENT) request (Master in accordance with this document).
SM_IdentStartup_4	In SM_IdentStartup the identification data (VID, DID, FID) are read and verified by the Master. In case of incompatibilities the Master SM writes the supported SDCI Revision (RID) and configured DeviceID (DID) to the Device. The state is left upon reception of a DL_Mode(INACTIVE), a DL_Mode(PREOPERATE) indication (compatibility check passed), or a DL_Write(MCmd_DEVICEIDENT) request (new compatibility requested).
SM_IdentCheck_5	In SM_IdentCheck the SM waits for new initialization of communication and identification parameters. The state is left on receiving a DL_Mode(INACTIVE) indication or a DL_Read(Direct Parameter page 1, addresses 0x02 = "MinCycleTime") request. Within this state the Device application shall check the RID and DID parameters from the SM and set these data to the supported values. Therefore the following sequence of services shall be executed between Device application and SM. Invoke SM_GetDeviceCom(configured RID, parameter list) Invoke SM_GetDeviceIdent(configured DID, parameter list) Invoke Device application checks and provides compatibility function and parameters Invoke SM_SetDeviceCom(new supported RID, new parameter list) Invoke SM_SetDeviceIdent(new supported DID, parameter list)
SM_CompStartup_6	In SM_CompatStartup the communication and identification data are reread and verified by the Master SM. The state is left on receiving a DL_Mode(INACTIVE) or a DL_Mode(PREOPERATE) indication.
SM_Preoperate_7	During SM_Preoperate the SerialNumber can be read and verified by the Master SM, as well as Data Storage and Device parameterization can be executed. The state is left on receiving a DL_Mode(INACTIVE), a DL_Mode(STARTUP) or a DL_Mode(OPERATE) indication.
SM_Operate_8	During SM_Operate the cyclic Process Data exchange and acyclic On-request Data transfer are active. The state is left on receiving a DL_Mode(INACTIVE) or a DL_Mode(STARTUP) indication.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	The Device is switched to the configured SIO mode by receiving the trigger SM_SetDeviceMode.req(SIO). Invoke PL_SetMode(DI|DO|INACTIVE) Invoke SM_DeviceMode(SIO)
T2	1	2	The Device is switched to the communication mode by receiving the trigger DL_Mode.ind(ESTABCOM). Invoke PL_SetMode(COMx) Invoke SM_DeviceMode(ESTABCOM)
T3	2,3,4,5,6, 7,8	0	The Device is switched to SM_Idle mode by receiving the trigger DL_Mode.ind(INACTIVE) . Invoke PL_SetMode(INACTIVE) Invoke SM_DeviceMode(IDLE)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T4	2	3	The Device application receives an indication on the baud rate with which the communication has been established in the DL triggered by DL_Mode.ind(COMx). Invoke SM_DeviceMode(COMx)
T5	3	4	The Device identification phase is entered by receiving the trigger DL_Write.ind(MCmd_MASTERIDENT). Invoke SM_DeviceMode(IDENTSTARTUP)
T6	4	5	The Device identity check phase is entered by receiving the trigger DL_Write.ind(MCmd_DEVICEIDENT). Invoke SM_DeviceMode(IDENTCHANGE)
T7	5	6	The Device compatibility startup phase is entered by receiving the trigger DL_Read.ind(Direct Parameter page 1, address 0x02 = "MinCycleTime").
T8	6	7	The Device's preoperate phase is entered by receiving the trigger DL_Mode.ind(PREOPERATE). Invoke SM_DeviceMode(PREOPERATE)
T9	7	8	The Device's operate phase is entered by receiving the trigger DL_Mode.ind(OPERATE). Invoke SM_DeviceMode(OPERATE)
T10	4	7	The Device's preoperate phase is entered by receiving the trigger DL_Mode.ind(PREOPERATE). Invoke SM_DeviceMode(PREOPERATE)
T11	3	8	The Device's operate phase is entered by receiving the trigger DL_Mode.ind(OPERATE). Invoke SM_DeviceMode(OPERATE)
T12	7	3	The Device's communication startup phase is entered by receiving the trigger DL_Mode.ind(STARTUP). Invoke SM_DeviceMode(STARTUP)
T13	8	3	The Device's communication startup phase is entered by receiving the trigger DL_Mode.ind(STARTUP). Invoke SM_DeviceMode(STARTUP)

INTERNAL ITEMS	TYPE	DEFINITION
COMx	Variable	Any of COM1, COM2, or COM3 transmission rates
DL_Write_MCmd_xxx	Service	DL Service writes MasterCommands (xxx = values out of Table B.2)

Figure 82 shows a typical sequence chart for the SM communication startup of a Device matching the Master Port configuration settings (regular startup).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

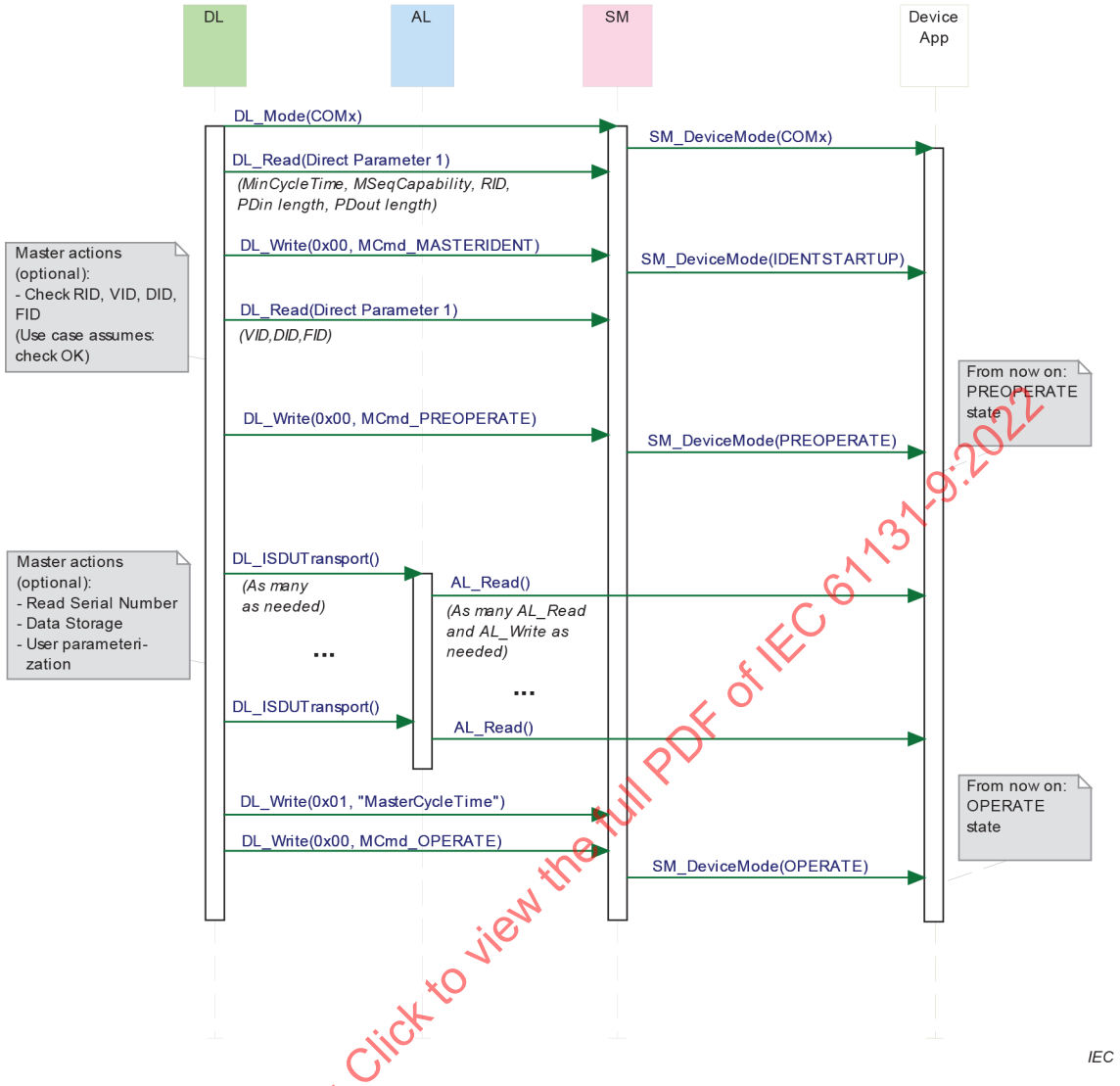


Figure 82 – Sequence chart of a regular Device startup

Figure 83 shows a typical sequence chart for the SM communication startup of a Device not matching the Master Port configuration settings (compatibility mode). In this mode, the Master tries to overwrite the Device's identification parameters to achieve a compatible and a workable mode.

The sequence chart in Figure 83 shows only the actions until the PREOPERATE state. The remaining actions until the OPERATE state can be taken from Figure 82.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

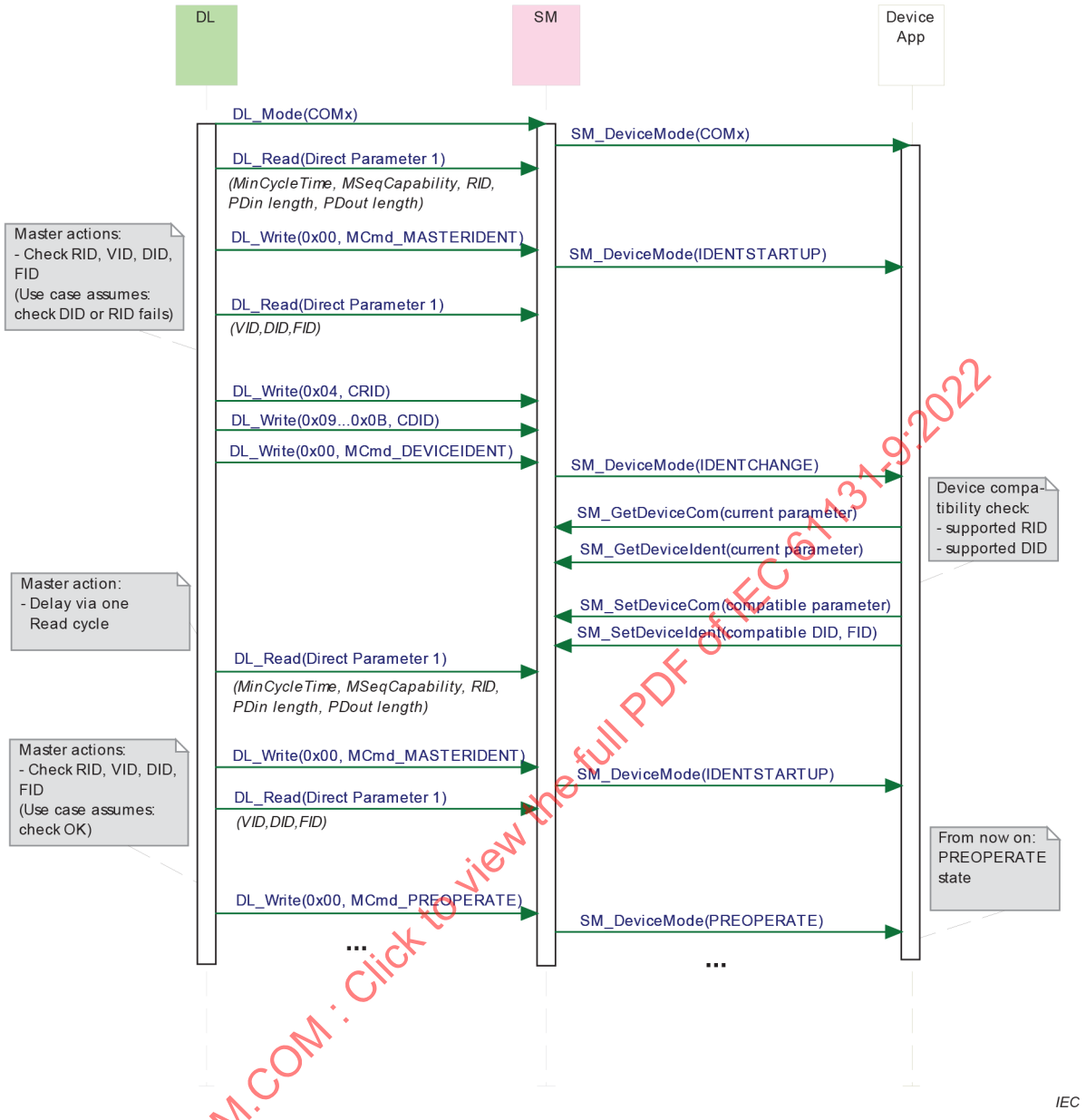


Figure 83 – Sequence chart of a Device startup in compatibility mode

Figure 84 shows a typical sequence chart for the SM communication startup of a Device not matching the Master Port configuration settings. The System Management of the Master tries to reconfigure the Device with alternative Device identification parameters (compatibility mode). In this use case, the alternative parameters are assumed to be incompatible.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

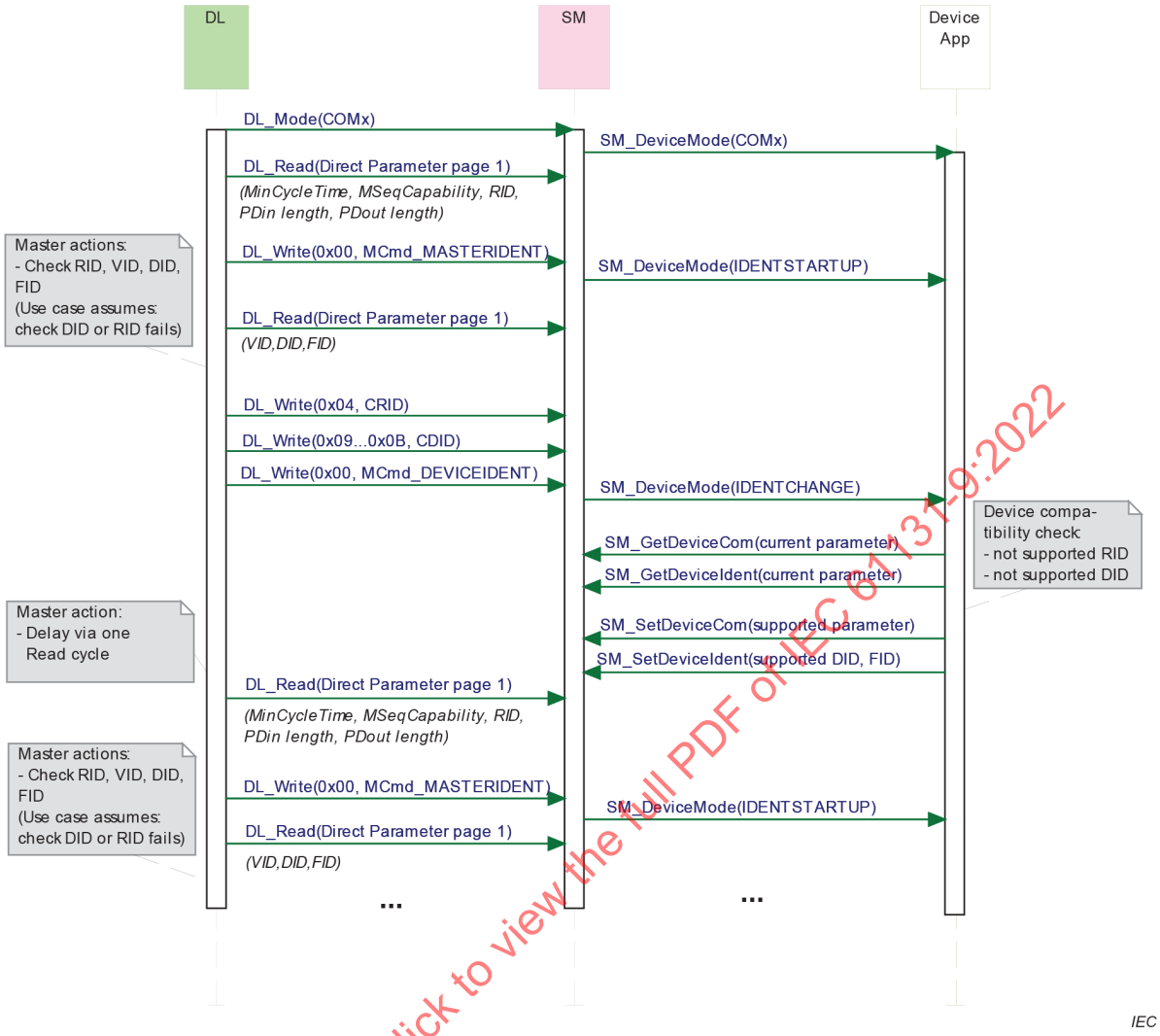


Figure 84 – Sequence chart of a Device startup when compatibility fails

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10 Device

10.1 Overview

Figure 85 provides an overview of the complete structure and services of a Device.

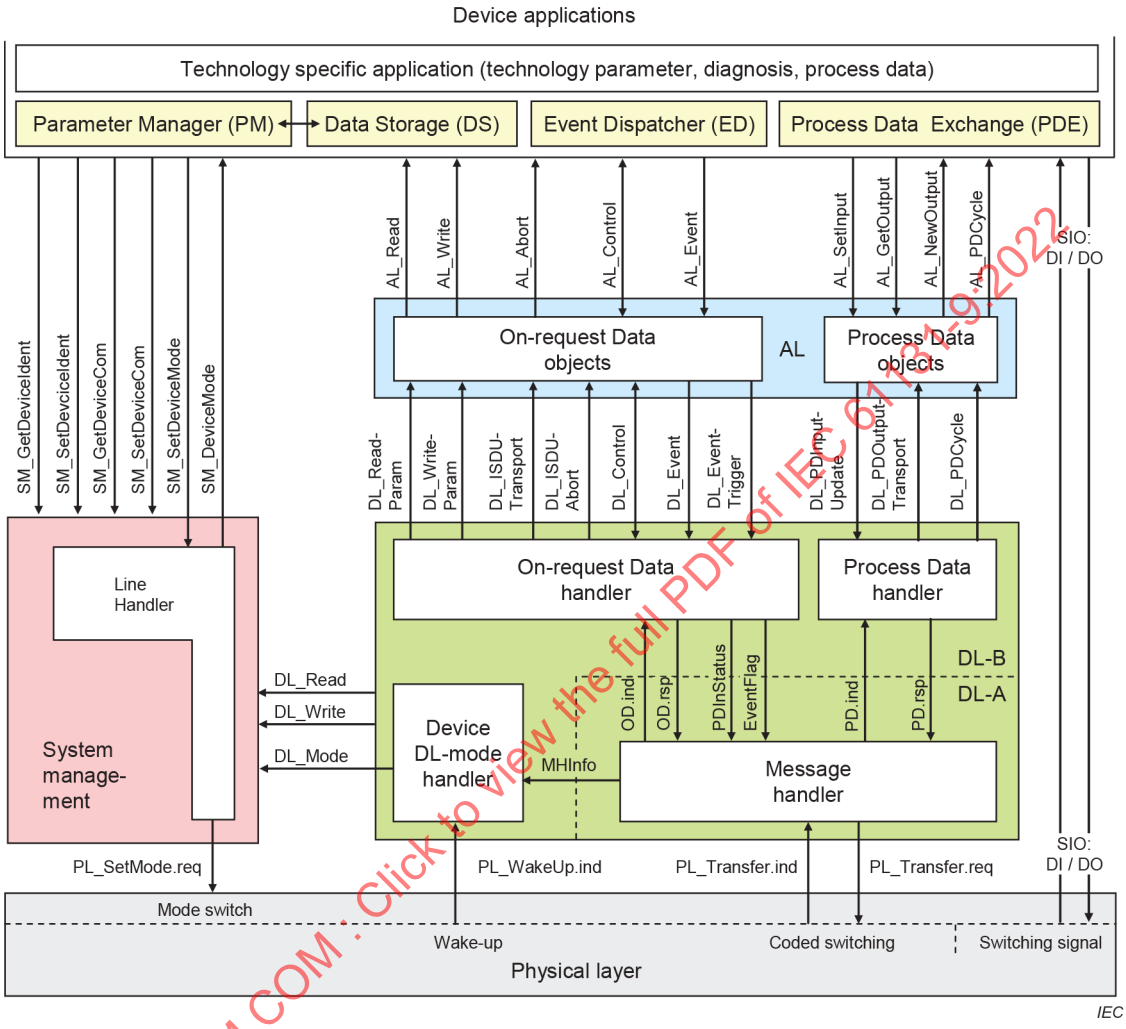


Figure 85 – Structure and services of a Device

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The Device applications comprise first the technology specific application consisting of the transducer with its technology parameters, its diagnosis information, and its Process Data. The common Device applications comprise:

– Parameter Manager (PM), dealing with compatibility and correctness checking of complete sets of technology (vendor) specific and common system parameters (see 10.3);

– Data Storage (DS) mechanism, which optionally uploads or downloads parameters to the Master (see 10.4);

– Event Dispatcher (ED), supervising states and conveying diagnosis information such as notifications, warnings, errors, and Device requests as peripheral initiatives (see 10.5);

– Process Data Exchange (PDE) unit, conditioning the data structures for transmission in case of a sensor or preparing the received data structures for signal generation. It also controls the operational states to ensure the validity of Process Data (see 10.2).

These Device applications provide standard methods or functions and parameters common to all Devices, and Device specific functions and parameters, all specified within Clause 10.

10.2 Process Data Exchange (PDE)

The Process Data Exchange unit cyclically transmits and receives Process Data without interference from the On-request Data (parameters, commands, and Events).

An actuator (output Process Data) shall observe the cyclic transmission and enter a default appropriate state, for example keep last value, stop, or de-energize, whenever the data transmission is interrupted (see 7.3.3.5 and 10.8.3). The actuator shall wait on the MasterCommand "ProcessDataOutputOperate" (see Table B.2, output Process Data "valid") prior to regular operation after restart in case of an interruption.

Within cyclic data exchange, an actuator (output Process Data) receives a Master-Command "DeviceOperate", whenever the output Process Data are invalid and a Master-Command "ProcessDataOutputOperate", whenever they become valid again (see Table B.2).

There is no need for a sensor Device (input Process Data) to monitor the cyclic data exchange. However, if the Device is not able to guarantee valid Process Data, the PD status "Process Data invalid" (see A.1.5) shall be signaled to the Master application.

10.3 Parameter Manager (PM)

10.3.1 General

A Device can be parameterized via two basic methods using the Direct Parameters or the Index memory space accessible with the help of ISDUs (see Figure 6).

Mandatory for all Devices are the so-called Direct Parameters in page 1. This page 1 contains common communication and identification parameters (see Clause B.1).

Direct Parameter page 2 optionally offers space for a maximum of 16 octets of technology (vendor) specific parameters for Devices requiring not more than this limited number and with small system footprint (ISDU communication not implemented, easier fieldbus handling possible but with less comfort). Access to the Direct Parameter page 2 is performed via AL_Read and AL_Write (see 10.8.5).

The transmission of parameters to and from the spacious Index memory can be performed in two ways: single parameter by single parameter or as a block of parameters. Single parameter transmission as specified in 10.3.4 is secured via several checks and confirmation of the transmitted parameter. A negative acknowledgment contains an appropriate error description, and the parameter is not activated. Block Parameter transmission as specified in 10.3.5 defers parameter consistency checking and activation until after the complete transmission. The Device performs the checks upon reception of a special command and returns a confirmation or a negative acknowledgment with an appropriate error description. In this case the transmitted parameters shall be rejected and a roll back to the previous parameter set shall be performed to ensure proper functionality of the Device.

10.3.2 Parameter Manager state machine

The Device can be parameterized using ISDU mechanisms whenever the PM is active. The main functions of the PM are the transmission of parameters to the Master ("Upload"), to the Device ("Download"), and the consistency and validity checking within the Device ("ValidityCheck") as demonstrated in Figure 86.

The PM is driven by command messages of the Master (see Table B.9). For example, the guard [UploadStart] corresponds to the reception of the SystemCommand "ParamUploadStart" and [UploadEnd] to the reception of the SystemCommand "ParamUploadEnd".

NOTE 1 Following a communication interruption, the Master System Management uses the service SM_DeviceMode with the variable "INACTIVE" to stop the upload process and to return to the "IDLE" state.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Any new "ParamUploadStart" or "ParamDownloadStart" while another sequence is pending, for example due to an unexpected shut-down of a vendor parameterization tool, will abort the pending sequence. The corresponding parameter changes will be discarded.

NOTE 2 A PLC user program and a parameterization tool can conflict (multiple access), for example if during commissioning, the user did not disable accesses from the PLC program while changing parameters via the tool.

The Parameter Manager mechanism in a Device is always active and the DS_ParUpload.req in transition T4 is used to trigger the Data Storage (DS) mechanism in 10.4.2.

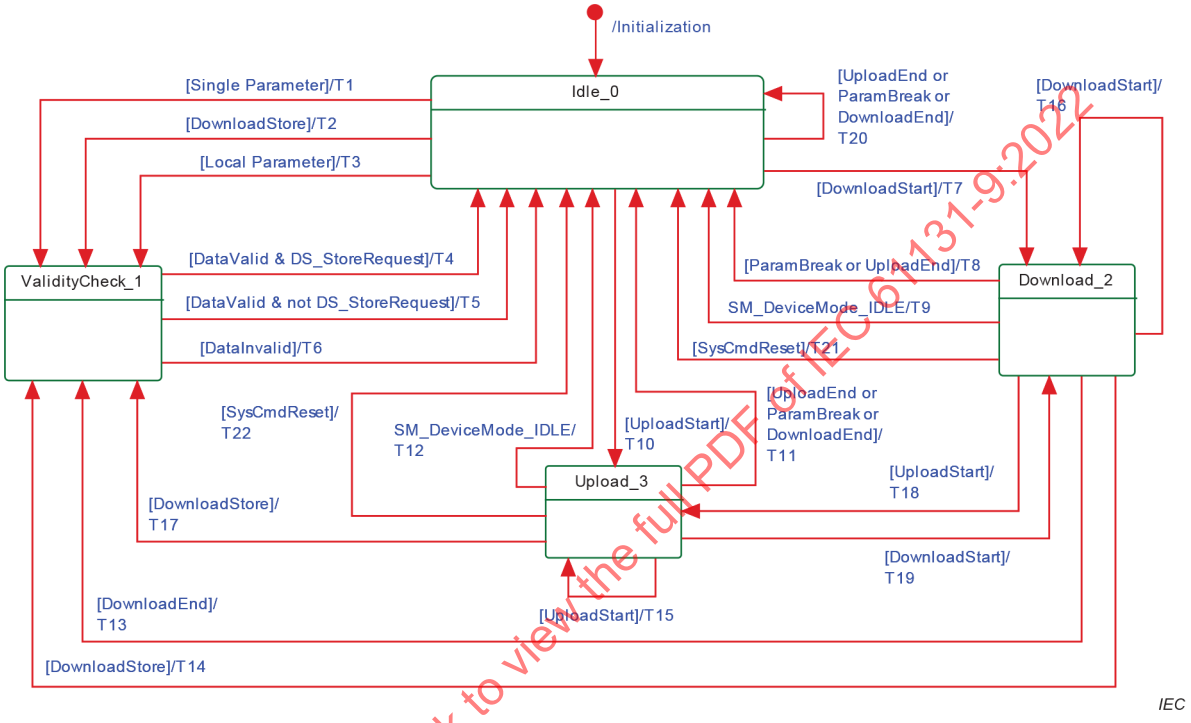


Figure 86 – Parameter Manager (PM) state machine

Table 96 shows the state transition table of the Device Parameter Manager (PM) state machine.

Table 96 – State transition table of the PM state machine

STATE NAME	STATE DESCRIPTION
Idle_0	Waiting on parameter transmission
ValidityCheck_1	Check of consistency and validity of current parameter set.
Download_2	Parameter download active; local parameterization locked (e.g. teach-in). All Read services to Indices other than 3 (DataStorageIndex) shall be rejected (ISDU ErrorType 0x8022 – "Service temporarily not available – Device control")
Upload_3	Parameter upload active; parameterization globally locked. All write accesses for parameter changes not covered in the state machine shall be rejected (ISDU ErrorType 0x8022 – "Service temporarily not available – Device control")

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	-
T2	0	1	Set "StoreRequest" (= TRUE)
T3	0	1	Set "StoreRequest" (= TRUE)
T4	1	0	Mark parameter set as valid; invoke DS_ParUpload.req to DS; enable positive acknowledgment of transmission; reset "StoreRequest" (= FALSE)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T5	1	0	Mark parameter set as valid; enable positive acknowledgment of transmission
T6	1	0	Mark parameter set as invalid; enable negative acknowledgment of transmission; reset "StoreRequest" (= FALSE); discard parameter buffer
T7	0	2	Lock local parameter access
T8	2	0	Unlock local parameter access; discard parameter buffer
T9	2	0	Unlock local parameter access; discard parameter buffer
T10	0	3	Lock local parameter access
T11	3	0	Unlock local parameter access
T12	3	0	Unlock local parameter access
T13	2	1	Unlock local parameter access
T14	2	1	Unlock local parameter access; set "StoreRequest" (= TRUE)
T15	3	3	Lock local parameter access
T16	2	2	Discard parameter buffer, so that a possible second start will not be blocked.
T17	3	1	Unlock local parameter access; set "StoreRequest" (= TRUE)
T18	2	3	Discard parameter buffer, so that a possible second start will not be blocked.
T19	3	2	–
T20	0	0	Return ErrorType 0x8036 – *Function temporarily unavailable* if Block Parameterization supported or ErrorType 0x8035 – *Function not available* if Block Parameterization is not supported.
T21	2	0	Unlock local parameter access; discard parameter buffer
T22	3	0	Unlock local parameter access

INTERNAL ITEMS	TYPE	DEFINITION
DownloadStore	Bool	SystemCommand "ParamDownloadStore" received, see Table B.9
DataValid	Bool	Positive result of conformity and validity checking
DataInvalid	Bool	Negative result of conformity and validity checking
DownloadStart	Bool	SystemCommand "ParamDownloadStart" received, see Table B.9
DownloadEnd	Bool	SystemCommand "ParamDownloadEnd" received, see Table B.9
DS_StoreRequest	Bool	Flag for a requested Data Storage sequence, i.e. SystemCommand "ParamDownloadStore" received (= TRUE)
ParamBreak	Bool	SystemCommand "ParamBreak" received, see Table B.9
SysCmdReset	Bool	One of the parameter reset SystemCommands received, see Table 101
UploadStart	Bool	SystemCommand "ParamUploadStart" received, see Table B.9
UploadEnd	Bool	SystemCommand "ParamUploadEnd" received, see Table B.9
Single Parameter	Bool	In case of "single parameter" as specified in 10.3.4
Local Parameter	Bool	In case of "local parameter" as specified in 10.3.3
"Parameter access locking" shall not be confused with "Device access locking" in Table B.12		

The Parameter Manager (PM) supports handling of "single parameter" (Index and Subindex) transfers as well as "Block Parameter" transmission (entire parameter set).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

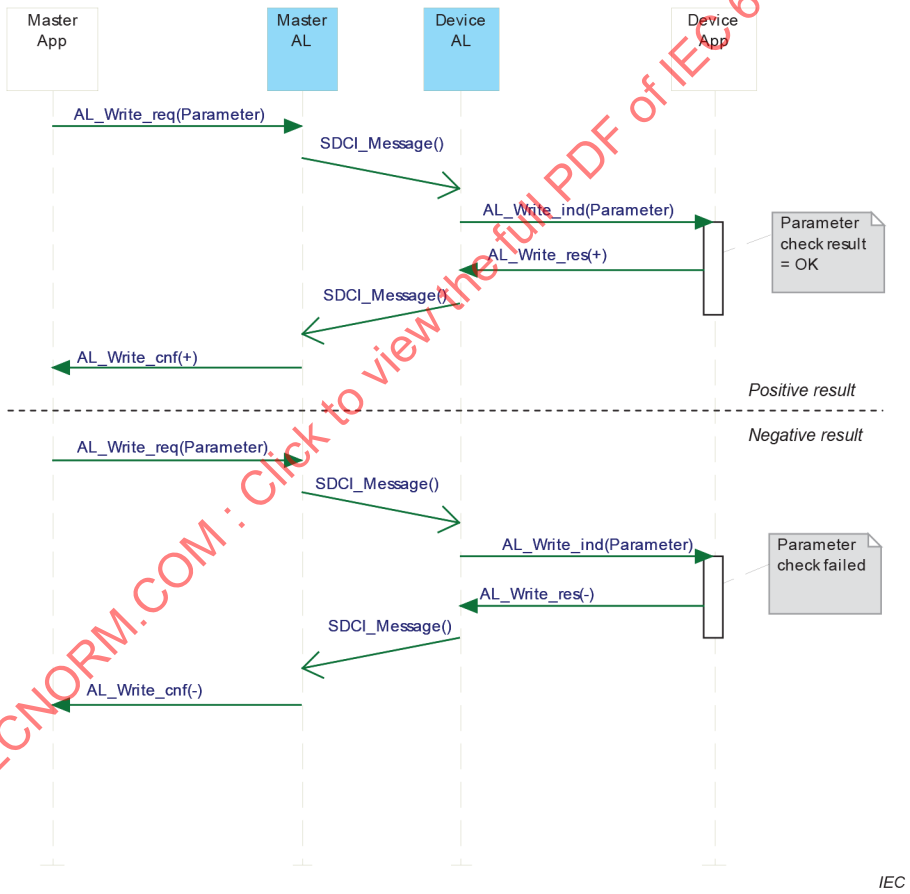
10.3.3 Dynamic parameter

Parameters accessible through SDCI read or write services can also be changed via on-board control elements (for example teach-in button) or the human machine interface of a Device. These changes shall undergo the same validity checks as a single parameter access. Thus, in case of a positive result "DataValid" in Figure 86, the "StoreRequest" flag shall be applied in order to achieve Data Storage consistency. In case of a negative result "InvalidData", the previous values of the corresponding parameters shall be restored ("roll back"). In addition, a Device specific indication on the human machine interface is recommended as a positive or negative feedback to the user.

It is recommended to avoid concurrent access to a parameter via local control elements and SDCI write services at the same point in time.

10.3.4 Single parameter

Sample sequence charts for valid and invalid single parameter changes are specified in Figure 87.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 87 – Positive and negative parameter checking result

If single parameterization is performed via ISDU objects, the Device shall check the access, structure, validity and consistency (see Table 97) of the transmitted data within the context of the entire parameter set and return the result in the confirmation. Via positive confirmation, the Device indicates that parameter contents

– passed all checks of Table 97 in the specified order 1 to 4,
– are stored in non-volatile memory in case of non-volatile parameters, and
– are activated in the Device specific technology if applicable.

The negative confirmation carries one of the ErrorTypes of Table C.2.

Table 97 – Sequence of parameter checks

Step	Parameter check	Definition	Error indication
1	Access	Check for valid access rights for this Index/Subindex, independent from data content (Index/Subindex permanently or temporarily unavailable; write/read access on read/write only Index)	See C.2.3 to C.2.8
2	Structure	Check for valid data structure like data size, only complete data structures can be written, for example 2 octets to an UInteger16 data type	See C.2.12 and C.2.13
3	Validity	Check for valid data content of single parameters, testing for data limits	See C.2.9 to C.2.11, C.2.14, C.2.15
4	Consistency	Check for valid data content of the entire parameter set, testing for interference or correlations between parameters	See C.2.16 and C.2.17
NOTE These checks are valid for single and Block Parameters (see 10.3.5).			

10.3.5 Block Parameter

User applications such as function blocks within PLCs and parameterization tool software can use start and end commands to indicate the beginning and end of a Block Parameter transmission. For the duration of the Block Parameter transmission the Device application shall inhibit all the parameter changes originating from other sources, for example local parameterization, or teach-in. In case parameter access is locked, any user application shall unlock "Parameter (write) access" (see Table B.12) prior to downloading a parameter set.

A sample sequence chart for valid Block Parameter changes with an optional Data Storage request is demonstrated in Figure 88.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

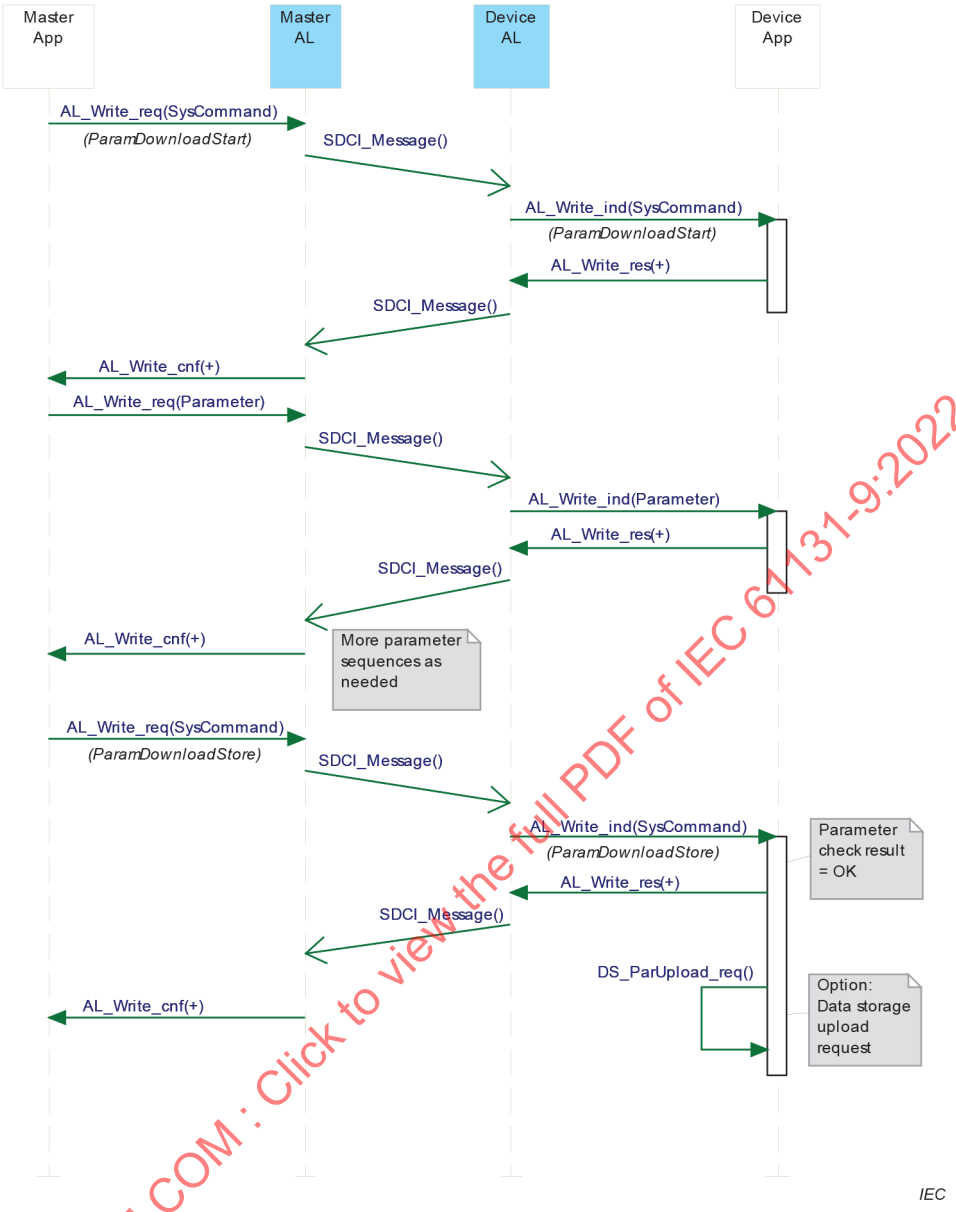


Figure 88 – Positive Block Parameter download with Data Storage request

A sample sequence chart for invalid Block Parameter changes is demonstrated in Figure 89.

The "ParamDownloadStart" command (see Table B.9) indicates the beginning of the Block Parameter transmission in download direction (from user application to the Device). The SystemCommand "ParamDownloadEnd" or "ParamDownloadStore" terminates this sequence. Both functions are similar. However, in addition the SystemCommand "ParamDownloadStore" causes the Data Storage (DS) mechanism to upload the parameter set through the DS_UPLOAD_REQ Event (see 10.4.2).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

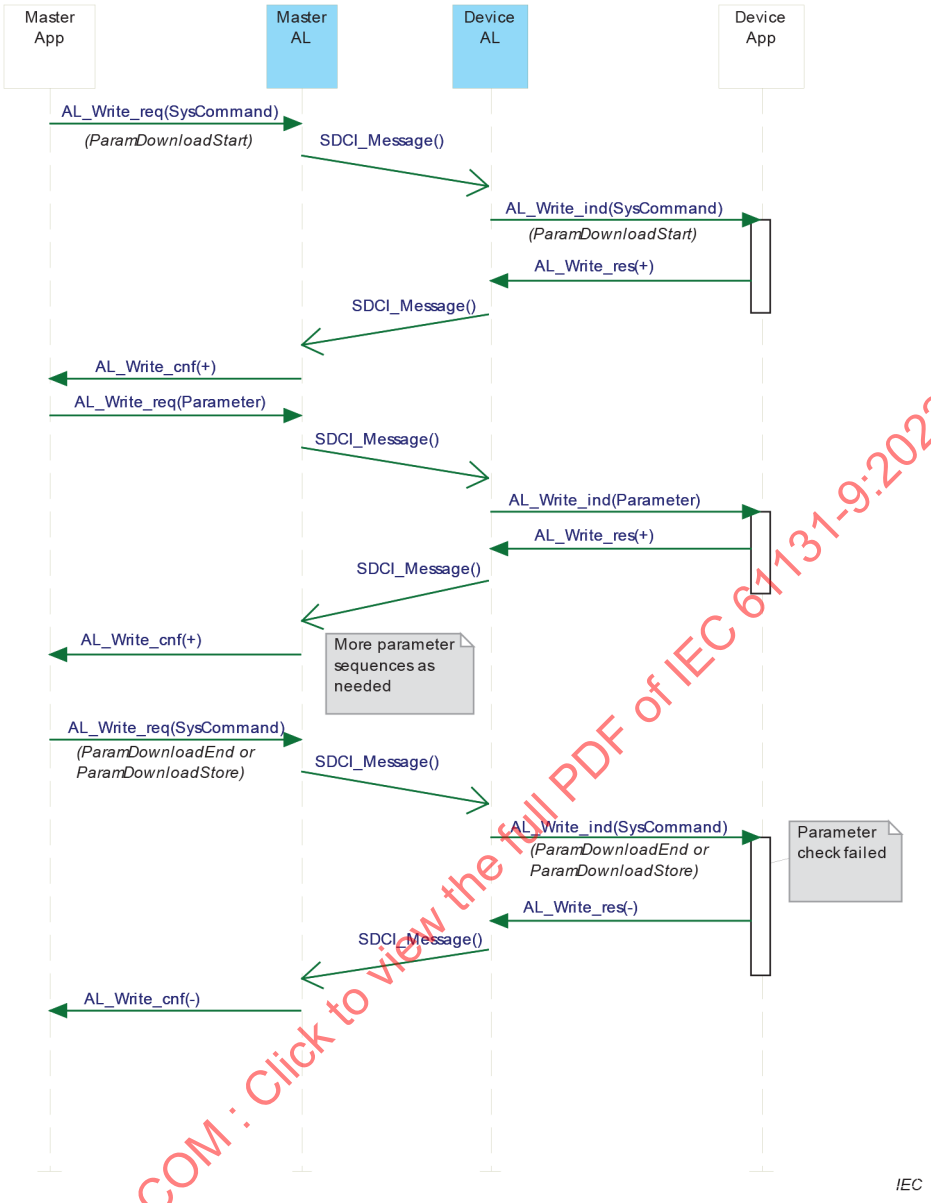


Figure 89 – Negative Block Parameter download

The checking steps and rules in Table 98 apply.

Table 98 – Steps and rules for Block Parameter checking

Rule	Action
1	At first, access and structure checks shall always be performed for each parameter (see Table 97).
2	Then, optionally, validity checks can be performed for each parameter.
3	At this time, consistency checking for transferred parameters shall be disabled and the single parameters shall not be activated.
4	The Parameter Manager shall not exit from block transfer mode in case of invalid accesses or structure violations or validity faults. The parameter set shall be treated as invalid if one of these checks failed.
5	With command "ParamDownloadEnd" or "ParamDownloadStore", the Device checks the validity of each parameter if not already performed and consistency of the entire parameter set. The parameter set shall be treated as invalid if one of these checks failed. The result of the check is indicated to the originator of the Block Parameter transmission within the ISDU acknowledgment in return to the command.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Rule	Action
6	Via positive confirmation the Device indicates that parameters – passed all checks of Table 97, – are stored in non-volatile memory in case of non-volatile parameters, – are activated in the Device specific technology if applicable.
7	Via negative confirmation, the Device indicates that any of the checks of Table 97 failed and the parameter set is invalid. The previous parameter set shall remain active. A Data Storage upload request shall not be triggered. The corresponding negative confirmation shall contain the ErrorType 0x8041 – Inconsistent parameter set (see C.2.17).

The "ParamUploadStart" command (see Table B.9) indicates the beginning of the Block Parameter transmission in upload direction (from the Device to the user application). The SystemCommand "ParamUploadEnd" terminates this sequence and indicates the end of transmission.

A Block Parameter transmission is aborted if the Parameter Manager receives a SystemCommand "ParamBreak". In this case the block transmission quits without any changes in parameter settings.

In any case, the response to all "ParamXXX" commands (see Table B.9) shall be transmitted after execution of the requested action.

10.3.6 Concurrent parameterization access

There is no mechanism to secure parameter consistency within the Device in case of concurrent accesses from different user applications above Master level. This shall be ensured or blocked on user level (see 13.2.2).

10.3.7 Command handling

Application commands are conveyed in form of parameters. As ISDU response, the appropriate priority level of the list in Table 99 shall be used.

Table 99 – Prioritized ISDU responses on command parameters

Priority	ISDU response	Condition
1	"Index not available", see C.2.3	Command parameter is not supported by the Device
2	"Function not available", see C.2.14	Command is not supported by the Device regardless of the Device state
3	"Function temporarily not available", see C.2.15	Command is supported but the actual state of the Device does not permit the requested command.
4	Write response (+)	Command is supported and accepted in the current state of the Device and the action is finished. However, within the context of certain commands, the action is just started. This exception is defined at a certain command.

In any case the ISDU timeout shall be observed (see Table 102).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.4 Data Storage (DS)

10.4.1 General

The Data Storage (DS) mechanism enables the consistent and up-to-date buffering of the Device parameters on upper levels like PLC programs or fieldbus parameter server. Data Storage between Masters and Devices is specified within this document, whereas the adjacent upper data storage mechanisms depend on the individual fieldbus or system. The Device holds a standardized set of objects providing information about parameters for Data Storage such as memory size requirements as well as control and state information of the Data Storage mechanism (see Table B.10). Revisions of Data Storage parameter sets are identified via a Parameter Checksum.

During Data Storage the Device shall apply the same checking rules as specified for the Block Parameter transfer in 10.3.5.

The implementation of the DS mechanism specified in this document is highly recommended for Devices. If this mechanism is not supported, it is the responsibility of the Device vendor to describe how parameterization of a Device after replacement can be ensured in a system conformant manner without tools.

10.4.2 Data Storage state machine

Any changed set of valid parameters leads to a new Data Storage upload. The upload is initiated by the Device by raising a "DS_UPLOAD_REQ" Event (see Table D.1). The Device shall store the internal state "Data Storage Upload" in non-volatile memory (see Table B.10, State Property), until it receives a Data Storage command "DS_UploadEnd" or "DS_DownloadEnd".

The Device shall generate an Event "DS_UPLOAD_REQ" (see Table D.1) only if the parameter set is valid and

– parameters assigned for Data Storage have been changed locally on the Device (for example teach-in, human machine interface), or
– the Device receives a SystemCommand "ParamDownloadStore".

With this Event information the Data Storage mechanism of the Master is triggered and initiates a Data Storage upload or download sequence depending on Port configuration. The state machine in Figure 90 specifies the Device Data Storage mechanism.

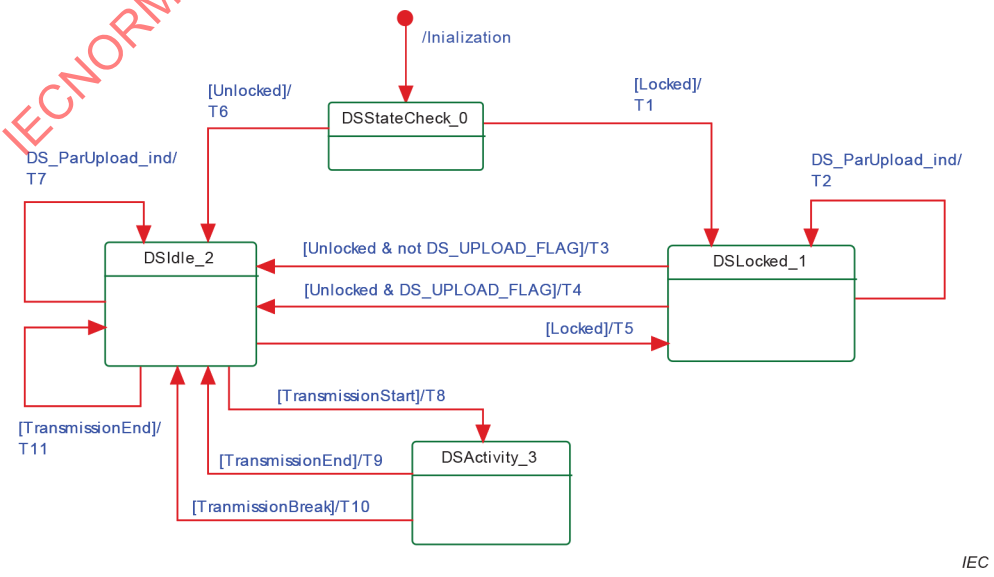


Figure 90 – Data Storage (DS) state machine

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 100 shows the state transition table of the Device Data Storage (DS) state machine. See Table B.10 for details on DataStorageIndex assignments.

Table 100 – State transition table of the Data Storage state machine

STATE NAME	STATE DESCRIPTION
DSStateCheck_0	Check activation state after initialization.
DSLocked_1	Waiting on Data Storage state machine to become unlocked. This state will become obsolete in future releases since Device access lock "Data Storage" shall not be used anymore (see Table B.12).
DSIdle_2	Waiting on Data Storage activities.
DSActivity_3	Provide parameter set; local parameterization locked.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	Set State_Property = "Data Storage access locked"
T2	1	1	Set DS_UPLOAD_FLAG = TRUE
T3	1	2	Set State_Property = "Inactive"
T4	1	2	Invoke AL_EVENT.req (EventCode: DS_UPLOAD_REQ), Set State_Property = "Inactive"
T5	2	1	Set State_Property = "Data Storage access locked"
T6	0	2	Set State_Property = "Inactive"
T7	2	2	Set DS_UPLOAD_FLAG = TRUE, invoke AL_EVENT.req (EventCode: DS_UPLOAD_REQ)
T8	2	3	Lock local parameter access, set State_Property = "Upload" or "Download"
T9	3	2	Set DS_UPLOAD_FLAG = FALSE, unlock local parameter access, Set State_Property = "Inactive"
T10	3	2	Unlock local parameter access. Set State_Property = "Inactive"
T11	2	2	Set DS_UPLOAD_FLAG = FALSE

INTERNAL ITEMS	TYPE	DEFINITION
Unlocked	Bool	Data Storage unlocked, see B.2.4
Locked	Bool	Data Storage locked, see B.2.4
DS_ParUpload.ind	Service	Device internal service between PM and DS (see Figure 86)
TransmissionStart	Bool	DS_Command "DS_UploadStart" or "DS_DownloadStart" has been invoked
TransmissionEnd	Bool	DS_Command "DS_UploadEnd" or "DS_DownloadEnd" has been invoked
TransmissionBreak	Bool	SM_MODE_INACTIVE or DS_Command "DS_Break" received
"Parameter access locking" shall not be confused with "Device access locking" in Table B.12		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The truncated sequence chart in Figure 91 demonstrates the important communication sequences after the parameterization.

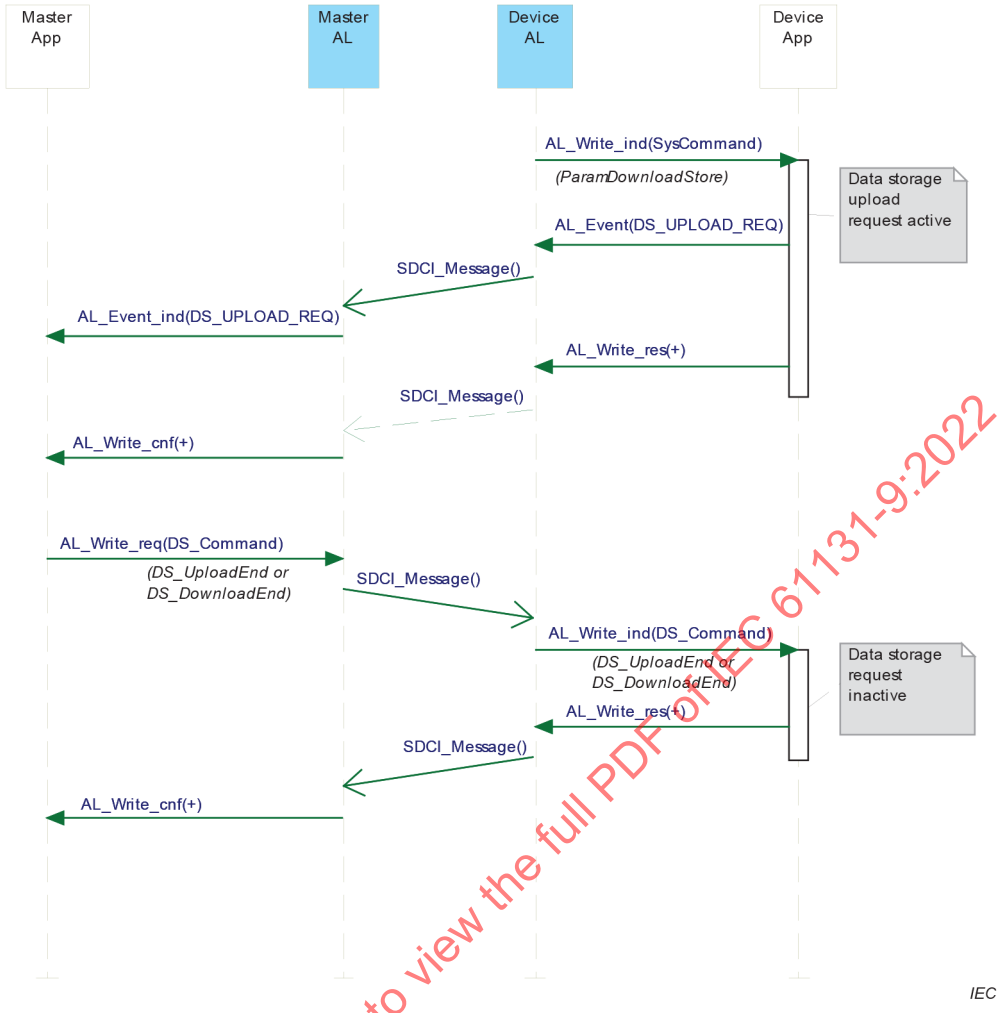


Figure 91 – Data Storage request message sequence

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.4.3 DS configuration

The Data Storage mechanism inside the Device can be disabled via the Master, for example by a tool or a PLC program. See B.2.4 for further details. This is recommended during commissioning or system tests to avoid intensive communication.

NOTE This functionality will be removed in future releases and the Data Storage mechanism will then only be controlled via Port configuration in the Master.

10.4.4 DS memory space

To handle the requested data amount for Data Storage under any circumstances, the requested number of Indices to be saved and the required total memory space are given in the Data Storage Size parameter, see Table B.10. The required total memory space (including the structural information) shall not exceed 2 048 octets (see Annex G). The Data Storage mechanism of the Master shall be able to support this amount of memory per Port.

10.4.5 DS Index_List

The Device is the "owner" of the DS Index_List (see Table B.10). Its purpose is to provide all the necessary information for a Device replacement. The DS Index_List shall be fixed for any specific DeviceID. Otherwise, the data integrity between Master and Device cannot be guaranteed. The Index List shall contain the termination marker (see Table B.10), if the Device does not support Data Storage (see 10.4.1). The required storage size shall be 0 in this case.

10.4.6 DS parameter availability

All Indices listed in the Index List shall be readable and writeable between the SystemCommands "DS_UploadStart" or "DS_DownloadStart" and "DS_UploadEnd" or "DS_DownloadEnd" (see Table B.10). If one of the Indices is rejected by the Device, the Data Storage Master will abort the up- or download with a SystemCommand "DS_Break". In this case no retries of the Data Storage sequence will be performed.

10.4.7 DS without ISDU

The support of ISDU transmission in a Device is a precondition for the Data Storage of parameters. Parameters in Direct Parameter page 2 cannot be saved and restored by the Data Storage mechanism.

10.4.8 DS parameter change indication

The Parameter_Checksum specified in Table B.10 is used as an indicator for changes in a parameter set. This document does not require a specific mechanism for detecting parameter changes. A set of recommended methods is provided in the informative Annex K.

10.5 Event Dispatcher (ED)

Any of the Device applications can generate predefined system status information when SDCI operations fail or technology specific information (diagnosis), as a result of technology specific diagnostic methods, occur. The Event Dispatcher turns this information into an Event according to the definitions in Clause A.6. The Event consists of an EventQualifier indicating the properties of an incident and an EventCode ID representing a description of this incident together with possible remedial measures. Table D.1 comprises a list of predefined IDs and descriptions for application-oriented incidents. Ranges of IDs are reserved for profile specific and vendor specific incidents. Table D.2 comprises a list of predefined IDs for SDCI specific incidents.

Events are classified in "Errors", "Warnings", and "Notifications". See 10.10.2 for these classifications and see 11.6 for how the Master is controlling and processing these Events.

All Events provided at one point in time are acknowledged with one single command. Therefore, the Event acknowledgment may be delayed by the slowest acknowledgment from upper system levels.

10.6 Device features

10.6.1 General

The following Device features are defined to a certain degree in order to achieve a common behavior. They are accessible via standardized or Device specific methods or parameters. The availability of these features is defined in the IODD of a Device.

10.6.2 Device backward compatibility

This feature enables a Device to play the role of a previous Device revision. In the start-up phase the Master System Management overwrites the Device's inherent DeviceID (DID) with the requested former DeviceID. The Device's technology application shall switch to the former functional sets or subsets assigned to this DeviceID. Device backward compatibility support is optional for a Device.

As a Device can provide backward compatibility to previous DeviceIDs (DID), these compatible Devices shall support all parameters and communication capabilities of the previous DeviceID. Thus, the Device is permitted to change any communication (except transmission rate) or identification parameter in this case.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.6.3 Protocol revision compatibility

This feature enables a Device to adjust its protocol layers to a previous SDCI protocol version such as for example to the legacy protocol version of a legacy Master or in the future from version V(x) to version V(x-n). In the start-up phase the Master System Management can overwrite the Device's inherent protocol RevisionID (RID) in case of discrepancy with the RevisionID supported by the Master. A legacy Master does not write the MasterCommand "MasterIdent" (see Table B.2) and thus the Device can adjust to the legacy protocol (V1.0). Revision compatibility support is optional for a Device.

Devices supporting both V1.0 and V1.1 mode are permitted

– to use the same predefined parameters, Events, and ErrorTypes in both modes;
– to support Block Parameterization with full functionality including the Event "DS_UP-LOAD_REQ". A legacy Master propagates such an Event without any further action.

10.6.4 Visual SDCI indication

This feature indicates the operational state of the Device's SDCI interface. The indication of the SDCI mode is specified in 10.10.3. Indication of the SIO mode is vendor specific and not covered by this definition. The function is triggered by the indication of the System Management (within all states except SM_Idle and SM_SIO in Figure 81). SDCI indication is optional for a Device.

10.6.5 Parameter access locking

This feature enables a Device to globally lock or unlock write access to all writeable Device parameters accessible via the SDCI interface (see B.2.4). The locking is triggered by the reception of a system parameter "Device Access Locks" (see Table B.8). The support for these functions is optional for a Device. However, it is highly recommended not to implement this (legacy) feature since it will be omitted in future releases.

10.6.6 Data Storage locking

Setting this lock will cause the "State_Property" in Table B.10 to switch to "Data Storage locked" and the Device not to send a DS_UPLOAD_REQ Event. Support of this function is optional for a Device if the Data Storage mechanism is implemented. However, it is highly recommended not to implement this (legacy) feature since it will be omitted in future releases.

10.6.7 Locking of local parameter entries

Setting this lock shall have the effect of read only or write protection for local entries at the Device (Bit 2 in Table B.12). Support of this function is optional for a Device, see B.2.4.

10.6.8 Locking of local user interface

Setting this lock shall have the effect of complete disabling of controls and displays, for example the shutdown of on-board human machine interface such as keypads on a Device (Bit 3 in Table B.12). Support of this function is optional for a Device.

10.6.9 Offset time

The OffsetTime t_{offset} is a parameter to be configured by the user (see B.2.24). It determines the beginning of the Device's technology data processing in respect to the start of the M-sequence cycle, i.e. the beginning of the Master (Port) message. The offset enables

– Data processing of a Device to be synchronized with the Master (Port) cycle within certain limits,

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

– Data processing of multiple Devices on different Master Ports to be synchronized with one another,
– Data processing of multiple Devices on different Master Ports to run with a defined offset.

Figure 92 demonstrates the timing of messages in respect to the data processing in Devices.

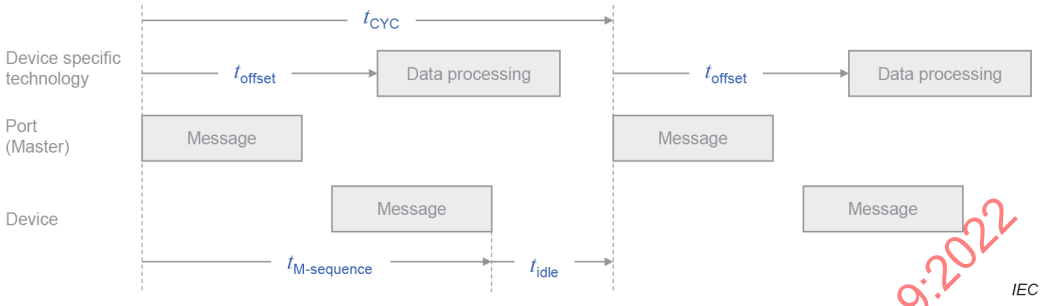


Figure 92 – Cycle timing

The OffsetTime defines a trigger relative to the start of an M-sequence cycle. The support for this function is optional for a Device.

10.6.10 Data Storage concept

The Data Storage mechanism in a Device allows to automatically save parameters in the Data Storage server of the Master and to restore them upon Event notification. Data consistency is checked in either direction within the Master and Device. Data Storage mainly focuses on configuration parameters of a Device set up during commissioning (see 10.4 and 11.4).

10.6.11 Block Parameter

The Block Parameter transmission feature in a Device allows transfer of parameter sets from a PLC program without checking the consistency single data object by single data object. The validity and consistency check are performed at the end of the Block Parameter transmission for the entire parameter set. This function mainly focuses on exchange of parameters of a Device to be set up at runtime (see 10.3). The support of this function is optional for a Device.

10.7 Device reset options

10.7.1 Overview

There are five possibilities for the user to put a Device into a certain defined condition by using either

– Power supply off/on (PowerCycle), or
– SystemCommand "Device reset" (128), or
– SystemCommand "Application reset" (129), or
– SystemCommand "Restore factory settings" (130), or
– SystemCommand "Back to box" (131).

Table B.9 defines which of these SystemCommands are mandatory, highly recommended or optional.

Table 101 provides an overview on impacted items when performing one of these options.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 101 – Overview on reset options and their impact on Devices

Impacted item	Power-Cycle	Device reset	Application reset	Restore factory settings	Back-to-box
Diagnosis and status	"0"	"0"	No	Clear	"0"
History recorder	No	No	No	No	No
Technology specific parameters (adjustable, teachable)	No	No	Default	Default	Default
Identification or tags, e.g. location, function	No	No	No	Default	Default
Data Storage behavior	No	No	Upload required DS_UPLOAD_REQ =1, DS Event	Delete upload request DS_UPLOAD_REQ =0	Delete upload request DS_UPLOAD_REQ =0
RevisionID	Default	Default	No	Default	Default
DeviceID	No	No	No	Default	Default
COM behavior	Restart via Master	Restart triggered by Device	No	Restart triggered by Device if active COM parameter differ from default	Device stops and disables communication until next PowerCycle
Access locks	No	No	Default	Default	Default
Block Parameter transfer (see Figure 86)	–	Discard	Discard	Discard	Discard

Key

"0"	The numerical parameter or list of parameters is set to zero
PowerCycle	Device power on → off → on
Initial	Set to initial values according to power up state
COM	Communication
No	Not affected
Clear	Set to "0" in case of no COM restart. All active Events will be sent with "Disappear" to clear DeviceStatus. After a performed "Restore factory settings", pending Events can be resent.
Default	Reset to initial value of state of delivery to customer
Event	Trigger upload via DS_UPLOAD_REQ flag
Discard	Transferred parameters not activated
History recorder	For example: Operation hour meter

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.7.2 Device reset

This feature enables a Device to perform a "warm start". It is especially useful whenever a Device needs to be reset to an initial state such as power-on, which means that communication will be interrupted.

This feature is triggered upon reception of a SystemCommand "Device reset" (see Table B.9). In any case, the ISDU response to this SystemCommand shall be transmitted immediately to the Master, also allowing retries in case of communication faults.

It is optional for a Device.

10.7.3 Application reset

This feature enables a Device to reset the technology specific application. It is especially useful whenever a technology specific application needs to be set to a predefined operational state without communication interruption and a shut-down cycle. Contrary to "Restore factory settings" only the application specific parameters are reset to "Default". Each and every identification parameter remains unchanged.

This feature is triggered upon reception of a SystemCommand "Application reset" (see Table B.9). In any case, the ISDU response to this SystemCommand shall be transmitted to the Master after successful execution of the requested action.

It is highly recommended for a Device.

10.7.4 Restore factory settings

This feature enables a Device to restore parameters to the original delivery status. It is triggered upon reception of the SystemCommand "Restore factory settings" (see Table B.9). The DS_UPLOAD_FLAG (see Table B.10) and other dynamic parameters such as "ErrorCount" (see B.2.19), "DeviceStatus" (see B.2.20), and "DetailedDeviceStatus" (see B.2.21) shall be reset when this feature is applied. This does not include vendor specific parameters such as for example counters of operating hours.

NOTE In this case an existing stored parameter set within the Master will be automatically downloaded into the Device after the next communication restart. This can be avoided by using the “Back to box” SystemCommand (see 10.7.5).

It is the Device vendor's responsibility to guarantee the correct function under any circumstances. If any communication parameter (see Direct Parameter page 1 in Table B.1) is changed during this restore, the communication shall be stopped by the Device to trigger a new communication start using the updated communication parameters. In this case, the response to this SystemCommand shall be transmitted to the Master after successful execution of the requested action, also allowing retries in case of communication faults.

It is optional for a Device.

10.7.5 Back-to-box

This feature enables a Device to restore parameters to the original delivery values without any interaction with upper-level mechanisms such as Data Storage or PLC based parameterization. It is especially useful whenever a Device is removed from an already parameterized installation and reactivated for example as a spare part. If the Device remains in an automation application beyond the next PowerCycle, all parametrizations will be overwritten just as if it were a replacement.

It is triggered upon reception of the SystemCommand "Back-to-box" (see Table B.9), i.e. the Device shall stop and disable communication until the next PowerCycle. If the Device provides a user interface, the special state "Waiting for PowerCycle" shall be visible. In any case, the response to a SystemCommand shall be transmitted to the Master, also allowing retries in case of communication faults.

It is mandatory for a Device.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.8 Device design rules and constraints

10.8.1 General

In addition to the protocol definitions in form of state, sequence, activity, and timing diagrams some more rules and constraints are required to define the behavior of the Devices. An overview of the major protocol variables scattered all over this document is concentrated in Table 102 with associated references.

10.8.2 Process Data

The process communication channel transmits the cyclic Process Data without any interference of the On-request Data communication channels. Process Data exchange starts automatically whenever the Device is switched into the OPERATE state via a message from the Master.

The format of the transmitted data is Device specific and varies from no data octets up to 32 octets in each communication direction.

Recommendations:

– Data structures should be suitable for use by PLC applications.
– It is highly recommended to comply with the rules in F.3.3 and in [6].

See A.1.5 for details on the indication of valid or invalid Process Data via a PDValid flag within cyclic data exchange.

10.8.3 Communication loss

It is the responsibility of the Device designer to define the appropriate behavior of the Device in case communication with the Master is lost (transition T10 in Figure 44 handles detection of the communication loss, while 10.2 defines resulting Device actions). This is especially important for actuators such as valves or motor management.

10.8.4 Direct Parameter

The Direct Parameter page communication provides no handshake mechanism to ensure proper reception or validity of the transmitted parameters. The Direct Parameter page can only be accessed single octet by single octet (Subindex) or as a whole (16 octets). The consistency of parameters larger than 1 octet cannot be guaranteed.

The parameters from the Direct Parameter page cannot be saved and restored via the Data Storage mechanism.

10.8.5 ISDU communication channel

The ISDU communication channel provides a powerful means for the transmission of parameters and commands (see Clause B.2).

The following rules shall be considered when using this channel (see Figure 7).

– Index 0 is not accessible via the ISDU communication channel. The access is redirected by the Master to the Direct Parameter page 1 using the page communication channel.
– Index 1 is not accessible via the ISDU communication channel. The access is redirected by the Master to the Direct Parameter page 2 using the page communication channel.
– Index 3 cannot be accessed by a PLC application program. The access is limited to the Master application only (Data Storage).
– After reception of an ISDU request from the Master, the Device shall respond within 5 000 ms (see Table 102). Any violation causes the Master to abandon the current task.
– Parameters with attribute write-only (W) shall be treated like a SystemCommand. Only basic data types are permitted.

10.8.6 DeviceID rules related to Device variants

Devices with a certain DeviceID and VendorID shall not deviate in communication and functional behavior. This applies for sensors and actuators. Those Devices may vary for example in

– cable lengths,
– housing materials,
– mounting mechanisms,
– other features, and environmental conditions.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.8.7 Protocol constants

Table 102 gives an overview of the major protocol constants for Devices.

Table 102 – Overview of the protocol constants for Devices

System variable	References	Values	Definition
ISDU acknowledgment time, for example after a SystemCommand	B.2.2	5 000 ms	Time from reception of an ISDU for example SystemCommand and the beginning of the response message of the Device (see Figure 63)
Maximum number of entries in Index List	B.2.3	70	Each entry comprises an Index and a Subindex. 70 entries result in a total of 210 octets.
Preset values for unused or reserved parameters, for example FunctionID	Annex B	0 (if numbers) 0x00 (if characters)	Engineering shall set all unused parameters to the preset values.
Wake-up procedure	7.3.2.2	See Table 42 and Table 43	Minimum and maximum timings and number of retries
MaxRetry	7.3.3.3	2, see Table 46	Maximum number of retries after communication errors
MinCycleTime	A.3.7 and B.1.3	See Table A.11 and Table B.3	Device defines its minimum cycle time to aquire input or process output data.
Usable Index range	Clause B.2	See Table B.8	This document reserves some areas within the total range of 65 535 Indices.
Errors and warnings	10.10.2	50 ms	An Event with MODE "Event appears" shall stay at least for the duration of this time.
EventCount	8.2.2.11	1	Constraint for AL_Event.req

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.9 IO Device description (IODD)

An IODD (I/O Device Description) is a file that provides all the necessary properties to establish communication and the necessary parameters and their boundaries to establish the desired function of a sensor or actuator.

An IODD (I/O Device Description) is a file that formally describes a Device.

An IODD file shall be provided for each Device and shall include all information necessary to support this document.

The IODD can be used by engineering tools for PLCs and/or Masters for the purpose of identification, configuration, definition of data structures for Process Data exchange, parameterization, and diagnosis decoding of a particular Device.

NOTE Details of the IODD language to describe a Device can be found in [6].

10.10 Device diagnosis

10.10.1 Concepts

This document provides only most common EventCodes in Clause D.2. The purpose of these common diagnosis information is to enable an operator or maintenance person to take fast remedial measures without deep knowledge of the Device's technology. Thus, the text associated with a particular EventCode shall always contain a corrective instruction together with the diagnosis information.

Fieldbus-Master-Gateways tend to only map few EventCodes to the upper system level. Usually, vendor specific EventCodes defined via the IODD can only be decoded into readable instructions via a Port and Device Configuration Tool (PDCT) or specific vendor tool using the IODD.

Condensed information of the Device's "state of health" can be retrieved from the parameter "DeviceStatus" (see B.2.20). Table 103 provides an overview of the various possibilities for Devices and shows examples of consumers for this information.

If implemented, it is also possible to read the number of faults since power-on or reset via the parameter "ErrorCount" (see B.2.19) and more information in case of profile Devices via the parameter "DetailedDeviceStatus" (see B.2.21).

NOTE Profile specific values for the "DetailedDeviceStatus" are given in [7].

If required, it is highly recommended to provide additional "deep" technology specific diagnosis information in the form of Device specific parameters (see Table B.8) that can be retrieved via Port and Device configuration tools for Masters or via vendor specific tools. Usually, only experts or service personnel of the vendor are able to draw conclusions from this information.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 103 – Classification of Device diagnosis incidents

Diagnosis incident	Appear/ disappear	Single shot	Parameter	Destination	Consumer
Error (fast remedy; standard EventCodes)	yes	–	–	PLC or HMI (fieldbus mapping)	Maintenance and repair personnel
Error (IODD: vendor specific EventCodes; see Table D.1)	yes	–	–	PDCT or vendor tool	Vendor service personnel
Error (via Device specific parameters)	–	–	See Table B.8	PDCT or vendor tool	Vendor service personnel
Warning (fast remedy; standard EventCodes)	yes	–	–	PLC or HMI	Maintenance and repair personnel
Warning (IODD: vendor specific EventCodes; see Table D.1)	yes	–		PDCT or vendor tool	Vendor service personnel
Warning (via Device specific parameters)	–	–	See Table B.8		
Notification (Standard EventCodes)	–	yes		PDCT	Commissioning personnel
Detailed Device status	–	–		PDCT or vendor tool	Commissioning personnel and vendor service personnel
Number of faults via parameter "ErrorCount"	–	–	See B.2.19		
Device "health" via parameter "DeviceStatus"	–	–	See B.2.20, Table B.13	HMI, Tools such as "Asset Management"	Operator

10.10.2 Events

MODE values shall be assigned as follows (see A.6.4):

– Events of TYPE "Error" shall use the MODEs "Event appears/disappears",
– Events of TYPE "Warning" shall use the MODEs "Event appears/disappears",
– Events of TYPE "Notification" shall use the MODE "Event single shot".

The following requirements apply:

– All Events already placed in the Event queue are discarded by the Event Dispatcher when communication is interrupted or cancelled. Once communication resumed, the technology specific application is responsible for proper reporting of the current Event causes.

– It is the responsibility of the Event Dispatcher to control the "Event appears" and "Event disappears" flow. Once the Event Dispatcher has sent an Event with MODE "Event appears" for a given EventCode, it shall not send it again for the same EventCode before it has sent an Event with MODE "Event disappears" for this same EventCode.

– Each Event shall use static mode, type, and instance attributes.

– Each vendor specific EventCode shall be uniquely assigned to one of the TYPEs (Error, Warning, or Notification).

In order to prevent the diagnosis communication channel (see Figure 7) from being flooded, the following requirements apply:

– The same diagnosis information shall not be reported at less than 1 s intervals. This means that the Event Dispatcher shall not invoke the AL_Event service with the same EventCode and EventQualifier more often than once per second. This measure avoids frequent repetitions of Events.

– The Event Dispatcher shall not issue an "Event disappears" less than 50 ms after the corresponding "Event appears".

– Subsequent incidents of errors or warnings with the same root cause shall be disregarded, i.e. one root cause shall lead to a single error or warning.

– The Event Dispatcher shall invoke the AL_Event service with an EventCount equal to one.

– Errors are prioritized over Warnings.

Figure 93 shows how two successive errors are processed, and the corresponding flow of "Event appears"/"Event disappears" Events for each error.

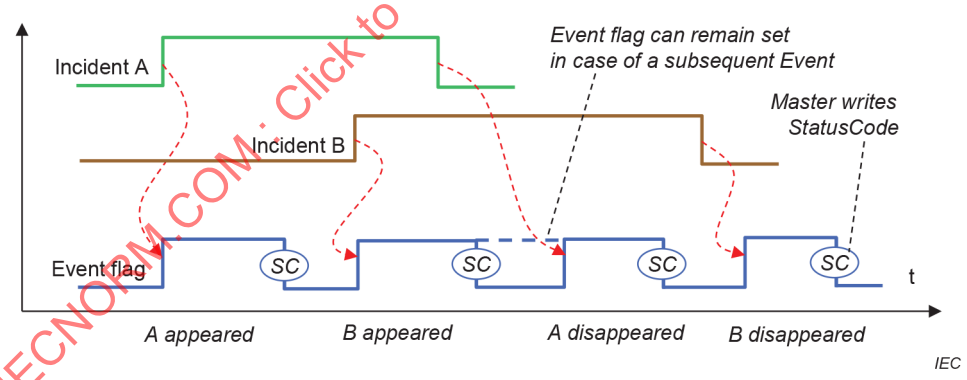


Figure 93 – Event flow in case of successive errors

10.10.3 Visual indicators

The indication of SDCI communication on the Device is optional. The SDCI indication shall use a green indicator. The indication follows the timing and specification shown in Figure 94.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

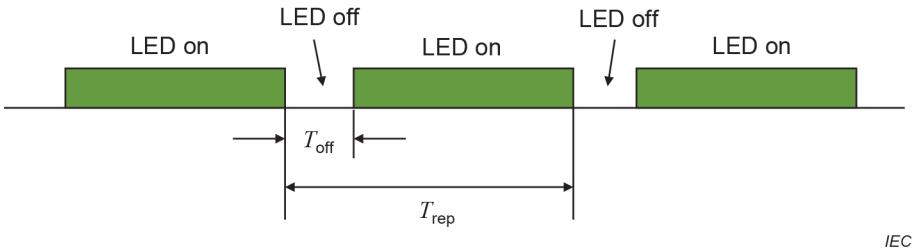


Figure 94 – Device LED indicator timing

Table 104 defines the timing for the LED indicator of Devices.

Table 104 – Timing for LED indicators

Timing	Minimum	Typical	Maximum	Unit
T_{rep}	750	1 000	1 250	ms
T_{off}	75	100	150	ms
T_{off} /T_{rep}	7,5	10	12,5	%
NOTE The timings are defined such that the general perception would be "power is on".				

A short periodical interruption indicates that the Device is in COMx communication state. In order to avoid flickering, the indication cycle shall start with a "LED off" state and shall always be completed (see Table 104).

10.11 Device connectivity

See 5.5 for the different possibilities of connecting Devices to Master Ports and the corresponding cable types as well as the color coding.

NOTE For compatibility reasons, this document does not prevent SDCI devices from providing additional wires for connection to functions outside the scope of this document (for example to transfer analog output signals).

11 Master

11.1 Overview

11.1.1 Positioning of Master and Gateway Applications

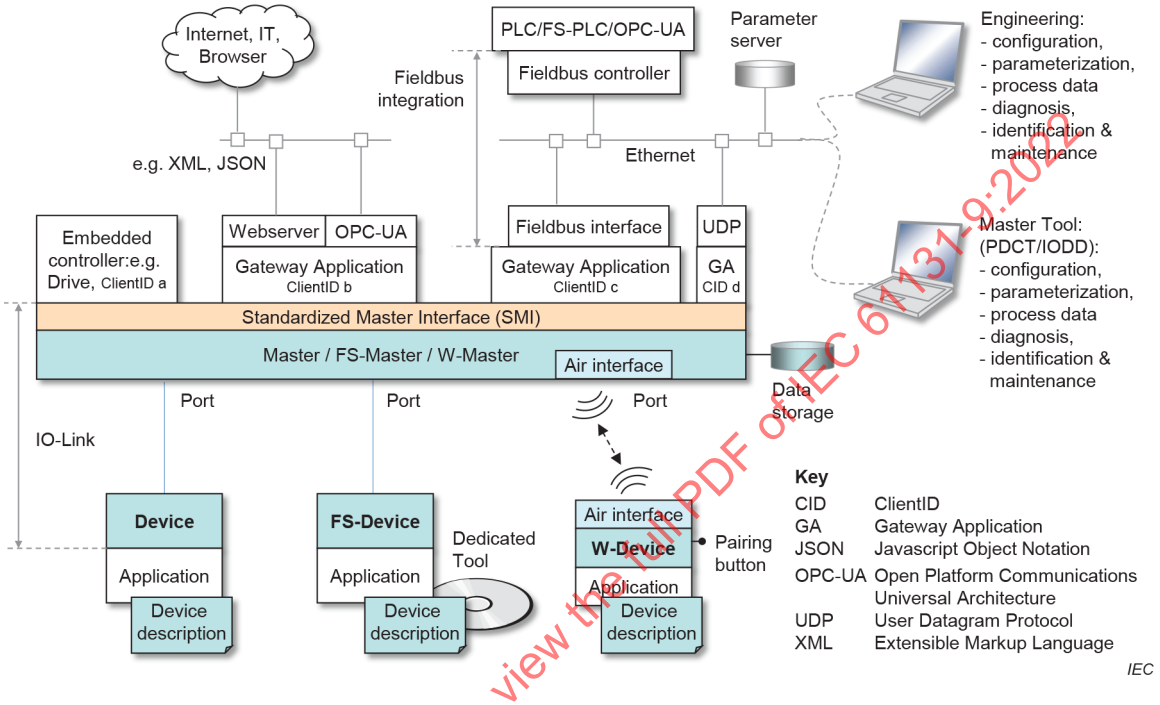
In 4.2 the domain of the SDCI technology within the automation hierarchy is already illustrated. Figure 95 shows the recommended relationship between the SDCI technology and a fieldbus technology. Even though this may be the major use case in practice, this does not automatically imply that the SDCI technology depends on the integration into fieldbus systems. It can also be directly integrated into PLC systems, industrial PCs, or other automation systems without fieldbus communication in between.

For the sake of preferably uniform behavior of Masters, Figure 95 shows a Standardized Master Interface (SMI) as layer in between the Master and the Gateway Applications or embedded systems on top. This Standardized Master Interface is intended to also serve the safety system extensions as well as the wireless system extensions. In case of FS-Masters, attention shall be payed to the fact that this SMI, in some aspects, requires implementation according to safety standards.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The Standardized Master Interface is specified in 11.2 via services and data objects similar to the other layers (PL, DL, and AL) in this document. It is designed using few uniform base structures that both upper layer fieldbus and upper layer IT systems can use in an efficient manner: push ("write"), pull ("read"), push/pull ("write/read"), and indication ("Event").

The specification of Gateway Applications is not within the scope of this document. Designers shall observe the realtime requirements of control functions and safety functions in case of concurrent Gateway Applications (see 13.2).



NOTE Blue and orange shaded areas indicate features specified in this document except those for functional safety (FS) and wireless (W).

Figure 95 – Generic relationship of SDCI and automation technology

11.1.2 Structure, applications, and services of a Master

Figure 96 provides an overview of the complete structure and the services of a Master.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

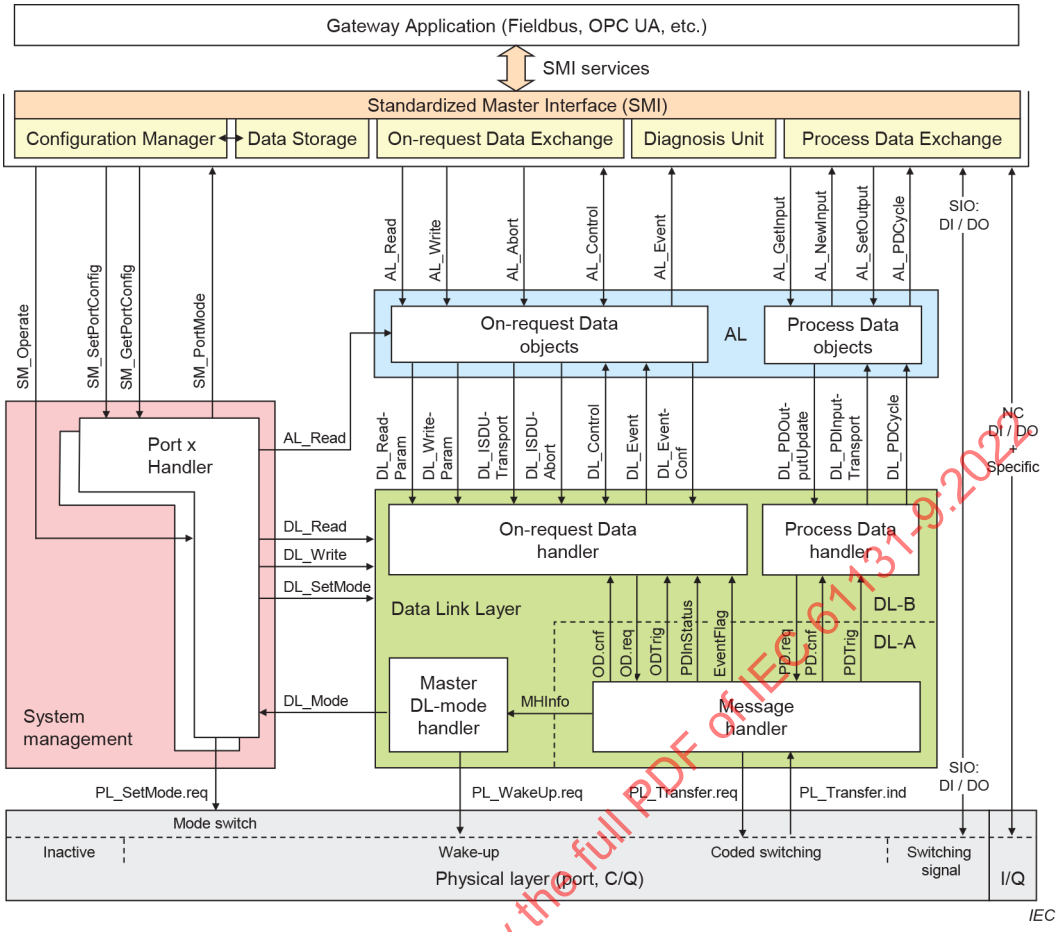


Figure 96 – Structure, applications and services of a Master

The Master applications are located on top of the Master structure and consist of:

– Configuration Manager (CM), which transforms the user configuration assignments into Port set-ups,

– On-request Data Exchange (ODE), which provides for example acyclic parameter access,

– Data Storage (DS) mechanism, which can be used to save and restore the Device parameters,

– Diagnosis Unit (DU), which routes Events from the AL to the Data Storage unit or the gateway application,

– Process Data Exchange (PDE), building the bridge to upper-level automation instruments.

They are accessible by the gateway applications (and others) via the Standardized Master Interface (SMI) and its services and methods.

These services and corresponding functions are specified in an abstract manner within 11.2.2 to 11.2.22 and Annex E.

Master applications are described in detail in 11.3 to 11.7. The Configuration Manager (CM) and the Data Storage mechanism (DS) require special coordination with respect to On-request Data.

11.1.3 Object view of a Master and its Ports

Figure 97 illustrates the data object model of a Master and its Ports from an SMI point of view.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

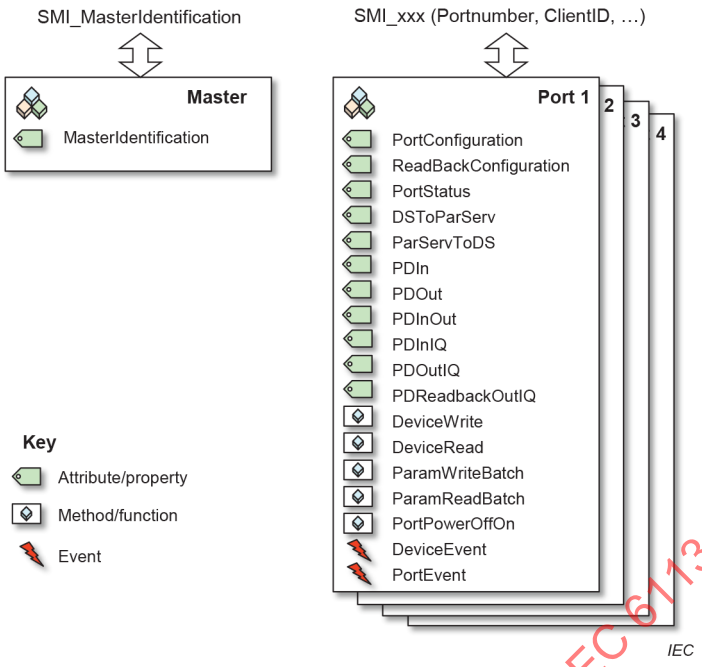


Figure 97 – Object model of Master and Ports

Each object comes with attributes and methods that can be accessed by SMI services. They all are specified in 11.2.

11.2 Services of the Standardized Master Interface (SMI)

11.2.1 Overview

Figure 98 illustrates the individual SMI services available for example to gateway applications.

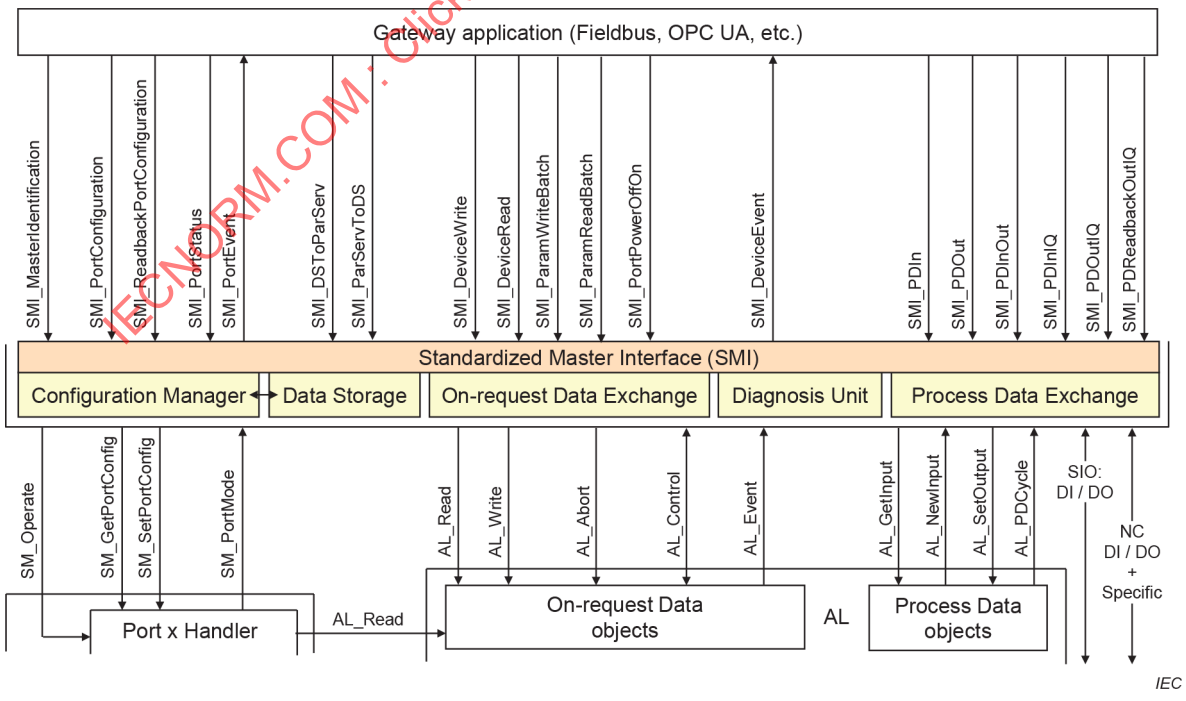


Figure 98 – SMI services

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Communication interfaces such as Fieldbus, OPC UA, JSON, UDP or alike are responsible to provide access to the SMI services. It is mandatory for upper-level communication systems to refer to the SMI definitions in their adaptations. Functionality behind SMI is mandatory unless it is specifically declared as optional.

Table 105 lists the SMI services available to gateway applications or other clients.

Table 105 – SMI services

Service name	Master	M/O/C	Purpose
SMI_MasterIdentification	R	M	Universal service to identify any Master
SMI_PortConfiguration	R	M	Setting up Port configuration
SMI_ReadbackPortConfiguration	R	M	Retrieve current Port configuration
SMI_PortStatus	R	M	Retrieve Port status
SMI_DSToParServ	R	M	Transfer Data Storage to parameter server
SMI_ParServToDS	R	M	Transfer Parameter server to Data Storage
SMI_DeviceWrite	R	M	ISDU transport to Device
SMI_DeviceRead	R	M	ISDU transport from Device
SMI_ParamWriteBatch	R	O	Batch ISDU transport of parameters (write)
SMI_ParamReadBatch	R	O	Batch ISDU transport of parameters (read)
SMI_PortPowerOffOn	R	O	PortPowerOffOn
SMI_DeviceEvent	I	M	Universal "Push" service for Device Events
SMI_PortEvent	I	M	Universal "Push" service for Port Events
SMI_PDIn	R	M	Retrieve PD from InBuffer
SMI_PDOut	R	M	Set PD in OutBuffer
SMI_PDInOut	R	M	Retrieve In- and OutBuffer
SMI_PDInIQ	R	C	Process data in at I/Q (Pin 2 on M12)
SMI_PDOutIQ	R	C	Process data out at I/Q (Pin 2 on M12)
SMI_PDReadbackOutIQ	R	C	Retrieve process data out at I/Q (Pin 2 on M12)
Key I Initiator of service M Mandatory	R Receiver (Responder) of service O Optional		C Conditional

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.2.2 Structure of SMI service arguments

The SMI service arguments contain a fixed structure of standard elements, which are characterized in the following.

ClientID

Gateway Applications can use the SMI services concurrently as clients of the SMI (see 11.2.3). Thus, SMI services will assign a unique ClientID to each individual client. It is the responsibility of the Gateway Application(s) to coordinate these SMI service activities and to route responses to the calling client. The maximum number of concurrent clients is Master specific.

Data type: Unsigned8

Permitted values: 1 to vendor specific maximum number of concurrent clients. "0" is solely used for broadcast purposes in case of indications, see 11.2.15 and 11.2.16.

PortNumber

Each SMI service contains the Port number in case of an addressed Port object (job) or in case of a triggered Port object (event).

Data type: Unsigned8.

Permitted values: 1 to MaxNumberOfPorts. "0" is solely used to address the entire Master (see 11.2.4).

ExpArgBlockID

This element specifies the expected ArgBlockID to carry the response data of a service request. The IDs are defined in Table E.1.

Data type: Unsigned16.

Permitted values: 1 to to 65 535.

RefArgBlockID

Within results, this element specifies the ID of the Argblock sent by the service request. The IDs are defined in Table E.1.

Data type: Unsigned16.

Permitted values: 1 to to 65 535.

ArgBlockLength

This element specifies the total length of the subsequent ArgBlock. Vendor specific extensions are not permitted.

Data type: Unsigned16.

Permitted values: 2 to to 65 535.

ArgBlock

All SMI services contain an ArgBlock characterized by an ArgBlockID and its description. Service results provide the ArgBlock associated to the ExpArgBlockID, which is part of this ArgBlock. The possibly variable length of the ArgBlock is predefined through the definition in this document.

Pairs of ExpArgBlock/RefArgBlock and ArgBlockID within one SMI structure shall be unique. Detailed coding of the ArgBlocks is specified in Annex E. ArgBlock types and their ArgBlockIDs are defined in Table E.1. Service errors are listed at each individual service and in Clause C.4.

11.2.3 Concurrency and prioritization of SMI services

The following rules apply for concurrency of SMI services when accessing attributes:

– all SMI services with different PortNumber access different Port objects (disjoint operations),

– different SMI services using the same PortNumber access different attributes and methods of a Port object (concurrent operations),

– identical SMI services using the same PortNumber and different ClientIDs access identical attributes concurrently (consistency).

The following rules apply for SMI services when accessing methods:

– SMI services for methods using different PortNumbers access different Port objects (disjoint operations),

– SMI services for methods using the same PortNumber and different ClientIDs create job instances and will be processed in the order of their arrival (n Client concurrency),

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

– SMI_ParamWriteBatch (ArgBlock "DeviceBatch") shall be treated as a job instance that shall not be interrupted by any SMI_DeviceWrite or SMI_DeviceRead service.

Prioritization of SMI services within the Standardized Master Interface is not performed. All services accessing methods will be processed in the order of their arrival (first come, first serve).

11.2.4 SMI_MasterIdentification

So far, an explicit identification of a Master did not have priority in SDCI since gateway applications usually provided hard-coded identification and maintenance information as required by the fieldbus system. Owing to the requirement "one Master Tool (PCDT) fits different Master brands", corresponding new Master Tools shall be able to connect to Masters providing an SMI. For that purpose, the SMI_MasterIdentification service has been created. It allows Master Tools to adjust to individual Master brands and types, if a particular fieldbus gateway provides the SMI services in a uniform accessible coding (see Clause 13). A class of Masters with a certain MasterID and VendorID shall not deviate in communication and functional behavior (Master type identification). Table 106 shows the service SMI_MasterIdentification.

Table 106 – SMI_MasterIdentification

Parameter name		.req	.cnf
Argument		M	
ClientID		M	
PortNumber	(0x00)	M	
ExpArgBlockID	(e.g. 0x0001)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber	(0x00)		M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber	(0x00)		M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

This parameter contains a virtual Port addressing the entire Master unit (0x00).

ExpArgBlockID

This parameter contains an ArgBlockID of the MasterIdent family, e.g. 0x0001 (see Table E.1).

ArgBlockLength

This parameter contains the length of the "VoidBlock" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Table E.2).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order (see Table C.3):
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length).

11.2.5 SMI_PortConfiguration

With the help of this service, an SMI client such as a gateway application launches the indicated Master Port and the connected Device using the elements in parameter PortConfigList. The service shall be accepted immediately and performed without delay. Content of Data Storage for that Port will be deleted at each new Port configuration via "DS_Delete" (see Figure 99). Table 107 shows the structure of the service. The ArgBlock usually is different in SDCI Extensions such as safety and wireless and specified there (see [10] and [11]).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 107 – SMI_PortConfiguration

Parameter name		.req	.cnf
Argument		M	
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(e.g. 0x8000)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x8000)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x8000)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains an ArgBlock of the PortConfigList family, e.g. 0x8000 (see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x8000).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (0xFFF0).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x8000).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:
PORT_NUM_INVALID (incorrect Port number),
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT (incorrect ArgBlock content type).

11.2.6 SMI_ReadbackPortConfiguration

This service allows for retrieval of the effective configuration of the indicated Master Port. Table 108 shows the structure of the service. This service usually is different in SDCI Extensions such as safety and wireless (see [10] and [11]).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 108 – SMI_ReadbackPortConfiguration

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(e.g. 0x8000)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains an ArgBlockID of the PortConfigList family, e.g. 0x8000 (see Table E.1).

ArgBlockLength

This parameter contains the length of the "VoidBlock" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.3).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:
PORT_NUM_INVALID (incorrect Port number),
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length).

11.2.7 SMI_PortStatus

This service allows for retrieval of the effective status of the indicated Master Port. Table 109 shows the structure of the service. This service usually is different in SDCI Extensions such as safety and wireless (see [10] and [11]).

Table 109 – SMI_PortStatus

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(e.g. 0x9000)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains an ArgBlockID of the PortStatusList family, e.g. 0x9000 (see Table E.1).

ArgBlockLength

This parameter contains the length of the "VoidBlock" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0)

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.4).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length).

11.2.8 SMI_DSToParServ

With the help of this service, an SMI client such as a gateway application is able to retrieve the technology parameter set of a Device from Data Storage and back it up within an upper-level parameter server (see Figure 95, 11.4, and 13.4.2). Table 110 shows the structure of the service.

In case of DI or DO on this Port, content of Data Storage is cleared. The same applies if Data Storage is not enabled for this Port.

Table 110 – SMI_DSToParServ

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(0x7000)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID 0x7000 (see Table E.1).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

This parameter contains the length of the "VoidBlock" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.6).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length).

11.2.9 SMI_ParServToDS

With the help of this service, an SMI client such as a gateway application is able to restore the technology parameter set of a Device within Data Storage from an upper-level parameter server (see Figure 95, 11.4, and 13.4.2). Table 111 shows the structure of the service.

In case Data Storage is not supported or not activated on this Port, the service will be replied with Result(-) INCONSISTENT_DS_DATA. The same applies if Data Storage is not consistent with Port configuration, e.g. VendorID does not match.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 111 – SMI_ParServToDS

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(0x7000)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x7000)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x7000)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains the ArgBlock DS_Data (0x7000, see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7000).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID.

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7000).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:
PORT_NUM_INVALID (incorrect Port number),
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT (incorrect ArgBlock content type),
INCONSISTENT_DS_DATA (inconsistent Data Storage data).

11.2.10 SMI_DeviceWrite

This service allows for writing On-request Data (OD) for propagation to the Device. Table 112 shows the structure of the service.

Table 112 – SMI_DeviceWrite

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(0x3000)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x3000)		M
ArgBlockLength			M
ArgBlock	(associated to the ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x3000)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains the ArgBlock "On-requestData" (0x3000, see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x3000).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID.

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x3000).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT	(incorrect ArgBlock content type),
SERVICE_TEMP_UNAVAILABLE	(Master busy),
DEVICE_NOT_ACCESSIBLE	(Device not communicating),
Device ErrorType	(See Clauses C.2 and C.3).

11.2.11 SMI_DeviceRead

This service allows for reading On-request Data (OD) from the Device via the Master. Table 113 shows the structure of the service.

Table 113 – SMI_DeviceRead

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(0x3000)	M	
ArgBlockLength		M	
ArgBlock	("On-request Data/Index": 0x3001)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x3001)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x3001)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ExpArgBlockID

This parameter contains the ArgBlockID of "On-requestData" (0x3000, see Table E.1).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock "On-requestData/Index" (0x3001, see Clause E.5).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x3001).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Table E.5).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x3001).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT	(incorrect ArgBlock content type),
SERVICE_TEMP_UNAVAILABLE	(Master busy),
DEVICE_NOT_ACCESSIBLE	(Device not communicating),
Device ErrorType	(See Clauses C.2 and C.3).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.2.12 SMI_ParamWriteBatch

This service allows for the "push" transfer of a large number of consistent Device objects via multiple ISDUs. Table 114 shows the structure of the service. The following rules apply:

– The service transfers the ArgBlock "DeviceParBatch" to the Master that conveys the content object by object to the Device via AL_Write (ISDU).

– The same ArgBlock structure is returned as Result (+). However, a value "0x0000" indicates success of a particular AL_Write or an ISDU ErrorType of a failed AL_Write instead of a parameter record.

– Result (-) is only returned in case of a failing service via "JobError".

NOTE 1 This service supposes the use of Block Parameterization and sufficient buffer ressources.

NOTE 2 This service may have an unexpected longer duration.

This service is optional. Availability is indicated via Master identification (see Table E.2).

Table 114 – SMI_ParamWriteBatch

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	("DeviceParBatch": 0x7001)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x7001)		M
ArgBlockLength			M
ArgBlock	(associated to the ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x7001)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains the ArgBlock "DeviceParBatch" (0x7001, see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7001).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Table E.7).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7001).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:
SERVICE_NOT_SUPPORTED (Service unknown),
PORT_NUM_INVALID (incorrect Port number),
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT (incorrect ArgBlock content type),
MEMORY_OVERRUN (insufficient memory),
SERVICE_TEMP_UNAVAILABLE (Master busy),
DEVICE_NOT_ACCESSIBLE (Device not communicating).

11.2.13 SMI_ParamReadBatch

This service allows for the "pull" transfer of a large number of consistent Device parameters via multiple ISDUs. Table 114 shows the structure of the service. The following rules apply:

– The service transfers the ArgBlock "IndexList" to the Master that transforms the content entry by entry into AL_Read (ISDU) to the Device.
– The corresponding ArgBlock "DeviceParBatch is returned as Result (+). In case of a successful AL_Read of an object, the corresponding parameter record or an ISDU ErrorType of a failed AL_Read instead of a parameter record is returned.
– Result (-) is only returned in case of a failing service via "JobError".

NOTE1 This service supposes the use of Block Parameterization and sufficient buffer ressources.

NOTE 2 This service may have an unexpected longer duration.

This service is optional. Availability is indicated via Master identification (see Table E.2).

Table 115 – SMI_ParamReadBatch

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	("DeviceParBatch": 0x7001)	M	
ArgBlockLength		M	
ArgBlock	("IndexList": 0x7002)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x7002)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x7002)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID of "DeviceParBatch" (0x7001, see Table E.1).

ArgBlockLength

This parameter contains the length of the ArgBlock "IndexList".

ArgBlock

This parameter contains the ArgBlock "IndexList" (0x7002, see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7002).

ArgBlockLength

This parameter contains the conditional length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Table E.7).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7002).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

SERVICE_NOT_SUPPORTED	(Service unknown),
PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT	(incorrect ArgBlock content type),
MEMORY_OVERRUN	(insufficient memory),
SERVICE_TEMP_UNAVAILABLE	(Master busy),
DEVICE_NOT_ACCESSIBLE	(Device not communicating).

11.2.14 SMI_PortPowerOffOn

This service allows for switching Power 1 of a particular Port off and on (see 5.4.1). It returns upon the elapsed time provided within the ArgBlock. Table 116 shows the structure of the service.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 116 – SMI_PortPowerOffOn

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	("PortPowerOffOn": 0x7003)	M	
Result (+)			S
ClientID			M
PortNumber			M
ExpArgBlockID	(ID of request ArgBlock 0x7003)		M
ArgBlockLength			M
ArgBlock	(associated to the ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
ExpArgBlockID	(ID of request ArgBlock 0x7003)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains the ArgBlock "PortPowerOffOn" (0x7003, see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7003).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (0xFFF0).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x7003).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:
PORT_NUM_INVALID (incorrect Port number),
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT (incorrect ArgBlock content type),
SERVICE_TEMP_UNAVAILABLE (Master busy).

11.2.15 SMI_DeviceEvent

This service allows for signaling a Master Event created by the Device. Table 117 shows the structure of the service.

Table 117 – SMI_DeviceEvent

Parameter name		.ind	.rsp
Argument			
ClientID	(= "0" → Broadcast)	M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	("DeviceEvent": 0xA000)	M	
Acknowledgment			S
ClientID	(= "0")		M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xA000)		M
ArgBlockLength			M
ArgBlock	(VoidBlock: 0xFFF0)		M

Argument

The specific parameters of this indication are transmitted in the argument (see 11.2.2).

ClientID

For this indication, the ClientID shall be "0" ("broadcast" to upper-level system).

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the reported ArgBlock 0xA000.

ArgBlock

This parameter contains the ArgBlock "DeviceEvent" (0xA000, see Table E.1).

Acknowlegment

This selection parameter indicates that the service request has been executed successfully.

ClientID

The ClientID shall be "0".

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xA000).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (0xFFF0).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.2.16 SMI_PortEvent

This service allows for signaling a Master Event created by the Port. Table 118 shows the structure of the service.

Table 118 – SMI_PortEvent

Parameter name		.ind	.rsp
Argument			
ClientID	(= "0" → Broadcast)	M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(PortEvent: 0xA001)	M	
Acknowledgment			S
ClientID	(= "0")		M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xA001)		M
ArgBlockLength			M
ArgBlock	(VoidBlock: 0xFFF0)		M

Argument

The specific parameters of this indication are transmitted in the argument (see 11.2.2).

ClientID

For this indication, the ClientID shall be "0" ("broadcast" to upper-level system).

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the reported ArgBlock 0xA001.

ArgBlock

This parameter contains the ArgBlock "PortEvent" (0xA001, see Table E.1).

Acknowlegment

This selection parameter indicates that the service request has been executed successfully.

ClientID

The ClientID shall be "0".

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xA001).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (0xFFF0).

11.2.17 SMI_PDIn

This service allows for cyclically reading input Process Data from an InBuffer (see 11.7.2.1). Table 119 shows the structure of the service. This service usually has companion services in SDCI Extensions such as safety and wireless (see [10] and [11]).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 119 – SMI_PDIn

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(e.g. 0x1001)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains an ArgBlockID of the Process Data family, e.g. 0x1001 (see Table E.1).

ArgBlockLength

This parameter contains the length of the "VoidBlock" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock: PDIn

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.10).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:
PORT_NUM_INVALID (incorrect Port number),
ARGBLOCK_NOT_SUPPORTED (ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID (incorrect ArgBlock length),
DEVICE_NOT_IN_OPERATE (Process Data not accessible).

11.2.18 SMI_PDOut

This service allows for cyclically writing output Process Data to an OutBuffer (see 11.7.3.1). Table 120 shows the structure of the service. This service usually has companion services in SDCI Extensions such as safety and wireless (see [10] and [11]).

Table 120 – SMI_PDOut

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(e.g. 0x1002)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x1002)		M
ArgBlockLength			M
ArgBlock	(VoidBlock: 0xFFF0)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x1002)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains ArgBlock of the Process Data family, e.g. 0x1002 (see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x1002).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (0xFFF0).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x1002).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT	(incorrect ArgBlock content type),
DEVICE_NOT_IN_OPERATE	(Process Data not accessible).

11.2.19 SMI_PDInOut

This service allows for periodically reading input from an InBuffer (see 11.7.2.1) and periodically reading output Process Data from an OutBuffer (see 11.7.3.1). Table 121 shows the structure of the service. This service usually has companion services in SDCI Extensions such as safety and wireless (see [10] and [11]).

Table 121 – SMI_PDInOut

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(e.g. 0x1003)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains an ArgBlockID of the "Process Data" family, e.g. 0x1003 (see Table E.1).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.12).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length),
DEVICE_NOT_IN_OPERATE	(Process Data not accessible).

11.2.20 SMI_PDInIQ

This service allows for cyclically reading input Process Data from an InBuffer (see 11.7.2.1) containing the value of the input "I" signal (Pin 2 at M12). Table 122 shows the structure of the service.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 122 – SMI_PDInIQ

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(e.g. 0x1FFE)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains an ArgBlockID of the "Process Data" family, e.g. 0x1FFE (see Table E.1).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.13).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

SERVICE_NOT_SUPPORTED	(Service unknown),
PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length).

11.2.21 SMI_PDOutIQ

This service allows for cyclically writing output Process Data to an OutBuffer (see 11.7.3.1), which contains the value of the output "Q" signal (Pin 2 at M12). Table 123 shows the structure of the service.

Table 123 – SMI_PDOutIQ

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(e.g. 0x1FFF)	M	
Result (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x1FFF)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID of request ArgBlock 0x1FFF)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains the ArgBlockID "VoidBlock" (0xFFF0, see Clause E.17).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock to be "pushed".

ArgBlock

This parameter contains an ArgBlock of the "Process Data" family, e.g. 0x1FFF (see Table E.1).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x1FFF).

ArgBlockLength

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (0xFFF0).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0x1FFF).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

SERVICE_NOT_SUPPORTED	(Service unknown),
PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length),
ARGBLOCK_INCONSISTENT	(incorrect ArgBlock content type).

11.2.22 SMI_PDReadbackOutIQ

This service allows for cyclically reading back input Process Data from an OutBuffer (see 11.7.3.1) containing the value of the output "Q" signal (Pin 2 at M12). Table 124 shows the structure of the service.

Table 124 – SMI_PDReadbackOutIQ

Parameter name		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(e.g. 0x1FFF)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Result (+)			S
ClientID			M
PortNumber			M
ExpArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associated to ExpArgBlockID)		M
Result (−)			S
ClientID			M
PortNumber			M
ExpArgBlockID	(ID of request ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

The specific parameters of the service request are transmitted in the argument (see 11.2.2).

ClientID

PortNumber

ExpArgBlockID

This parameter contains an ArgBlockID of the "Process Data" family, e.g. 0x1FFF (see Table E.1).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFF0, see Clause E.17).

Result (+):

This selection parameter indicates that the service request has been executed successfully.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock.

ArgBlock: PDOutIQ

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Clause E.14).

Result (−):

This selection parameter indicates that the service request failed.

ClientID

PortNumber

RefArgBlockID

This parameter contains as reference the ID of the ArgBlock sent by the request (0xFFF0).

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock.

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see Clause E.18).

Permitted values in prioritized order:

SERVICE_NOT_SUPPORTED	(Service unknown),
PORT_NUM_INVALID	(incorrect Port number),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock unknown),
ARGBLOCK_LENGTH_INVALID	(incorrect ArgBlock length).

11.3 Configuration Manager (CM)

11.3.1 Coordination of Master applications

Figure 99 illustrates the coordination between Master applications. The main responsibility is assigned to the Configuration Manager (CM), who initializes Port start-ups and who starts or stops the other Master applications depending on a respective Port state.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

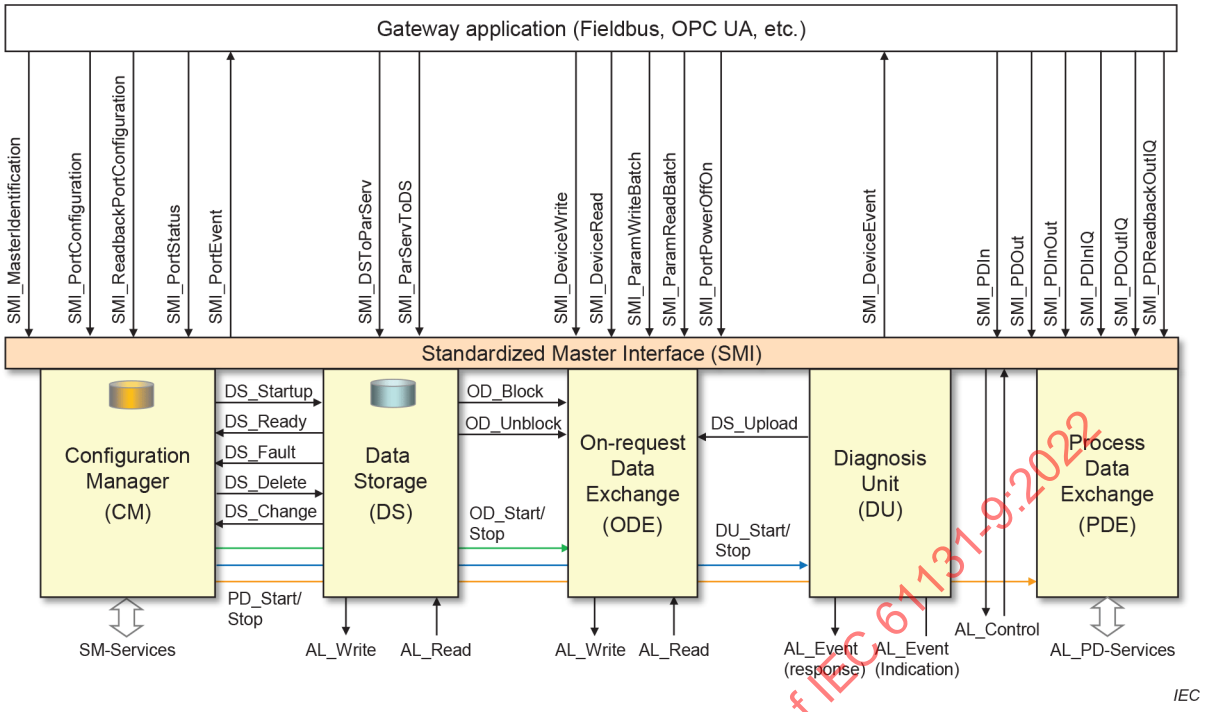


Figure 99 – Coordination of Master applications

Internal variables and Events controlling Master applications are listed in Table 125.

Table 125 – Internal variables and Events controlling Master applications

Internal Variable	Definition
DS_Startup	This variable triggers the Data Storage (DS) state machine causing an Upload or Download of Device parameters if required (see 11.4).
DS_Ready	This variable indicates the Data Storage has been accomplished successfully; operating mode is CFGCOM or AUTOCOM (see 9.2.2.2).
DS_Fault	This variable indicates the Data Storage has been aborted due to a fault.
DS_Delete	Any verified change of Device configuration leads to a deletion of the stored data set in the Data Storage.
DS_Change	This variable indicates a content change of Data Storage triggered by service SMI_ParServToDS.
DS_Upload	This variable triggers the Data Storage state machine in the Master due to the special Event "DS_UPLOAD_REQ" from the Device.
OD_Start	This variable enables On-request Data access via AL_Read and AL_Write.
OD_Stop	This variable indicates that On-request Data access via AL_Read and AL_Write is acknowledged with a negative response to the gateway application.
OD_Block	Data Storage upload and download actions disable the On-request Data access through AL_Read or AL_Write. Access by the gateway application is denied.
OD_Unblock	This variable enables On-request Data access via AL_Read or AL_Write.
DU_Start	This variable enables the Diagnosis Unit to propagate remote (Device) Events to the gateway application.
DU_Stop	This variable indicates that the Device Events are not propagated to the gateway application and not acknowledged. Available Events are blocked until the DU is enabled again.
PD_Start	This variable enables the Process Data exchange with the gateway application.
PD_Stop	This variable disables the Process Data exchange with the gateway application.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Restart of a Port is basically driven by two activities:

– SMI_PortConfiguration service (Port parameter setting and start-up or changes and restart of a Port);
– SMI_ParServToDS service (Download of Data Storage data if Data Storage is activated).

The Configuration Manager (CM) is launched upon reception of a "SMI_PortConfiguration" service. The elements of parameter "PortConfigList" are stored in non-volatile memory within the Master. The service "SMI_ReadbackPortConfiguration" allows for checking correct storage.

The CM uses the values of ArgBlock "PortConfigList", initializes the Port start-up in case of value changes and empties the Data Storage via "DS_Delete" or checks emptiness (see Figure 99).

A gateway application can poll the actual Port state via "SMI_PortStatus" to check whether the expected Port state is reached. In case of fault this service provides the corresponding information.

After successfully setting up the Port, the CM starts the Data Storage mechanism and returns via the parameter element "PortStatusInfo" either "OPERATE" or "PORT_FAULT" to the gateway application.

In case of "OPERATE", the CM activates the state machines of the associated Master applications Diagnosis Unit (DU), On-request Data Exchange (ODE), and Process Data Exchange (PDE).

In case of a fault in SM_PortMode such as COMP_FAULT, REVISION_FAULT, or SERNUM_FAULT according to 9.2.3, only the ODE state machine shall be activated to allow for parameterization.

Figure 100 illustrates the start-up of a Port via SMI_ PortConfiguration service in a sequence diagram.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

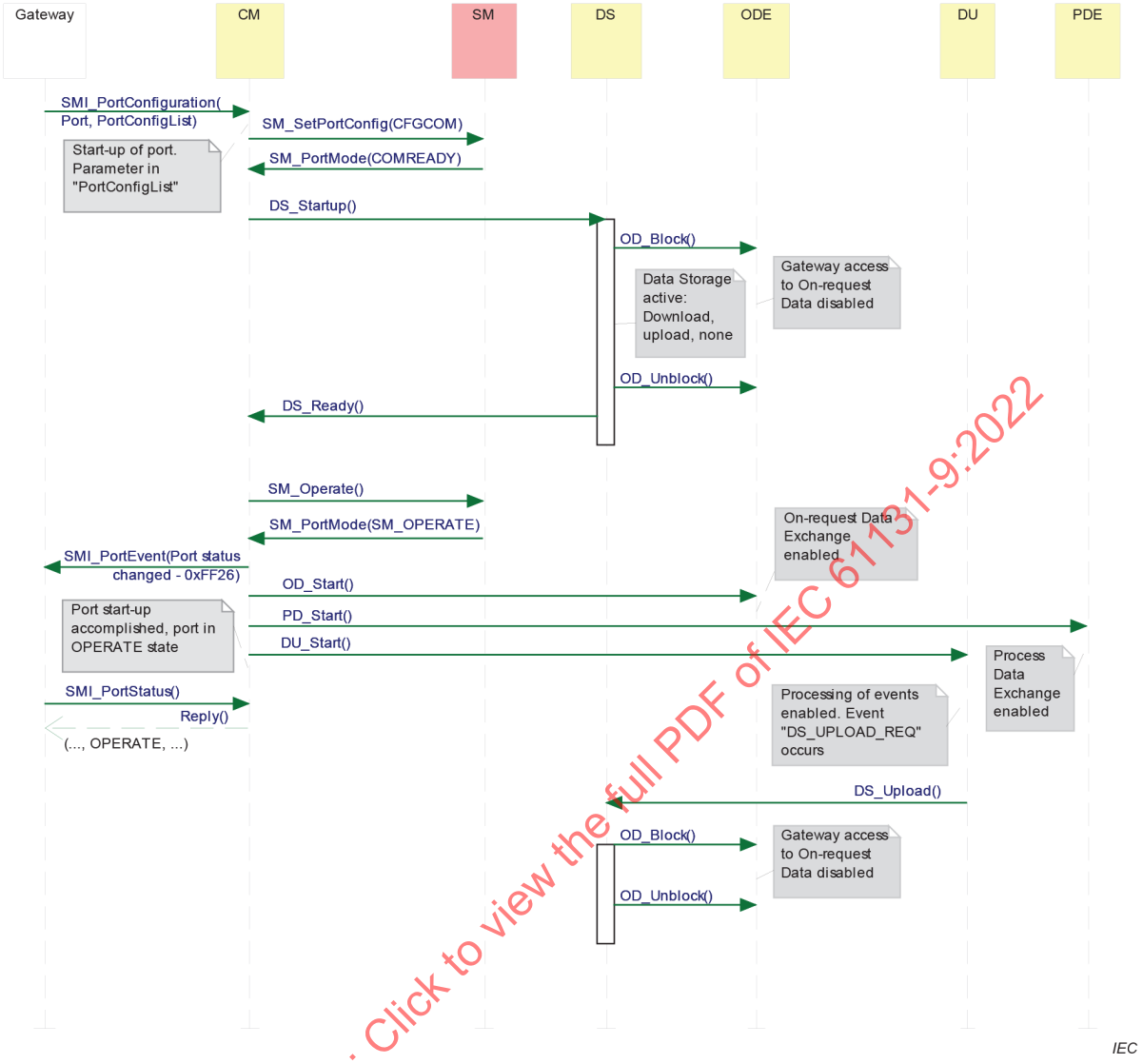


Figure 100 – Sequence diagram of start-up via Configuration Manager

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.3.2 State machine of the Configuration Manager

Figure 101 shows the state machine of the Configuration Manager. In general, states and transitions correspond to those of the message handler: STARTUP, PREOPERATE (fault or Data Storage), and at the end OPERATE. Dedicated "SM_PortMode" services drive the transitions (see 9.2.2.4). A special state is related to SIO mode DI or DO.

The Configuration Manager can receive the information COMLOST from Port x Handler through "SM_PortMode" at any time. It also can receive a service "SMI_PortConfiguration" from the gateway application with changed values in "PortConfigList" at any time (see 11.2.5).

Via the service "SMI_ParServToDS", it also can receive a Data Storage object with a changed parameter set from the gateway application triggering action in the Configuration Manager if Data Storage is activated.

Port x is started/restarted in all cases.

Figure 101 together with Table 126 also shows transitions leading to corresponding changes in "PortStatusInfo" of ArgBlock "PortStatusList" (see Table E.4). Based on these transitions, Events are triggered via SMI_PortEvent. For details see Clause D.3.

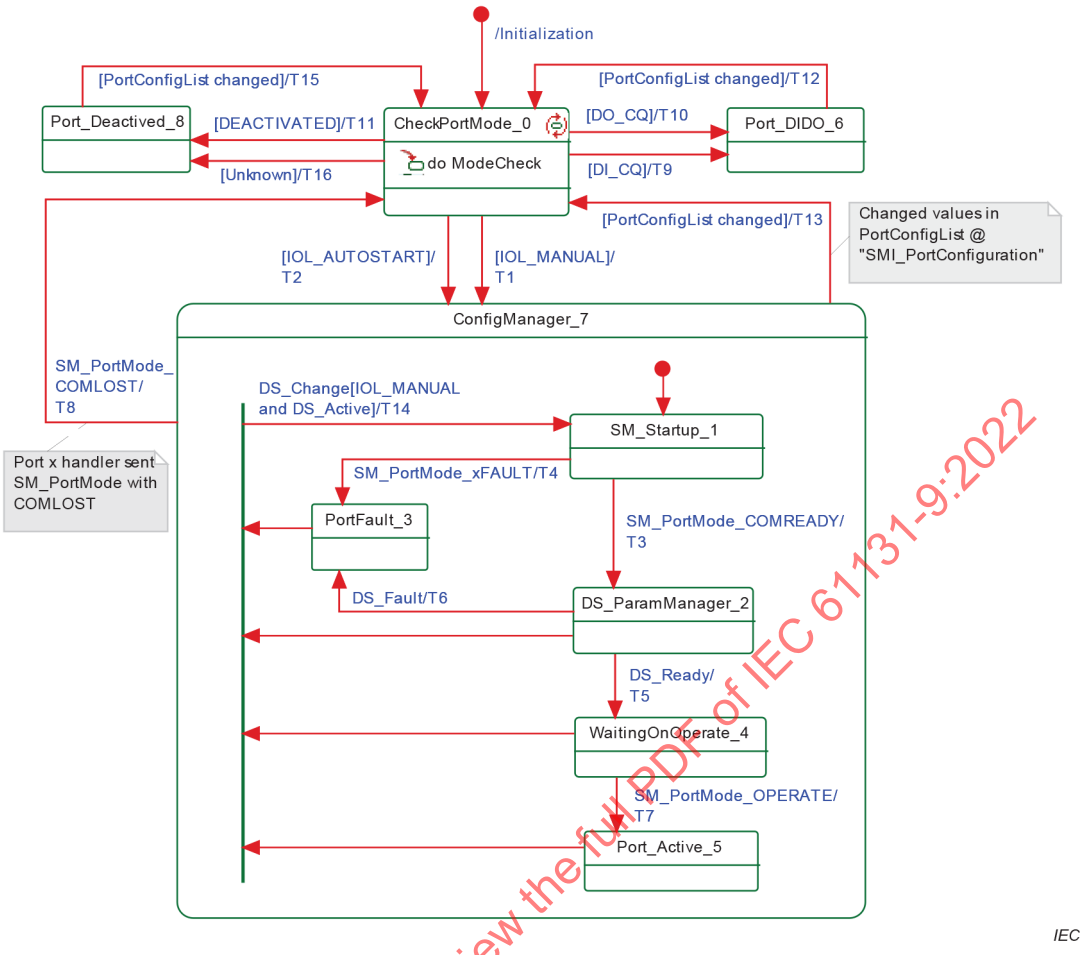


Figure 101 – State machine of the Configuration Manager

Table 126 shows the state transition table of the Configuration Manager.

Table 126 – State transition table of the Configuration Manager

STATE NAME	STATE DESCRIPTION
CheckPortMode_0	Check "Port Mode" element in parameter "PortConfigList" (see 11.2.5).
SM_Startup_1	Waiting on an established communication or loss of communication or any of the faults REVISION_FAULT, COMP_FAULT, or SERNUM_FAULT (see Table 85).
DS_ParamManager_2	Waiting on accomplished Data Storage startup. Parameter are downloaded into the Device or uploaded from the Device.
PortFault_3	Device in state PREOPERATE (communicating). However, one of the three faults REVISION_FAULT, COMP_FAULT, SERNUM_FAULT, or DS_Fault, or PORT_DIAG occurred.
WaitingOnOperate_4	Waiting on SM to switch to OPERATE.
Port_Active_5	Port is in OPERATE mode. The gateway application is exchanging Process Data and ready to send or receive On-request Data.
Port_DIDO_6	Port is in DI or DO mode. The gateway application is exchanging Process Data (DI or DO).
ConfigManager_7	This superstate handles Port communication operations and allows all states inside to react on COMLOST via SM_PortMode service. A Port restart is managed inside the superstate triggered by the DS_Change signal (see Table 125).
Port_Deactivated_8	Port is in DEACTIVATED mode.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	7	Invoke DS-Delete if identification (VendorID, DeviceID) within DS is different from configured Port identification. SM_SetPortConfig_CFGCOM
T2	0	7	Invoke DS-Delete. SM_SetPortConfig_AUTOCOM
T3	1	2	DS_Startup: The DS state machine is triggered. Update parameter elements of "PortStatusList": - PortStatusInfo = NOT_AVAILABLE - RevisionID = (real) RRID - Transmission rate = COMx - VendorID = (real) RVID - DeviceID = (real) RDID - MasterCycleTime = value - Port QualityInfo = invalid
T4	1	3	Update parameter elements of "PortStatusList": - PortStatusInfo = PORT_DIAG - RevisionID = (real) RRID - Transmission rate = COMx - VendorID = (real) RVID - DeviceID = (real) RDID - Port QualityInfo = invalid
T5	2	4	SM_Operate
T6	2	3	Data Storage failed. Rollback to previous parameter set. Update parameter elements of "PortStatusList": - PortStatusInfo = PORT_DIAG - RevisionID = (real) RRID - Transmission rate = COMx - VendorID = (real) RVID - DeviceID = (real) RDID - Port QualityInfo = invalid
T7	4	5	Update parameter elements of "PortStatusList": - PortStatusInfo = OPERATE - RevisionID = (real) RRID - Transmission rate = COMx - VendorID = (real) RVID - DeviceID = (real) RDID - Port QualityInfo = x
T8	1,2,3,4,5	0	Update parameter elements of "PortStatusList": - PortStatusInfo = NO_DEVICE - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).
T9	0	6	Invoke DS-Delete. SM_SetPortConfig_DI. Update parameter elements of "PortStatusList": - PortStatusInfo = DI_C/Q - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T10	0	6	Invoke DS-Delete. SM_SetPortConfig_DO. Update parameter elements of "PortStatusList": - PortStatusInfo = DO_C/Q - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).
T11	0	8	Invoke DS-Delete. SM_SetPortConfig_INACTIVE. Update parameter elements of "PortStatusList": - PortStatusInfo = DEACTIVATED - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).
T12	6	0	Update parameter elements of "PortStatusList": - PortStatusInfo = NOT_AVAILABLE - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).
T13	1,2,3,4,5	0	Update parameter elements of "PortStatusList": - PortStatusInfo = NOT_AVAILABLE - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).
T14	1,2,3,4,5	1	SM_SetPortConfig_CFGCOM Update parameter elements of "PortStatusList": - PortStatusInfo = NOT_AVAILABLE - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).
T15	8	0	Update parameter elements of "PortStatusList": - PortStatusInfo = NOT_AVAILABLE - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T16	0	8	Invoke DS-Delete. SM_SetPortConfig_INACTIVE. Update parameter elements of "PortStatusList": - PortStatusInfo = DEACTIVATED - RevisionID = 0 - Transmission rate = 0 - VendorID = 0 - DeviceID = 0 - Port QualityInfo = invalid Delete DiagEntries (SOURCE = Device) in PortStatusList (see Table E.4).

INTERNAL ITEMS	TYPE	DEFINITION
PortConfigList changed	Guard	Values of "PortConfigList" have changed.
DS_Ready	Signal	Data Storage sequence (upload, download) accomplished; see Table 125.
DS_Fault	Signal	See Table 125
DEACTIVATED	Guard	See Table E.3
IOL_MANUAL	Guard	See Table E.3
IOL_AUTOSTART	Guard	See Table E.3
DI_C/Q	Guard	See Table E.3
DO_C/Q	Guard	See Table E.3
DS_Change	Signal	See Table 125
DS_Active	Guard	Port configured to "Backup + Restore" (3) or "Restore" (4); see Table E.3.

State "CheckPortMode_0" contains an activity with complex logic for checking the Port mode within a received Port configuration (see Table E.3). Figure 102 shows this activity within the context of the state machine in Figure 101.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

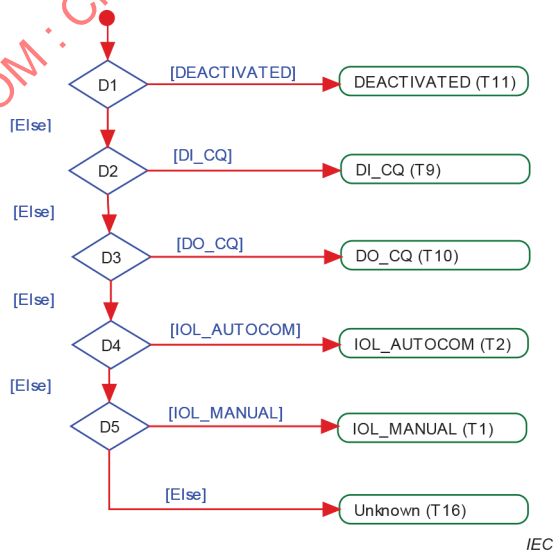


Figure 102 – Activity for state "CheckPortMode_0"

11.4 Data Storage (DS)

11.4.1 Overview

Data Storage between Master and Device is specified within this document, whereas the adjacent upper Data Storage mechanisms depend on the individual fieldbus or system. The Device holds a standardized set of objects providing parameters for Data Storage, memory size requirements, control and state information of the Data Storage mechanism. Changes of Data Storage parameter sets are detectable via the "Parameter Checksum" (see 10.4.8).

11.4.2 DS data object

The structure of a Data Storage data object is specified in Table G.1.

The Master shall always hold the header information (Parameter Checksum, VendorID, and DeviceID) for the purpose of checking and control. The object information (objects 1…n) will be stored within the non-volatile memory part of the Master (see Annex G). Prior to a download of the Data Storage data object (parameter block), the Master will check the consistency of the header information with the particular Device.

The maximum permitted size of the Data Storage data object is 2×2^{10} octets. It is mandatory for Masters to provide at least this memory space per Port if the Data Storage mechanism is implemented.

11.4.3 Backup and Restore

Gateways are able to retrieve a Port's current Data Storage object out of the Master using the service "SMI_DSToParServ", see 11.2.8.

In return, gateways are also able to write a Port's current Data Storage object into the Master using the service "SMI_ParServToDS" (see 11.2.9). This causes under certain conditions an implicit restart of the Device and activation of the parameters within the Device (see 11.3.2).

11.4.4 DS state machine

The Data Storage mechanism is called right after establishing the COMx communication, before entering the OPERATE mode. During this time any other communication with the Device shall be rejected by the gateway.

Figure 103 shows the state machine of the Data Storage mechanism.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

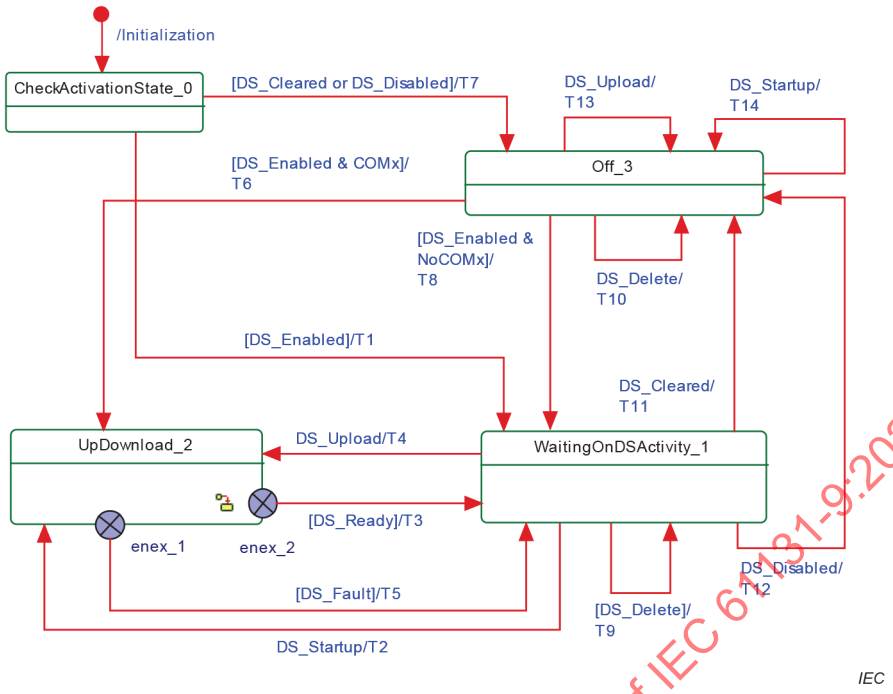


Figure 103 – Main state machine of the Data Storage mechanism

The internal parameter "ActivationState" (DS_Enabled, DS_Disabled, and DS_Cleared) is derived from the parameter "Backup behavior" in "SMI_PortConfiguration" service (see 11.2.5 and Table 127 / INTERNAL ITEMS).

Figure 104 shows the submachine of the state "UpDownload_2".

This submachine can be invoked by the Data Storage mechanism or during runtime triggered by a "DS_UPLOAD_REQ" Event..

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

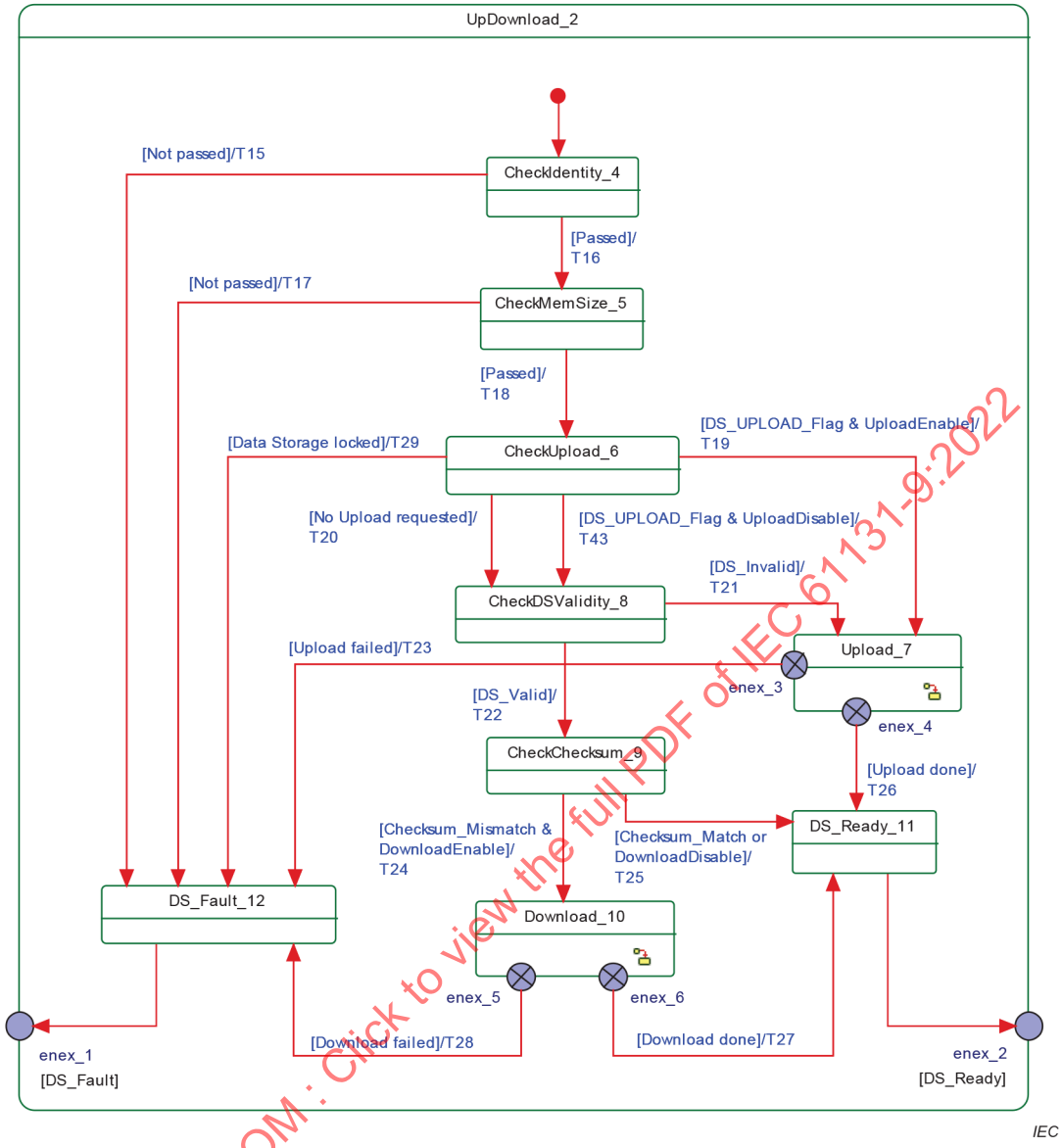


Figure 104 – Submachine "UpDownload_2" of the Data Storage mechanism

Figure 105 shows the submachine of the state "Upload_7".

This state machine can be invoked by the Data Storage mechanism or during runtime triggered by a DS_UPLOAD_REQ Event.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

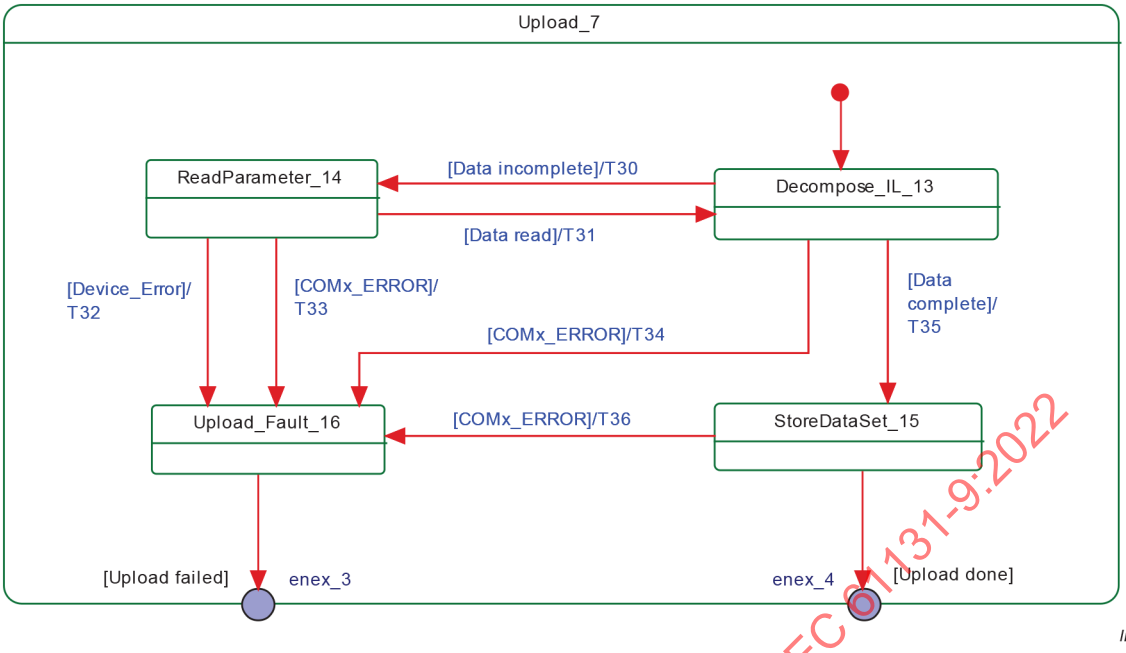


Figure 105 – Data Storage submachine "Upload_7"

Figure 106 demonstrates the Data Storage upload sequence using the DataStorageIndex (DSI) specified in B.2.3 and Table B.10. The structure of Index_List is specified in Table B.11. The DS_UPLOAD_FLAG shall be reset at the end of each sequence (see Table B.10).

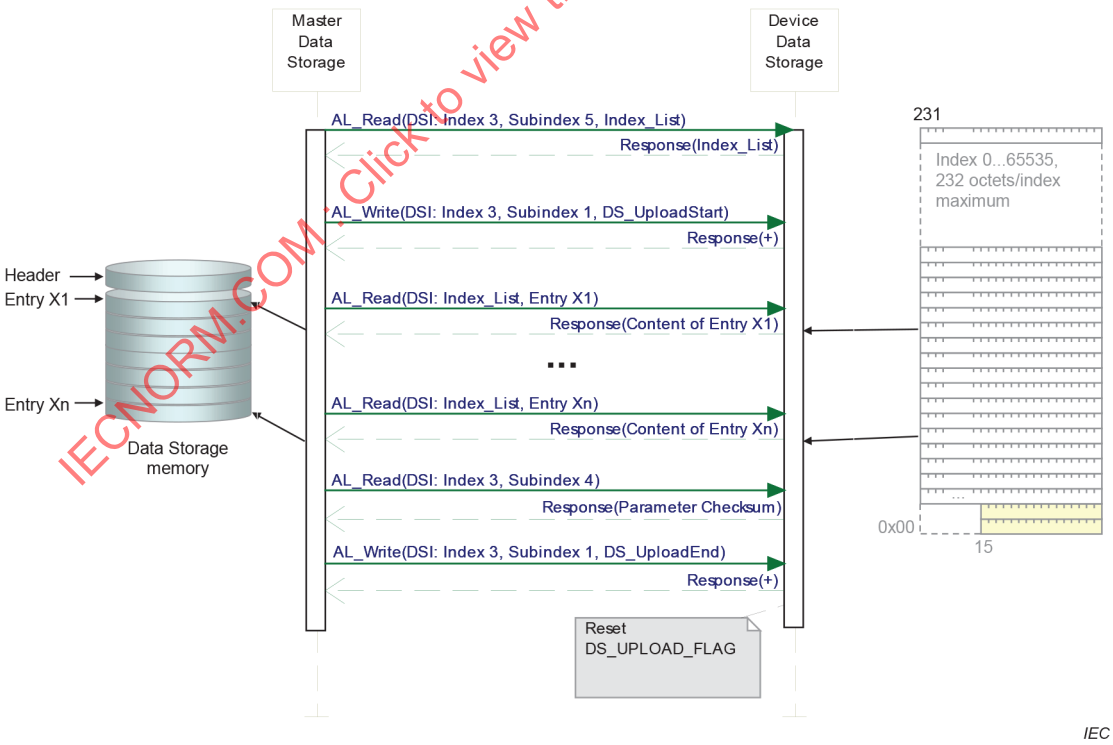


Figure 106 – Data Storage upload sequence diagram

Figure 107 shows the submachine of the state "Download_10".

This state machine can be invoked by the Data Storage mechanism.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

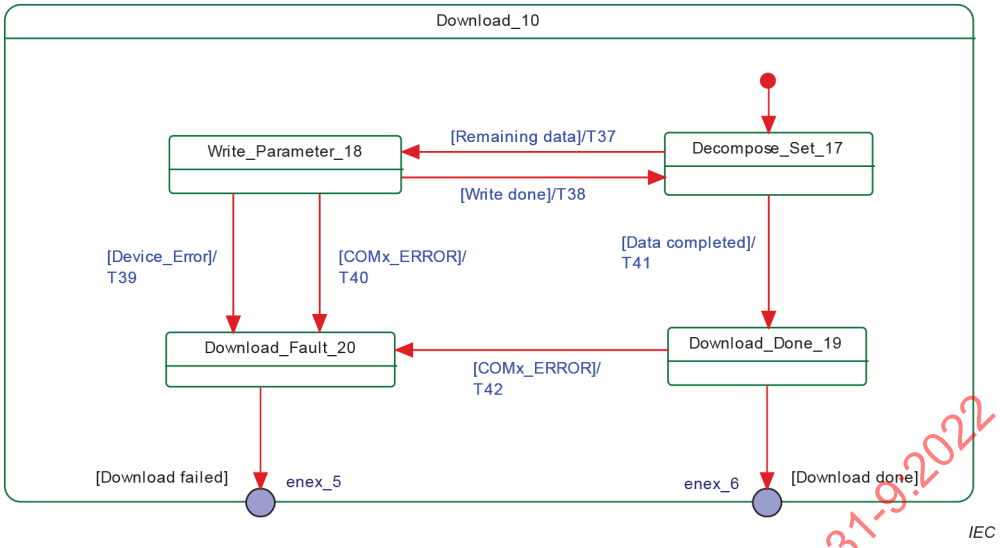


Figure 107 – Data Storage submachine "Download_10"

Figure 108 demonstrates the Data Storage download sequence using the DataStorageIndex (DSI) specified in B.2.3 and Table B.10. The structure of Index_List is specified in Table B.11. The DS_UPLOAD_FLAG shall be reset at the end of each sequence (see Table B.10).

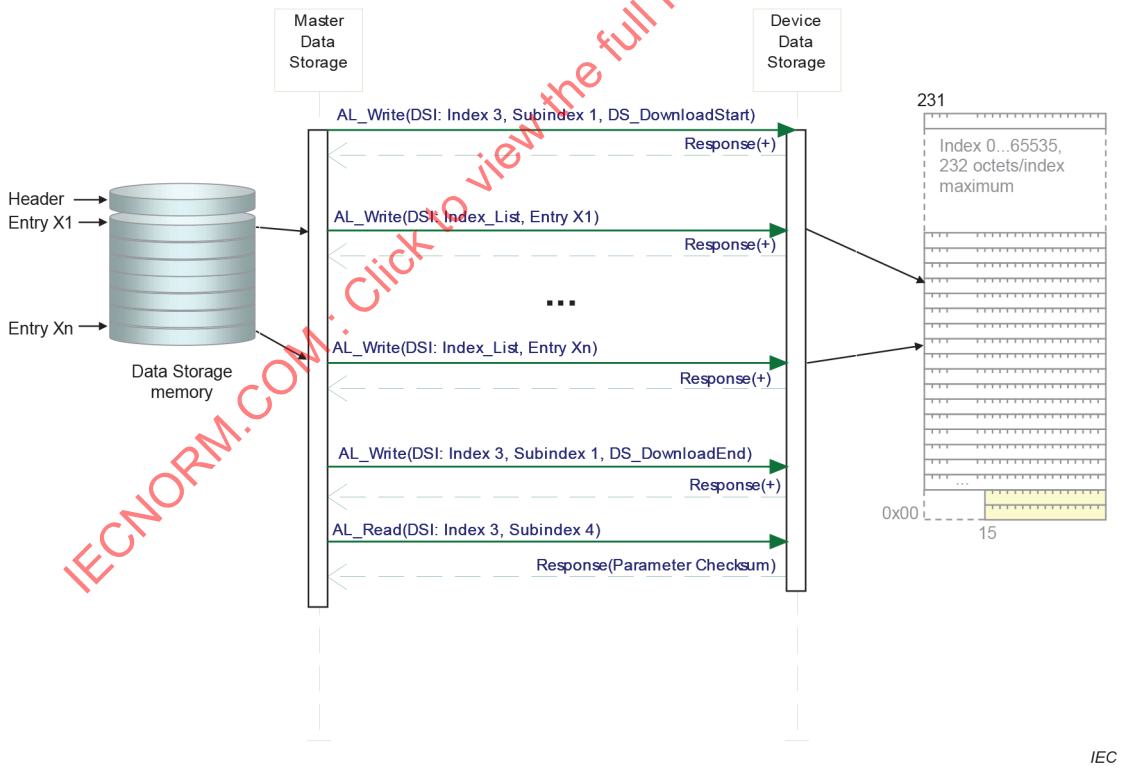


Figure 108 – Data Storage download sequence diagram

Table 127 shows the states and transitions of the Data Storage state machines.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table 127 – States and transitions of the Data Storage state machines

STATE NAME	STATE DESCRIPTION
CheckActivationState_0	Check current state of the DS configuration: Independently from communication status, DS_Startup from configuration management or an Event DS_UPLOAD_REQ is expected.
WaitingOnDSActivity_1	Waiting for upload request, Device startup, all changes of activation state independent of the Device communication state.
UpDownload_2	Submachine for up/download actions and checks
Off_3	Data Storage handling switched off or deactivated
SM: CheckIdentity_4	Check Device identification (DeviceID, VendorID) against parameter set within the Data Storage (see Table G.2). Empty content does not lead to a fault.
SM: CheckMemSize_5	Check data set size (Index 3, Subindex 3) against available Master storage size
SM: CheckUpload_6	Check for DS_UPLOAD_FLAG within the DataStorageIndex (see Table B.10)
SM: Upload_7	Submachine for the upload actions
SM: CheckDSValidity_8	Check whether stored data within the Master is valid or invalid. A Master could be replaced between upload and download activities. It is the responsibility of a Master designer to implement a validity mechanism according to the chosen use cases
SM: CheckChecksum_9	Check for differences between the data set content and the Device parameter via the "Parameter Checksum" within the DataStorageIndex (see Table B.10)
SM: Download_10	Submachine for the download actions
SM: DS_Ready_11	Prepare DS_Ready indication to the Configuration Management (CM)
SM: DS_Fault_12	Prepare DS_Fault indication from "Identification_Fault", "SizeCheck_Fault", "Upload_Fault", and "Download_Fault" to the Configuration Management (CM)
SM: Decompose_IL_13	Read Index List within the DataStorageIndex (see Table B.10). Read content entry by entry of the Index List from the Device (see Table B.11).
SM: ReadParameter_14	Wait until read content of one entry of the Index List from the Device is accomplished.
SM: StoreDataSet_15	Task of the gateway application: store entire data set according to Table G.1 and Table G.2
SM: Upload_Fault_16	Prepare Upload_Fault indication from "Device_Error" and "COM_ERROR" as input for the higher-level indication DS_Fault.
SM: Decompose_Set_17	Write parameter by parameter of the data set into the Device according to Table G.1.
SM: Write_Parameter_18	Wait until write of one parameter of the data set into the Device is accomplished.
SM: Download_Done_19	Download completed. Read back "Parameter Checksum" from the DataStorageIndex according to Table B.10. Save this value in the stored data set according to Table G.2.
SM: Download_Fault_20	Prepare Download_Fault indication from "Device_Error" and "COM_ERROR" as input for the higher-level indication DS_Fault.

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	–
T2	1	2	–
T3	2	1	OD_Unblock; Indicate DS_Ready to CM
T4	1	2	Confirm Event "DS_Upload" (see INTERNAL ITEMS)
T5	2	1	DS_Break (AL_Write, Index 3, Subindex 1); clear intermediate data (garbage collection); rollback to previous parameter state; DS_Fault (see Figure 98); OD_Unblock.
T6	3	2	–
T7	0	3	–
T8	3	1	–
T9	1	1	Clear saved parameter set (see Table G.1 and Table G.2)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T10	3	3	Clear saved parameter set (see Table G.1 and Table G.2)
T11	1	3	Clear saved parameter set (see Table G.1 and Table G.2)
T12	1	3	–
T13	3	3	Confirm Event "DS_Upload" (see INTERNAL ITEMS); no further action
T14	3	3	DS_Ready to CM
T15	4	12	Indicate DS_Fault(Identification_Fault) to the gateway application
T16	4	5	Read "Data Storage Size" according to Table B.10, OD_Block
T17	5	12	Indicate DS_Fault(SizeCheck_Fault) to the gateway application
T18	5	6	Read "DS_UPLOAD_FLAG" according to Table B.10
T19	6	7	DataStorageIndex 3, Subindex 1: "DS_UploadStart" (see Table B.10)
T20	6	8	–
T21	8	7	DataStorageIndex 3, Subindex 1: "DS_UploadStart" (see Table B.10)
T22	8	9	–
T23	7	12	DataStorageIndex 3, Subindex 1: "DS_Break" (see Table B.10). Indicate DS_Fault(Upload) to the gateway application
T24	9	10	DataStorageIndex 3, Subindex 1: "DS_DownloadStart" (see Table B.10)
T25	9	11	–
T26	7	11	DataStorageIndex 3, Subindex 1: "DS_UploadEnd"; read Parameter Checksum (see Table B.10)
T27	10	11	–
T28	10	12	DataStorageIndex 3, Subindex 1: "DS_Break" (see Table B.10). Indicate DS_Fault(Download) to the gateway application.
T29	6	12	Indicate DS_Fault(Data Storage locked) to the gateway application
T30	13	14	AL_Read (Index List)
T31	14	13	–
T32	14	16	–
T33	14	16	–
T34	13	16	–
T35	13	15	Read "Parameter Checksum" (see Table B.10).
T36	15	16	–
T37	17	18	Write parameter via AL_Write
T38	18	17	–
T39	18	20	–
T40	18	20	–
T41	17	19	DataStorageIndex 3, Subindex 1: "DS_DownloadEnd" (see Table B.10) Read "Parameter Checksum" (see Table B.10).
T42	19	20	–
T43	6	8	–

INTERNAL ITEMS	TYPE	DEFINITION
DS_Cleared	Bool	Data Storage handling switched off
DS_Disabled	Bool	Data Storage handling deactivated
DS_Enabled	Bool	Data Storage handling activated
COMx_ERROR	Bool	Error in COMx communication detected
Device_Error	Bool	Access to Index denied, AL_Read or AL_Write.cnf(-) with ErrorCode 0x80

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

INTERNAL ITEMS	TYPE	DEFINITION
DS_Startup	Variable	Trigger from CM state machine, see Figure 99
NoCOMx	Bool	No COMx communication
COMx	Bool	COMx communication working properly
DS_Upload	Variable	Trigger upon DS_UPLOAD_REQ, see Table D.1 and Table B.10
DS_UPLOAD_FLAG	Bool	See Table B.10 ("State property")
UploadEnable	Bool	Data Storage handling configuration
DownloadEnable	Bool	Data Storage handling configuration
DS_Valid	Bool	Valid parameter set available within the Master. See state description "SM: CheckDSValidity_8"
DS_Invalid	Bool	No valid parameter set available within the Master. See state description "SM: CheckDSValidity_8"
Checksum_Mismatch	Bool	Acquired "Parameter Checksum" from Device does not match the checksum within Data Storage (binary comparison)
Checksum_Match	Bool	Acquired "Parameter Checksum" from Device matches the checksum within Data Storage (binary comparison)
Data Storage locked	Bool	See Table B.10 ("State property")

11.4.5 Parameter selection for Data Storage

The Device designer defines the parameters that are part of the Data Storage mechanism.

The IODD marks all parameters not included in Data Storage with the attribute "excludedFromDataStorage". However, the Data Storage mechanism shall not consider the information from the IODD but rather the Parameter List read out from the Device.

11.5 On-request Data exchange (ODE)

Figure 109 shows the state machine of the Master's On-request Data Exchange. This behavior is mandatory for a Master.

The gateway application is able to read On-request Data (OD) from the Device via the service "SMI_DeviceRead". This service is directly mapped to service AL_Read with Port, Index, and Subindex (see 8.2.2.1).

The gateway application is able to write On-request Data (OD) to the Device via the service "SMI_DeviceWrite". This service is directly mapped to service AL_Write with Port, Index, and Subindex (see 8.2.2.2).

During an active data transmission of the Data Storage mechanism, all On-request Data requests are blocked.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

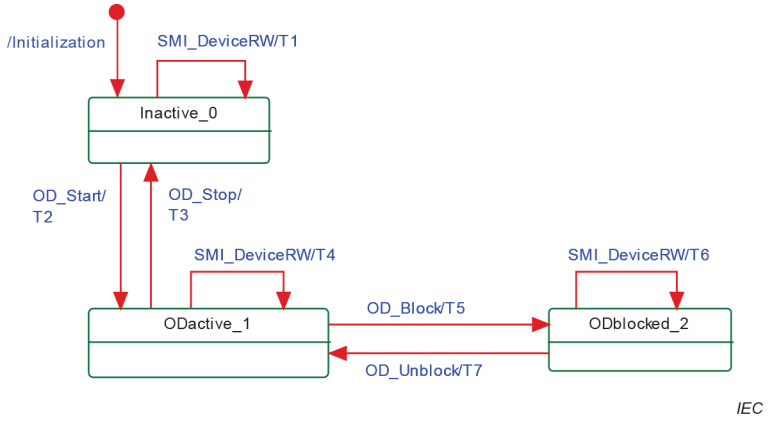


Figure 109 – State machine of the On-request Data Exchange

Table 128 shows the state transition table of the On-request Data Exchange state machine.

Table 128 – State transition table of the ODE state machine

STATE NAME			STATE DESCRIPTION
Inactive_0			Waiting for activation
ODactive_1			On-request Data communication active using AL_Read or AL_Write
ODblocked_2			On-request Data communication blocked
TRANSITION	**SOURCE STATE**	**TARGET STATE**	**ACTION**
T1	0	0	Access blocked (inactive): indicates "DEVICE_NOT_ACCESSIBLE" to the gateway application
T2	0	1	–
T3	1	0	–
T4	1	1	AL_Read or AL_Write
T5	1	2	–
T6	2	2	Access blocked temporarily: indicates "SERVICE_TEMP_UNAVAILABLE" to the gateway application
T7	2	1	–
INTERNAL ITEMS		**TYPE**	**DEFINITION**
SMI_DeviceRW		Variable	On-request Data read or write requested via SMI_DeviceRead, SMI_DeviceWrite, SMI_ParamWriteBatch, or SMI_ParamReadBatch

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.6 Diagnosis Unit (DU)

11.6.1 General

The Diagnosis Unit (DU) routes Device or Port specific Events via the SMI_DeviceEvent and the SMI_PortEvent service to the gateway application (see Figure 99). These Events primarily contain diagnosis information. The structure corresponds to the AL_Event in 8.2.2.11 with Instance, Mode, Type, Origin, and EventCode.

Additionally, the DU generates a Device or Port specific diagnosis status that can be retrieved by the SMI_PortStatus service in PortStatusList (see Table E.4 and 11.6.4).

11.6.2 Device specific Events

The SMI_DeviceEvent service provides Device specific Events directly to the gateway application. The special DS_UPLOAD_REQ Event (see 10.4 and Table D.1) of a Device shall be redirectted to the common Master application Data Storage. Those Events are acknowledged by the DU itself and not propagated via SMI_DeviceEvent to the gateway.

Device diagnosis information flooding is avoided by flow control as shown in Figure 110, which allows for only one Event per Device to be propagated via SMI_DeviceEvent to the gateway application at a time.

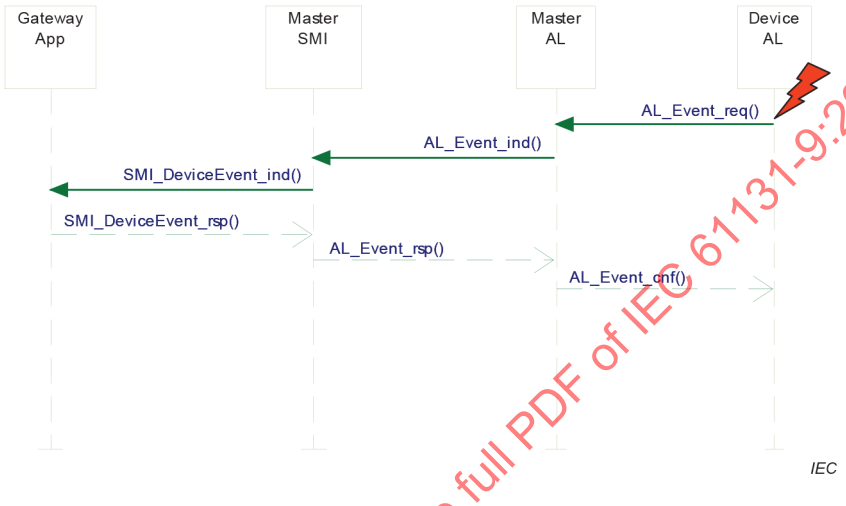


Figure 110 – DeviceEvent flow control

11.6.3 Port specific Events

The SMI_PortEvent service provides also Port specific Events directly to the gateway application. Those Events are similarly characterized by Instance = Application, Source = Master, Type = Error or Warning or Notification, and Mode Event appears or disappears or single shot (see A.6.4). Usually, only one Port Event at a time is pending as shown in Figure 111.

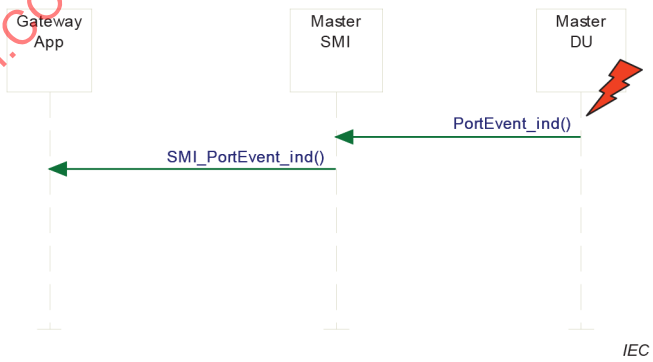


Figure 111 – Port Event flow control

The following rules apply:

– It is not required to send disappearing Port Events in case of Device communication interrupt (communication restart).

– Once communication is resumed, the gateway client is responsible for proper reporting of the current Event causes.

Port specific Events are specified in Clause D.3.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.6.4 Dynamic diagnosis status

The DU generates the diagnosis status by collecting all appearing DeviceEvents and PortEvents continuously in a buffer. Any disappearing Event will cause the DU to remove the corresponding Event with the same EventCode from the buffer. Thus, the buffer represents an actual image of the consolidated diagnosis status, which can be taken over as diagnosis entries within the PortStatusList (see Table E.4).

After COMLOST and during Device startup the buffer will be deleted.

11.6.5 Best practice recommendations

The main goal for diagnosis information is to alert an operator in an efficient manner i.e.:

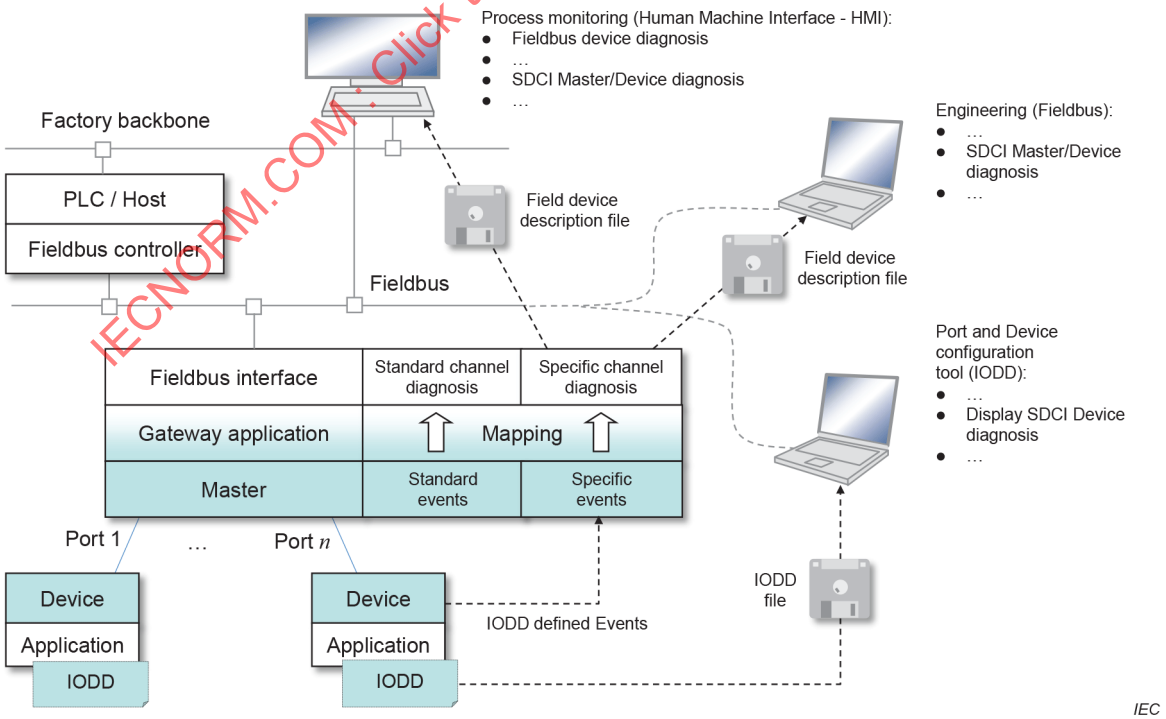
– no diagnosis information flooding;
– report of the root cause of an incident within a Device or within the Master and no subsequent correlated faults;
– diagnosis information shall provide information on how to maintain or repair the affected component for fast recovery of the automation system.

Figure 112 shows an example of the diagnosis information flow through a complete SDCI/fieldbus system.

NOTE 1 The flow can end at the Master/PDCT or be more integrated depending on the fieldbus capabilities.

Within SDCI, diagnosis information on Devices is conveyed to the Master via Events consisting of EventQualifiers and EventCodes (see Clause A.6). The associated human readable text is available for standardized EventCodes within this document (see Annex D) and for vendor specific EventCodes within the associated IODD file of a Device.

NOTE 2 The standardized EventCodes can be mapped to semantically identical or closest fieldbus channel diagnosis definitions within the gateway application.



NOTE Blue shaded areas indicate features specified in this document.

Figure 112 – SDCI diagnosis information propagation via Events

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.7 PD Exchange (PDE)

11.7.1 General

The Process Data Exchange provides the transmission of Process Data between the gateway application and the connected Device.

The Standard Master Interface (SMI) comes with the following three services for the gateway application:

– SMI_PDIn allows for reading input Process Data from the InBuffer together with Quality Information (PQI), see 11.2.17;
– SMI_PDOut allows for writing output Process Data to the OutBuffer, see 11.2.18;
– SMI_PDInOut allows for reading output Process Data from the OutBuffer and reading input Process Data from the InBuffer within one cycle, see 11.2.19.

After an established communication and Data Storage, the Port is ready for any On-request Data (ODE) transfers. Process Data exchange is enabled whenever the specific Port or all Ports are switched to the OPERATE mode.

11.7.2 Process Data input mapping

11.7.2.1 Port Modes "IOL_MANUAL" or "IOL_AUTOSTART"

Figure 99 shows how the Master application "Process Data Exchange" (PDE) is related to the other Master applications. It is responsible for the cyclic acquisition of input data using the service "AL_GetInput" (see 8.2.2.4) and of Port Qualifier (PQ) information using the service "AL_Control" (see 8.2.2.12). Both shall be synchronized for consistency.

A gateway application can get access to these data via the service "SMI_PDIn" (see 11.2.17). Figure 113 illustrates the principles of Process Data Input mapping and the content of the ArgBlock of this service (see Clause E.10) consisting of the ArgBlockID, the qualifier PQI, the parameter InputDataLength, and the input Process Data.

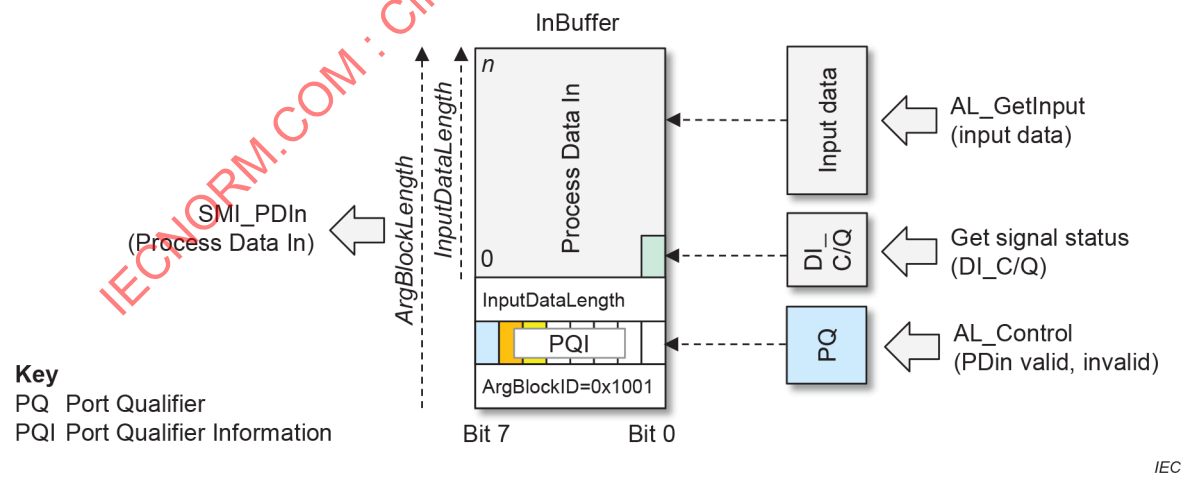


Figure 113 – Principles of Process Data Input mapping

At state OPERATE the input data are cyclicly copied into the InBuffer starting at offset "4".

The InBuffer is expanded by an octet "PQI" at offset "2", whose content shall be updated anytime the input data are read. Figure 114 illustrates the structure of this octet.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

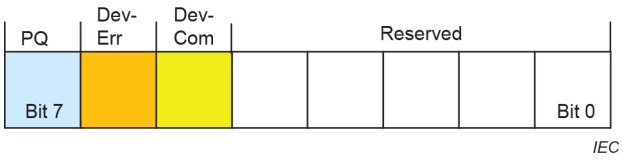


Figure 114 – Port Qualifier Information (PQI)

Bit 0 to 4: Reserved

These bits are reserved for future use.

Bit 5: DevCom

Parameter "PortStatusInfo" of service "SMI_PortStatus" provides the necessary information for this bit.

It will be set if a Device is detected and in PREOPERATE or OPERATE state. It will be reset if there is no Device available.

Bit 6: DevErr

Parameter "PortStatusInfo" and "DiagEntry x"of service "SMI_PortStatus" provide the necessary information for this bit.

It will be set if an Error or Warning occurred assigned to either Device or Port. It will be reset if there is no Error or Warning.

Bit 7: Port Qualifier (PQ)

A value VALID for Process Data in service "AL_CONTROL" will set this bit.

A value INVALID or PortStatusInfo <> "4" (see Clause E.4) will reset this bit.

11.7.2.2 Port Mode "DI_C/Q"

In this Port Mode the signal status of DI_C/Q will be mapped into octet 0, Bit 0 of the InBuffer (see Figure 113).

11.7.2.3 Port Mode "DEACTIVATED"

In this Port Mode the InBuffer will be filled with "0".

11.7.3 Process Data output mapping

11.7.3.1 Port Modes "IOL_MANUAL" or "IOL_AUTOSTART"

Master application "Process Data Exchange" (PDE) is responsible for the cyclic transfer of output data using the services "AL_SetOutput" (see 8.2.2.10) and "AL_Control" (see 8.2.2.12). Both shall be synchronized for consistency.

A gateway application can write data via the service "SMI_PDOut" into the OutBuffer (see 11.2.18). Figure 115 illustrates the principles of Process Data Output mapping and the content of the ArgBlock of this service (see Clause E.11) consisting of the ArgBlockID, the Output Enable bit, the parameter OutputDataLength, and the output Process Data.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

An ErrorType 0x4034 – *Incorrect ArgBlock length* will be returned if the length does not add up to Process Data Out plus four octets (see C.4.9).

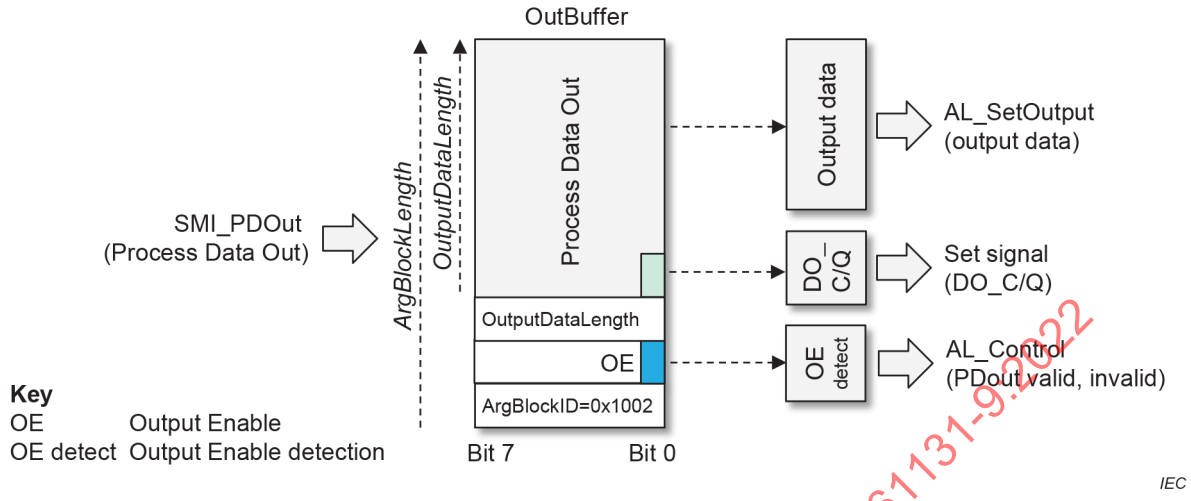


Figure 115 – Principles of Process Data Output mapping

At state OPERATE the Process Data Out are cyclicly copied to output data starting at offset "3".

The OutBuffer is expanded by an octet "OE" (Output Enable) at offset "2". Bit 0 indicates the validity of the Process Data Out. "0" means invalid, "1" means valid data. A change of this Bit from "0" to "1" will launch an AL_Control with "PDout valid". A change of this Bit from "1" to "0" will launch an AL_Control with "PDout invalid". See "OE detect" in Figure 115.

A substitute value will be activated when in Port mode "DO_C/Q".

11.7.3.2 Port Mode "DO_C/Q"

In this Port Mode octet 0, Bit 0 of the Process Data Out in the OutBuffer will be mapped into the signal status of DO_C/Q (see Figure 115).

11.7.4 Process Data invalid/valid qualifier status

A sample transmission of an output PD qualifier status "invalid" from Master AL to Device AL is shown in the upper section of Figure 116.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

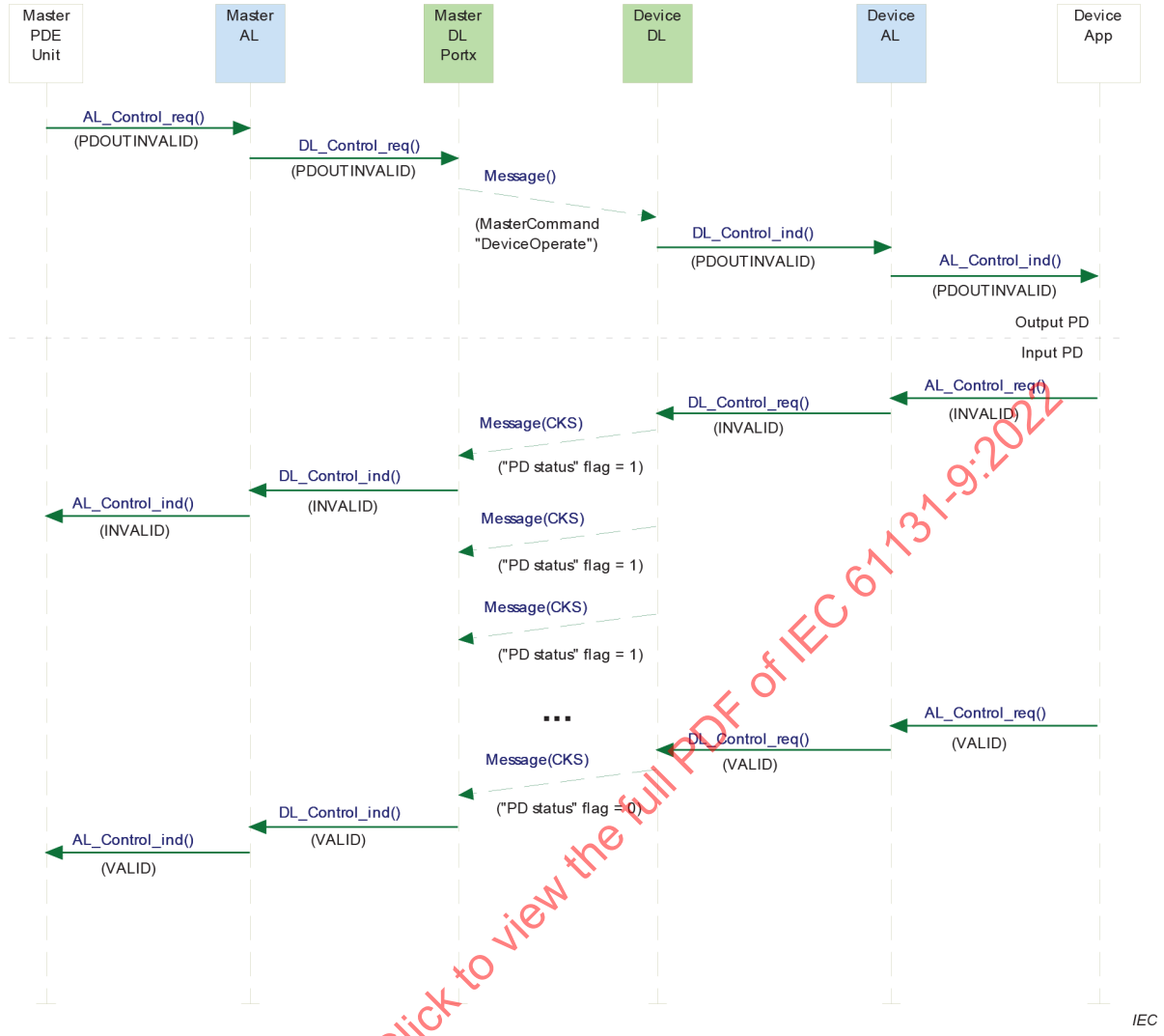


Figure 116 – Propagation of PD qualifier status between Master and Device

The Master informs the Device about the output Process Data qualifier status "valid/invalid" by sending MasterCommands (see Table B.2) to the Direct Parameter page 1 (see 7.3.7.1).

For input Process Data the Device sends the Process Data qualifier status in every single message as "PD status" flag in the Checksum/Status (CKS) octet (see A.1.5) of the Device message. A sample transmission of the input PD qualifier status "valid" from Device AL to Master AL is shown in the lower section of Figure 116.

Any perturbation while in interleave transmission mode leads to an input or output Process Data qualifier status "invalid" indication respectively.

12 Holistic view on Data Storage

12.1 User point of view

In Clause 12, the Data Storage mechanism is described from a holistic user's point of view as best practice pattern. This is in contrast to 10.4 and 11.4 where Device and Master are described separately and each with more features then used within the recommended concept herein after.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

12.2 Operations and preconditions

12.2.1 Purpose and objectives

The main purpose of the SDCI Data Storage mechanism is the replacement of obviously defect Devices or Masters by spare parts (new or used) without using configuration, parameterization, or other tools. The scenarios and associated preconditions are described in 12.2.2 to 12.2.4.

12.2.2 Preconditions for the activation of the Data Storage mechanism

The following preconditions shall be observed prior to the usage of Data Storage:

a) Data Storage is only available for Devices and Masters implemented according to this document (≥ *V1.1*).

b) The Inspection Level of that Master Port, the Device is connected to shall be adjusted to "type compatible" (corresponds to "TYPE_COMP" within Table 80).

c) The Backup Level of that Master Port to which the Device is connected shall be either "Backup/Restore" or "Restore", which corresponds to DS_Enabled in 11.4.4. See 12.4 within this document for details on Backup Level.

12.2.3 Preconditions for the types of Devices to be replaced

After activation of a Backup Level (Data Storage mechanism) a "faulty" Device can be replaced by a type equivalent or compatible other Device. In some exceptional cases, for example non-calibrated Devices, a user manipulation can be required such as teach-in, to guarantee the same functionality and performance.

Thus, two classes of Devices exist in respect to exchangeability, which shall be described in the user manual of the particular Device:

Data Storage class 1: automatic DS

The configured Device supports Data Storage in such a manner that the replacement Device plays the role of its predecessor fully automatically and with the same performance.

Data Storage class 2: semi-automatic DS

The configured Device supports Data Storage in such a manner that the replacement Device requires user manipulation such as teach-in prior to operation with the same performance.

The Data Storage class shall be described in the user manual of the Device. The Device designer is responsible in case of class 2 for preventing a dangerous system restart after the Device replacement, at least via descriptions within the user manual.

12.2.4 Preconditions for the parameter sets

Each Device operates with the configured set of active parameters. The associated set of backup parameters stored within the system (Master and upper-level system, for example PLC) can be different from the set of active parameters (see Figure 117).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

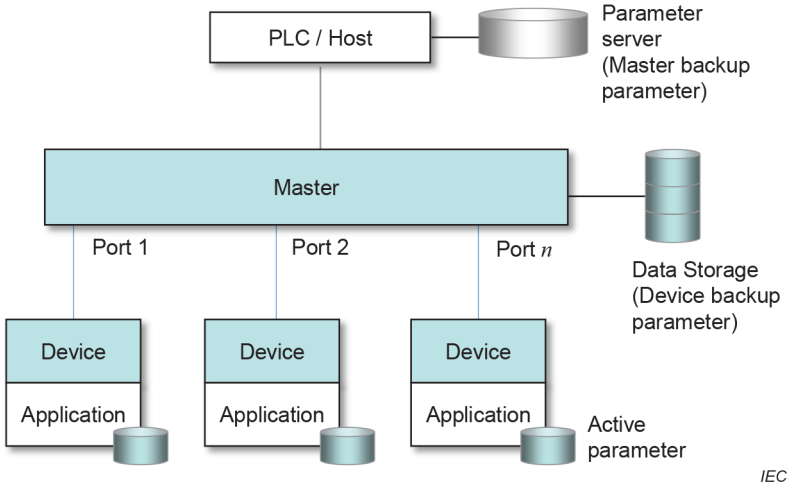


Figure 117 – Active and backup parameter

A replacement of the Device in operation will result in overwriting the active parameter set with the backup parameters in the newly connected Device.

12.3 Commissioning

12.3.1 On-line commissioning

Usually, the Devices are configured and parameterized along with the configuration and parameterization of the fieldbus and PLC system with the help of engineering tools. After the user has assigned values to the parameters, they are downloaded into the Device and become active parameters. Upon the system command "ParamDownloadStore", these parameters are uploaded (copied) into the Data Storage within the Master, which in turn will initiate a backup of all its parameters depending on the features of the upper-level system.

12.3.2 Off-site commissioning

Another possibility is the configuration and parameterization of Devices with the help of extra tools such as "USB-Masters" and the IODD of the Device away (off-site) from the machine or facility (see Figure 118).

The USB-Master tool will mark the parameter set after configuration, parameterization, and validation (to become "active") via DS_UPLOAD_FLAG (see Table 130 and Table B.10). After installation into the machine or facility these parameters are uploaded (copied) automatically into the Data Storage within the Master (backup).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

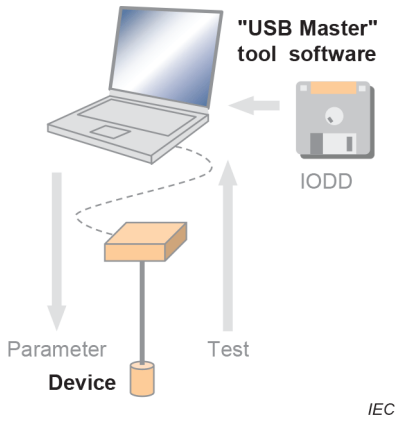


Figure 118 – Off-site commissioning

12.4 Backup Levels

12.4.1 Purpose

Within automation projects including SDCI usually three situations with different user requirements for backup of parameters via Data Storage can be identified:

– Commissioning ("Disable");
– Production ("Backup/Restore");
– Production ("Restore").

Accordingly, three different "Backup Levels" are defined allowing the user to adjust the system to the particular functionality such as for Device replacement, off-site commissioning, parameter changes at runtime (see Table 129).

These adjustment possibilities lead for example to drop-down menu entries for "Backup Level".

12.4.2 Overview

Table 129 shows the recommended practice for Data Storage within an SDCI system. It simplifies the activities and their comprehension since activation of the Data Storage implies transfer of the parameters.

Table 129 – Recommended Data Storage Backup Levels

Backup Level	Data Storage adjustments	Behavior
Commissioning ("Disable")	Master Port: Activation state: "DS_Cleared"	Any change of active parameters within the Device will not be copied and saved. Device replacement without automatic or semi-automatic Data Storage.
Production ("Backup/Restore")	Master Port: Activation state: "DS_Enabled" Master Port: UploadEnable Master Port: DownloadEnable	Changes of active parameters within the Device will be copied and saved. Device replacement with automatic or semi-automatic Data Storage supported.
Production ("Restore")	Master Port: Activation state: "DS_Enabled" Master Port: UploadDisable Master Port: DownloadEnable	Any change of active parameters within the Device will not be copied and saved. If the parameter set is marked to be saved, the "frozen" parameters will be restored by the Master. However, Device replacement with automatic or semi-automatic Data Storage of "frozen" parameters is supported.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Legacy rules and presetting:

– For (legacy) Devices according to [8] or Devices according to this document where the Port is preset to Inspection Level "NO_CHECK", only the Backup Level "Commissioning" shall be supported. This should also be the default presetting in this case.
– For Devices according to this document where the Port is preset to Inspection Level "TYPE_COMP" all three Backup Levels shall be supported. Default presetting in this case should be "Backup/Restore".

The following subclauses 12.4.3 to 12.4.5 describe the phases in detail.

12.4.3 Commissioning ("Disable")

Data Storage is disabled in the Master Port configuration, where configurations, parameterizations, and PLC programs are fine-tuned, tested, and verified. This includes the involved Masters and Devices. Usually, repeated saving (uploading) of the active Device parameters makes no sense in this phase. As a consequence, the replacement of Master and Devices with automatic/semi-automatic Data Storage is not supported.

12.4.4 Production ("Backup/Restore")

Data Storage in Master Port configuration will be enabled. Current active parameters within the Device will be copied and saved as backup parameters. Device replacement with automatic or semi-automatic Data Storage is now supported via download or copy of the backup parameters to the Device and thus turning them into active parameters.

Criteria for the particular copy activities are listed in Table 130. These criteria are the conditions to trigger a copy process of the active parameters to the backup parameters, thus ensuring the consistency of these two sets.

Table 130 – Criteria for backing up parameters ("Backup/Restore")

User action	Operations	Data Storage
Commissioning session (see 12.3.1)	Parameterization of the Device via Master tool (on-line). Transfer of active parameter(s) to the Device will cause backup activity.	Master tool sends ParamDownloadStore; Device sets "DS_UPLOAD_FLAG" and then triggers upload via "DS_UPLOAD_REQ" Event. "DS_UPLOAD_FLAG" is reset as soon as the upload is completed.
Switching from commissioning to production	Restart of Port and Device because Port configuration has been changed	During system startup, the "DS_UPLOAD_FLAG" triggers upload (copy). "DS_UPLOAD_FLAG" is reset as soon as the upload is completed.
Local modifications	Changes of the active parameters through teach-in or local parameterization at the Device (on-line)	Device technology application sets "DS_UPLOAD_FLAG" and then triggers upload via "DS_UPLOAD_REQ" Event. "DS_UPLOAD_FLAG" is reset as soon as the upload is completed.
Off-site commissioning (see 12.3.2)	Phase 1: Device is parameterized off-site via USB-Master tool (see Figure 118). Phase 2: Connection of that Device to a Master Port.	Phase 1: USB-Master tool sends ParamDownloadStore; Device sets "DS_UPLOAD_FLAG" (in non-volatile memory) and then triggers upload via "DS_UPLOAD_REQ" Event, which is ignored by the USB-Master. Phase 2: During system startup, the "DS_UPLOAD_FLAG" triggers upload (copy). "DS_UPLOAD_FLAG" is reset as soon as the upload is completed.
Changed Port configuration (in case of "Backup/Restore" or "Restore")	Whenever Port configuration has been changed via Master tool (on-line): e.g. Configured VendorID (CVID), Configured DeviceID (CDID), see 11.4.4.	Change of Port configuration to different VendorID and/or DeviceID as stored within the Master triggers "DS_Delete" followed by an upload (copy) to Data Storage (see 13.4.1, 11.3.1 and 11.4.4).
PLC program demand	Parameter change via user program followed by a SystemCommand	User program sends SystemCommand ParamDownloadStore; Device sets "DS_UPLOAD_FLAG" and then triggers upload via "DS_UPLOAD_REQ" Event. "DS_UPLOAD_FLAG" is reset as soon as the upload is completed.
Device reset (see 10.7)	Parameter change using one of the reset options in 10.7	See Table 101.
NOTE For details on "DS_UPLOAD_FLAG" see 11.4.4.		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

12.4.5 Production ("Restore")

Data Storage in Master Port configuration is enabled. However, only DS_Download operation is available. This means that unintended overwriting of Data Storage within the Master is prohibited.

Any changes of the active parameters through teach-in, tool based parameterization, or local parameterization will lead to a Data Storage Event, and State Property "DS_UPLOAD_FLAG" will be set in the Device.

In back-up level Production ("Restore") the Master shall ignore this flag and shall issue a DS_Download to overwrite the changed parameters.

Criteria for the particular copy activities are listed in Table 131. These criteria are the conditions to trigger a copy process of the active parameters to the backup parameters, thus ensuring the consistency of these two sets.

Table 131 – Criteria for backing up parameters ("Restore")

User action	Operations	Data Storage
Change Port configura-tion	Change of Port configuration via Master tool (on-line): e.g. Configured VendorID (CVID), Configured DeviceID (CDID), see 11.4.4.	Change of Port configuration triggers "DS_Delete" followed by an upload (copy) to Data Storage (see 13.4.1, 11.3.1, and 11.4.4).

12.5 Use cases

12.5.1 Device replacement ("Backup/Restore")

The stored (saved) set of back-up parameters overwrites the active parameters (e.g. factory settings) within the replaced compatible Device of same type. This one operates after a restart with the identical parameters as with its predecessor.

The preconditions for this use case are

a) Devices and Master Port adjustments according to 12.2.2;

b) Backup Level: "Backup/Restore";

c) The replacement Device shall be re-initiated to "factory settings" in case it is not a new Device out of the box (for "Back-to-box" see 10.7.5).

12.5.2 Device replacement ("Restore")

The stored (saved) set of back-up parameters overwrites the active parameters (e.g. factory settings) within the replaced compatible Device of same type. This one operates after a restart with the identical parameters as with its predecessor.

The preconditions for this use case are

a) Devices and Master Port adjustments according to 12.2.2;

b) Backup Level: "Restore".

12.5.3 Master replacement

12.5.3.1 General

This feature depends heavily on the implementation and integration concept of the Master designer and manufacturer as well as on the features of the upper-level system (fieldbus).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

12.5.3.2 Without fieldbus support (base level)

Principal approach for a replaced (new) Master using a Master tool:

a) set Port configurations: amongst others the Backup Level to "Backup/Restore" or "Restore";
b) Master "reset to factory settings": clear backup parameters of all Ports within the Data Storage in case it is not a new Master out of the box;
c) Active parameters of all Devices are automatically uploaded (copied) to Data Storage (backup).

12.5.3.3 Fieldbus support (comfort level)

Any kind of fieldbus specific mechanism to back up the Master parameter set including the Data Storage of all Devices is used. Even though these fieldbus mechanisms are similar to the SDCI approach, they follow their certain paradigm which could conflict with the described paradigm of the SDCI backup mechanism (see Figure 117).

12.5.3.4 PLC system

The Device and Master parameters are stored within the system specific database of the PLC and downloaded to the Master at system startup after replacement.

This top-down concept could conflict with the active parameter setting within the Devices.

12.5.4 Project replication

Following the concept of 12.5.3.3, the storage of complete Master parameter sets within the parameter server of an upper-level system can automatically initiate the configuration of Masters and Devices besides any other upper-level components and thus support the automatic replication of machines.

Following the concept of 12.5.3.4, after supply of the Master by the PLC, the Master can supply the Devices.

13 Integration

13.1 Generic Master model for system integration

Figure 119 shows the integration relevant excerpt of Figure 95. The basis is the Standardized Master Interface (SMI), which is specified in an abstract manner in 11.2. It transforms SDCI objects into services and objects appropriate for the upper-level systems such as embedded controllers, IT systems (JSON), fieldbuses and PLCs, engineering systems, as well as universal Master Tools (PDCT) for Masters of different brands.

It is an objective of this SMI to achieve uniform behavior of Masters of different brands from a user's point of view. Another objective is to provide a stringent specification for organizations developing integration specifications into their systems without administrative overhead.

In Figure 119, the green marked items are areas of responsibility of fieldbus organizations and their integration specifications. The blue marked items are areas of responsibility of IT organizations and their specifications. The red marked items are areas of responsibility of individual automation equipment manufacturers. The white marked item ("Gateway management") represents a coordination layer for the different gateway applications. A corresponding specification is elaborated by a joint working group [12].

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

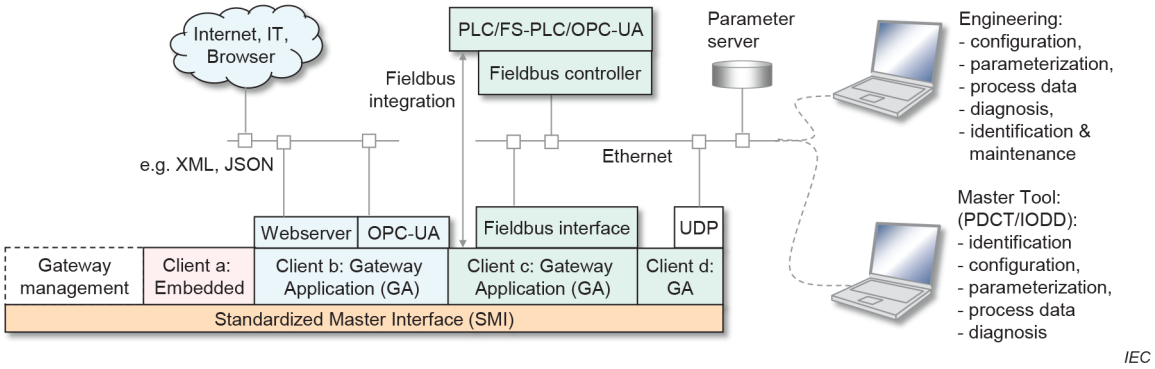


Figure 119 – Generic Master model for system integration

13.2 Role of gateway applications

13.2.1 Clients

It is the role of gateway applications to provide translations of SMI services into the target systems (clients). Table 105 provides an overview of specified mandatory and optional SMI services. The designer of a gateway application determines the SMI service call technology.

Gateway applications such as shown in Figure 119 include but are not limited to:

– pure coding tasks of the abstract SMI services, for example for embedded controllers,
– comfortable web server providing text and data for standard browsers using for example XML, JSON,
– OPC-UA server used for parameterization and data exchange via IT applications (security solutions available),
– adapters with a fieldbus interface for programmable logic controllers (PLCs) and human machine interfaces based on OPC-UA,
– adapters for a User Datagram Protocol (UDP) to connect engineering tools.

13.2.2 Coordination

It is the responsibility of gateway applications to prevent access conflicts such as

– different clients to one Device,
– concurrent tasks for one Device, for example prevent SystemCommand "Restore factory settings" while Block Parameterization is running.

13.3 Security

The aspect of security is important whenever access to Master and Device data is involved. In case of fieldbuses most of the fieldbus organizations provide dedicated guidelines on security. In general, the IEC 62443 series [13] is an appropriate source of protection strategies for industrial automation applications.

13.4 Special gateway applications

13.4.1 Changing Device configuration including Data Storage

After each change of Device configuration and/or parameterization (CVID and/or CDID, see 9.2.2.2), the associated previously stored data set within the Master shall be cleared or marked invalid via the variable DS_Delete.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

13.4.2 Parameter server and recipe control

The Master may combine the entire parameter sets of the connected Devices together with all other relevant data for its own operation and make this data available for upper-level applications. For example, this data may be saved within a parameter server which may be accessed by a PLC program to change recipe parameters, thus supporting flexible manufacturing.

NOTE The structure of the data exchanged between the Master and the parameter server is outside the scope of this document.

13.5 Port and Device Configuration Tool (PDCT)

13.5.1 Strategy

Figure 119 demonstrates the necessity of a tool to configure Ports, parameterize the Device, display diagnosis information, and provide identification and maintenance information. Depending on the degree of integration into a fieldbus system, the PDCT functions can be reduced, for example if the Port configuration can be achieved via the field device description file of the particular fieldbus (engineering).

13.5.2 Accessing Masters via SMI

Figure 120 illustrates sample sequences of a standardized PDCT access to Masters (SMI). The Standardized Master Interface is specified in 11.2.

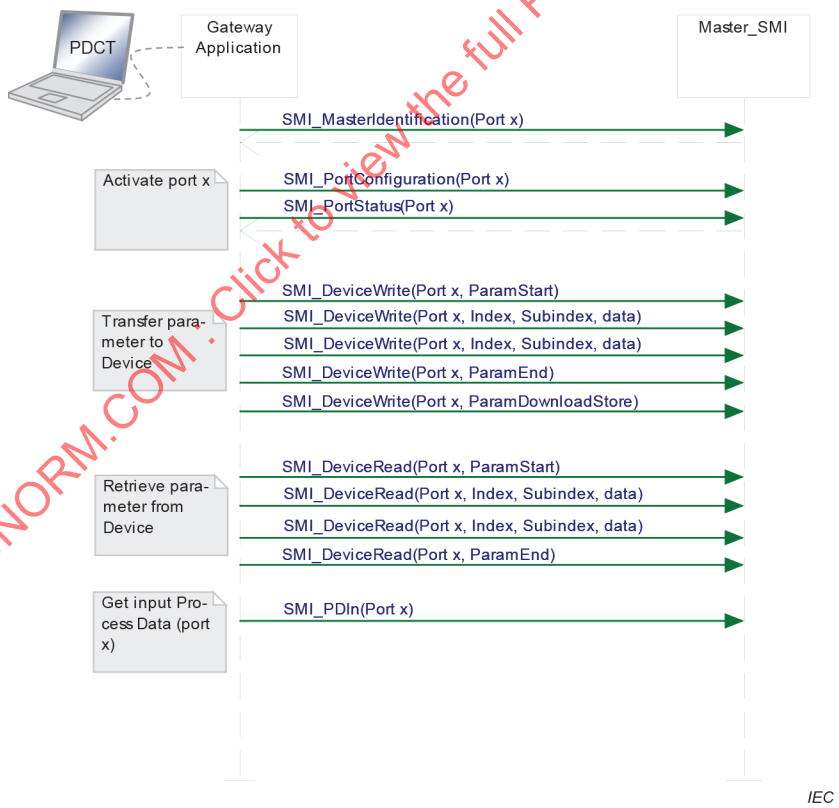


Figure 120 – PDCT via gateway application

13.5.3 Basic layout examples

Figure 121 shows one example of a PDCT display layout.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

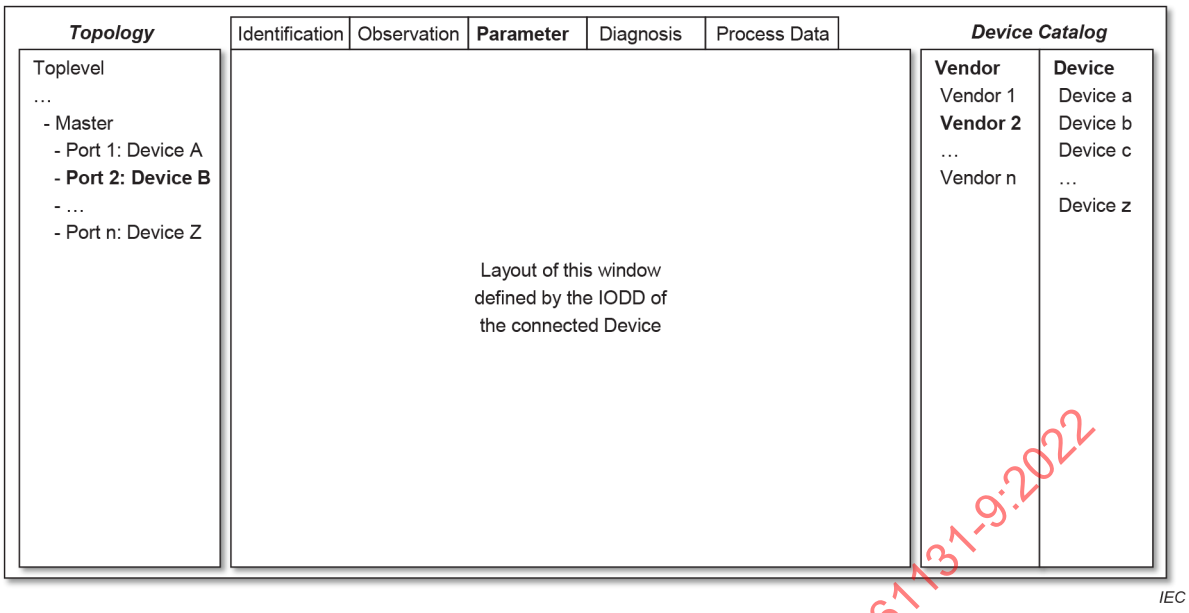


Figure 121 – Example 1 of a PDCT display layout

The PDCT display should always provide a navigation window for a project or a network topology, a window for the particular view on a chosen Device that is defined by its IODD, and a window for the available Devices based on the installed IODD files.

Figure 122 shows another example of a PDCT display layout.

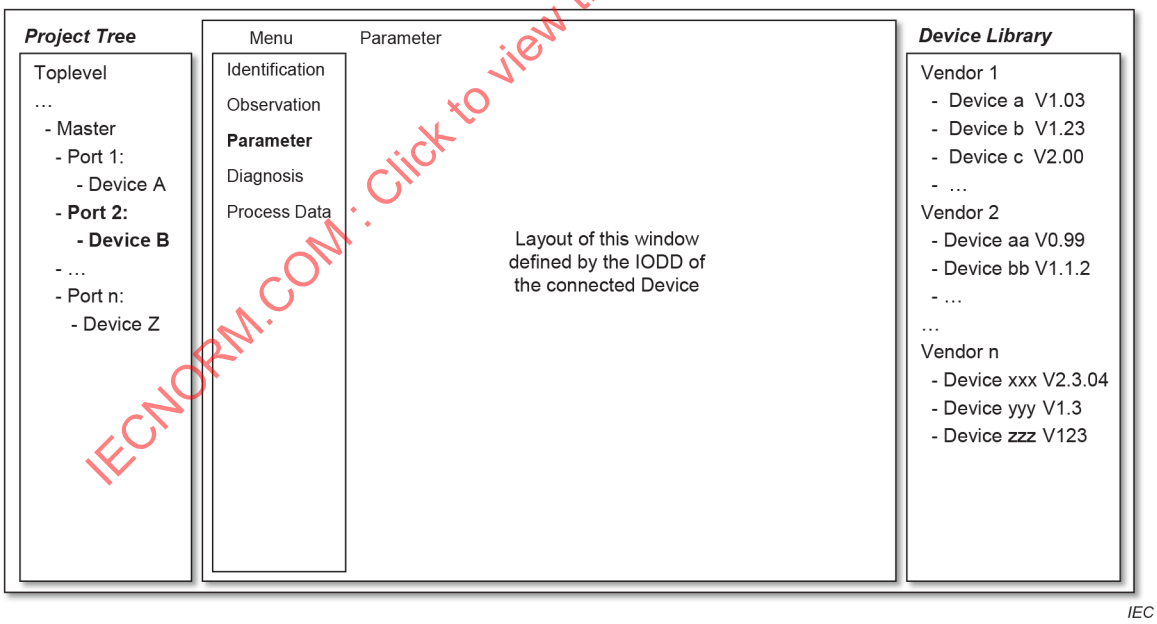


Figure 122 – Example 2 of a PDCT display layout

NOTE Further information can be retrieved from IEC TR 62453-61 [4].

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex A
(normative)

Codings, timing constraints, and errors

A.1 General structure and encoding of M-sequences

A.1.1 Overview

The general concept of M-sequences is outlined in 7.3.3.2. Subclauses A.1.2 to A.1.6 provide a detailed description of the individual elements of M-sequences.

A.1.2 M-sequence control (MC)

The Master indicates the manner the user data (see A.1.4) shall be transmitted in an M-sequence control octet. This indication includes the transmission direction (read or write), the communication channel, and the address (offset) of the data on the communication channel. The structure of the M-sequence control octet is shown in Figure A.1.

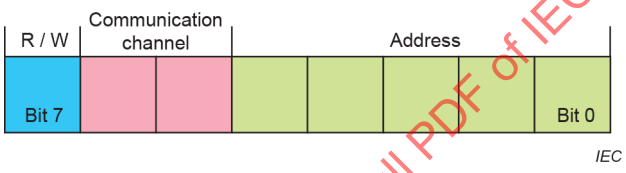


Figure A.1 – M-sequence control

Bits 0 to 4: Address

These bits indicate the address, i.e. the octet offset of the user data on the specified communication channel (see also Table A.1). In case of an ISDU channel, these bits are used for flow control of the ISDU data. The address, which means in this case the position of the user data within the ISDU, is only available indirectly (see 7.3.6.2).

Bits 5 to 6: Communication channel

These bits indicate the communication channel for the access to the user data. The defined values for the communication channel parameter are listed in Table A.1.

Table A.1 – Values of communication channel

Value	Definition
0	Process
1	Page
2	Diagnosis
3	ISDU

Bit 7: R/W

This bit indicates the transmission direction of the user data on the selected communication channel, i.e. read access (transmission of user data from Device to Master) or write access (transmission of user data from Master to Device). The defined values for the R/W parameter are listed in Table A.2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table A.2 – Values of R/W

Value	Definition
0	Write access
1	Read access

A Device is not required to support each and every of the 256 values of the M-sequence control octet. For read access to not implemented addresses or communication channels the value "0" shall be returned. A write access to not implemented addresses or communication channels shall be ignored.

A.1.3 Checksum/M-sequence type (CKT)

The M-sequence type is transmitted together with the checksum in the check/type octet. The structure of this octet is demonstrated in Figure A.2.

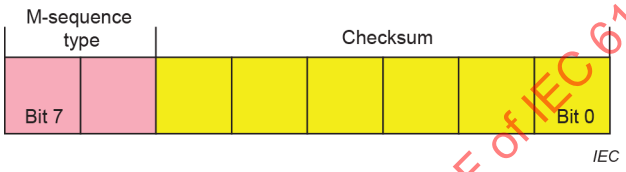


Figure A.2 – Checksum/M-sequence type octet

Bits 0 to 5: Checksum

These bits contain a 6 bit message checksum to ensure data integrity, see also A.1.6 and Clause I.1.

Bits 6 to 7: M-sequence type

These bits indicate the M-sequence type. Herewith, the Master specifies how the messages within the M-sequence are structured. Defined values for the M-sequence type parameter are listed in Table A.3.

Table A.3 – Values of M-sequence types

Value	Definition
0	Type 0
1	Type 1
2	Type 2 (see NOTE)
3	reserved
NOTE Subtypes depend on PD configuration and PD direction.	

A.1.4 User data (PD or OD)

User data is a general term for both Process Data and On-request Data. The length of user data can vary from 0 to 64 octets depending on M-sequence type and transmission direction (read/write). An overview of the available data types is shown in Table A.4. These data types can be arranged as records (different types) or arrays (same types).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table A.4 – Data types for user data

Data type	Reference
BooleanT	See Clause F.2
UIntegerT	See F.2.3
IntegerT	See F.2.4
StringT	See F.2.6
OctetStringT	See F.2.7
Float32T	See F.2.5
TimeT	See F.2.8
TimeSpanT	See F.2.9

The detailed coding of the data types can be found in Annex F.

A.1.5 Checksum/status (CKS)

The checksum/status octet is part of the reply message from the Device to the Master. Its structure is shown in Figure A.3. It comprises a 6-bit checksum, a flag to indicate valid or invalid Process Data, and an Event flag.

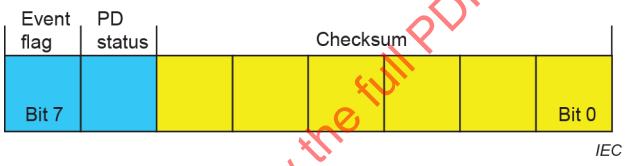


Figure A.3 – Checksum/status octet

Bits 0 to 5: Checksum

These bits contain a 6-bit checksum to ensure data integrity of the reply message. See also A.1.6 and Clause I.1.

Bit 6: PD status

This bit indicates whether the Device can provide valid Process Data or not. Defined values for the parameter are listed in Table A.5.

This PD status flag shall be used for Devices with input Process Data. Devices with output Process Data shall always indicate "Process Data valid".

If the PD status flag is set to "Process Data invalid" within a message, all the input Process Data of the complete Process Data cycle are invalid.

Table A.5 – Values of PD status

Value	Definition
0	Process Data valid
1	Process Data invalid

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Bit 7: Event flag

This bit indicates a Device initiative for the data category "Event" to be retrieved by the Master via the diagnosis communication channel (see Table A.1). The Device can report diagnosis information such as errors, warnings, or notifications via Event response messages. Permissible values for the parameter are listed in Table A.6.

Table A.6 – Values of the Event flag

Value	Definition
0	No Event
1	Event

A.1.6 Calculation of the checksum

The message checksum provides data integrity protection for data transmission from Master to Device and from Device to Master. Each UART data octet is protected by the UART parity bit (see Figure 21). Besides this individual data octet protection, all of the UART data octets in a message are XOR (exclusive or) processed octet by octet. The check/type octet is included with checksum bits set to "0". The resulting checksum octet is compressed from 8 to 6 bits in accordance with the conversion procedure in Figure A.4 and its associated formulas (see the set of equations in Formula (A.1)). The 6 bit compressed "Checksum6" is entered into the checksum/M-sequence type octet (see Figure A.2). The same procedure takes place to secure the message from the Device to the Master. In this case the compressed checksum is entered into the checksum/status octet (see Figure A.3).

A seed value of 0x52 is used for the checksum calculation across the message. It is XORed with the first octet of the message (MC).

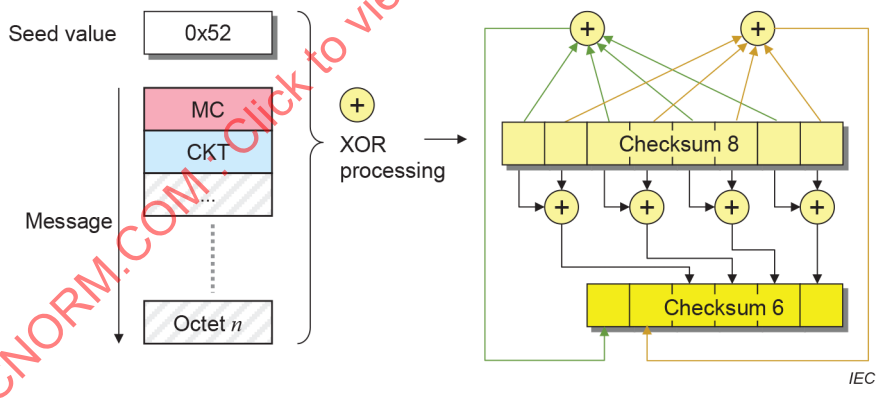


Figure A.4 – Principle of the checksum calculation and compression

The set of equations in Formula (A.1) define the compression procedure from 8 to 6 bits in detail.

$$
\begin{aligned}
D5_6 &= D7_8 \text{ xor } D5_8 \text{ xor } D3_8 \text{ xor } D1_8 \\
D4_6 &= D6_8 \text{ xor } D4_8 \text{ xor } D2_8 \text{ xor } D0_8 \\
D3_6 &= D7_8 \text{ xor } D6_8 \\
D2_6 &= D5_8 \text{ xor } D4_8 \\
D1_6 &= D3_8 \text{ xor } D2_8 \\
D0_6 &= D1_8 \text{ xor } D0_8
\end{aligned}
\qquad \text{(A.1)}
$$

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.2 M-sequence types

A.2.1 Overview

Process Data and On-request Data use separate cyclic and acyclic communication channels (see Figure 8) to ensure scheduled and deterministic delivery of Process Data while delivery of On-request Data does not have consequences on the Process Data transmission performance.

Within SDCI, M-sequences provide the access to the communication channels via the M-sequence Control octet. The number of different M-sequence types meets the various requirements of sensors and actuators regarding their Process Data width. See Figure 39 for an overview of the available M-sequence types that are specified in A.2.2 to A.2.5. See A.2.6 for rules on how to use the M-sequence types.

A.2.2 M-sequence TYPE_0

M-sequence TYPE_0 is mandatory for all Devices. It only transmits On-request Data. One octet of user data is read or written per cycle. This M-sequence is shown in Figure A.5.

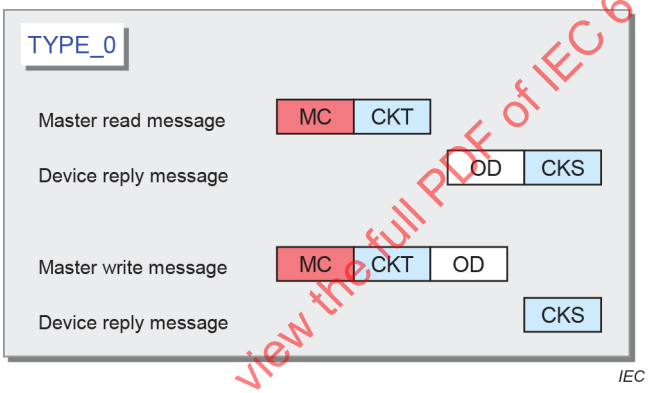


Figure A.5 – M-sequence TYPE_0

A.2.3 M-sequence TYPE_1_x

M-sequence TYPE_1_x is optional for all Devices.

M-sequence TYPE_1_1 is shown in Figure A.6.

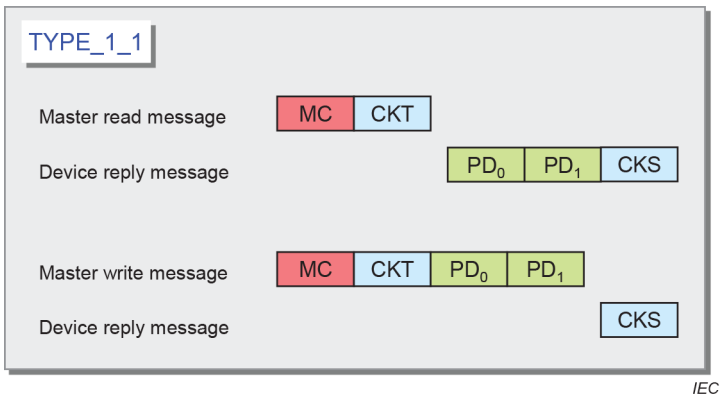


Figure A.6 – M-sequence TYPE_1_1

Two octets of Process Data are read or written per cycle. Address (bit offset) belongs to the process communication channel (see A.2.1).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

In case of interleave mode (see 7.3.4.2) and odd-numbered PD length the remaining octets within the messages are padded with 0x00.

M-sequence TYPE_1_2 is shown in Figure A.7. Two octets of On-request Data are read or written per cycle.

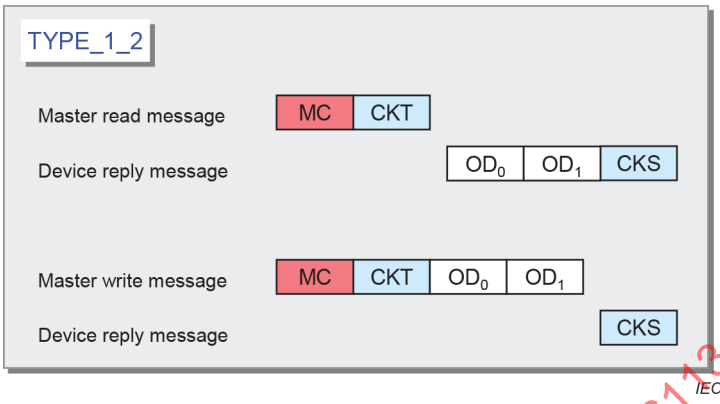


Figure A.7 – M-sequence TYPE_1_2

M-sequence TYPE_1_V providing variable (extendable) message length is shown in Figure A.8. A number of m octets of On-request Data are read or written per cycle.

When accessing octets via page and diagnosis communication channels using an M-sequence TYPE with multi-octet ODs, the following rules apply:

– At write access, only the first octet (OD_0) of On-request Data is relevant. The Master shall send all subsequent ODs filled with "0x00". Any Device shall evaluate only the first octet of ODs and ignore the remaining octets.
– At read access, the Device shall return the first relevant data octet as OD_0 and all subsequent ODs filled with either "0x00" or with subsequent data octets if appropriate. Master shall evaluate only the octet in OD_0.

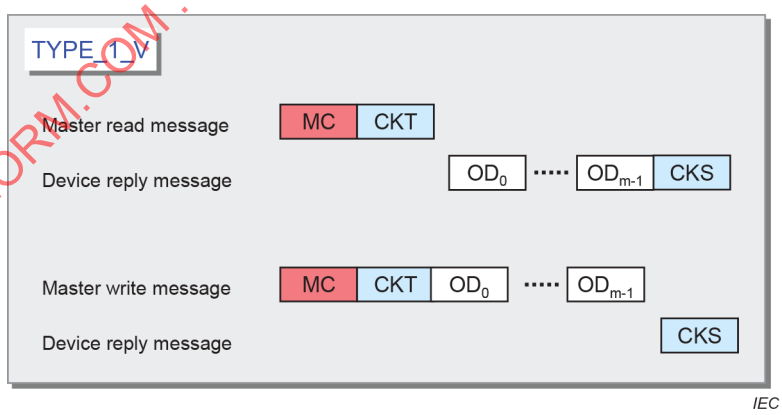


Figure A.8 – M-sequence TYPE_1_V

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.2.4 M-sequence TYPE_2_x

M-sequence TYPE_2_x is optional for all Devices. M-sequences TYPE_2_1 through TYPE_2_5 are defined. M-sequence TYPE_2_V provides variable (extendable) message length. M-sequence TYPE_2_x transmits Process Data and On-request Data in one message. The number of process and On-request Data read or written in each cycle depends on the type. The Address parameter (see Figure A.1) belongs in this case to the on-request communication channel. The Process Data address is specified implicitly starting at "0". The format of Process Data characterizes the M-sequence TYPE_2_x.

M-sequence TYPE_2_1 transmits one octet of read Process Data and one octet of read or write On-request Data per cycle. This M-sequence type is shown in Figure A.9.

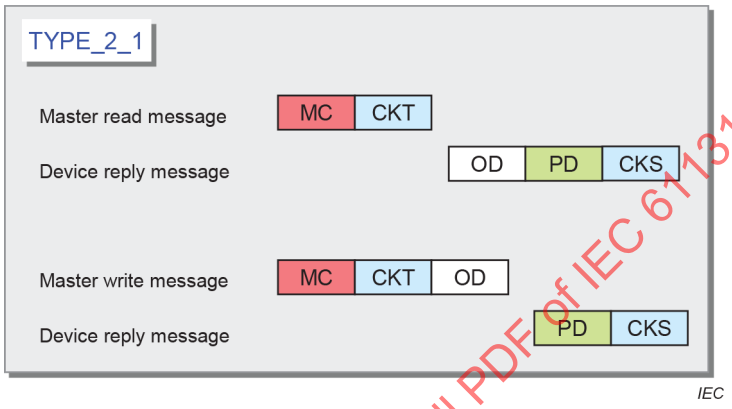


Figure A.9 – M-sequence TYPE_2_1

M-sequence TYPE_2_2 transmits 2 octets of read Process Data and one octet of On-request Data per cycle. This M-sequence type is shown in Figure A.10.

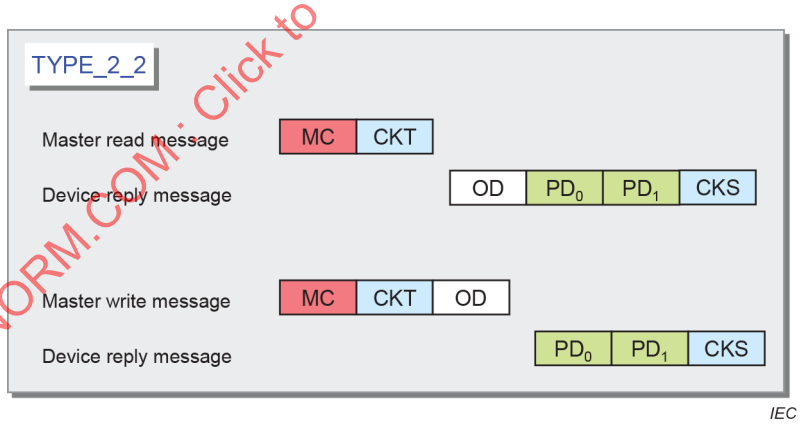


Figure A.10 – M-sequence TYPE_2_2

M-sequence TYPE_2_3 transmits one octet of write Process Data and one octet of read or write On-request Data per cycle. This M-sequence type is shown in Figure A.11.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

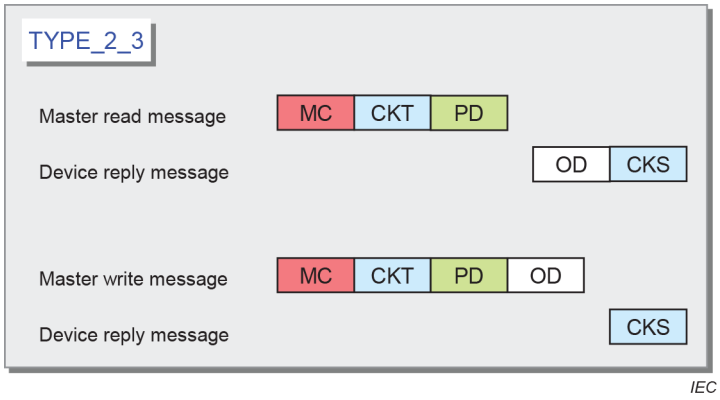


Figure A.11 – M-sequence TYPE_2_3

M-sequence TYPE_2_4 transmits 2 octets of write Process Data and one octet of read or write On-request Data per cycle. This M-sequence type is shown in Figure A.12.

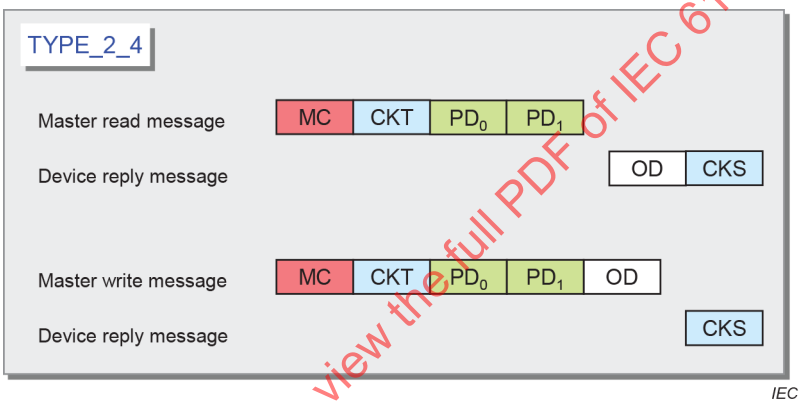


Figure A.12 – M-sequence TYPE_2_4

M-sequence TYPE_2_5 transmits one octet of write and read Process Data and one octet of read or write On-request Data per cycle. This M-sequence type is shown in Figure A.13.

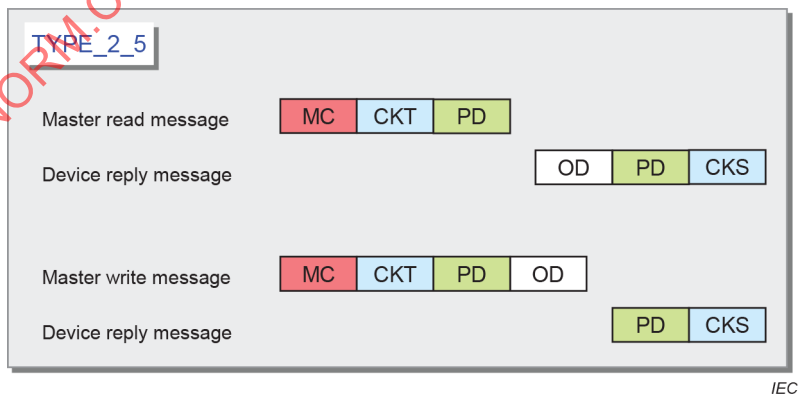


Figure A.13 – M-sequence TYPE_2_5

M-sequence TYPE_2_V transmits the entire write (read) ProcessDataIn n (k) octets per cycle. The range of n (k) is 0 to 32. Either PDin or PDout are not existing when n = 0 or k = 0. TYPE_2_V also transmits m octets of (segmented) read or write On-request Data per cycle using the address in Figure A.1. Permitted values for m are 1, 2, 8, and 32. This variable M-sequence type is shown in Figure A.14.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

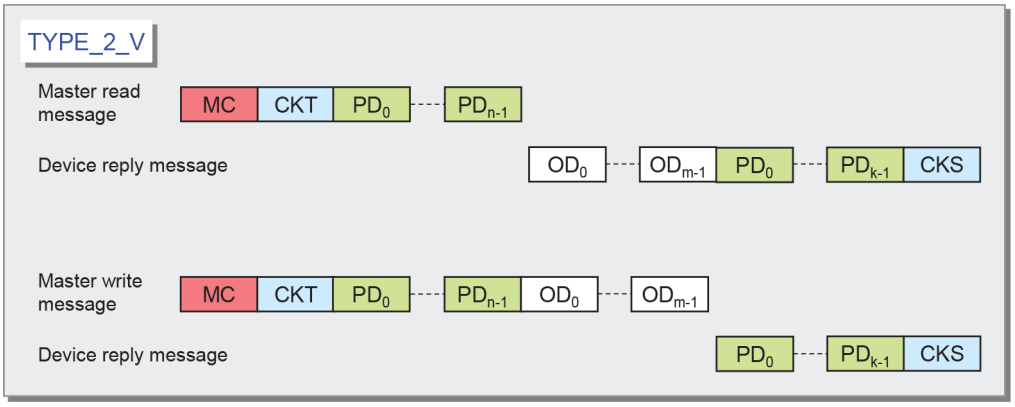


Figure A.14 – M-sequence TYPE_2_V

When using M-sequence TYPE with multi-octet ODs, the rules of M-sequence TYPE_1_V apply (see Figure A.8).

A.2.5 M-sequence type 3

M-sequence type 3 is reserved and shall not be used.

A.2.6 M-sequence type usage for STARTUP, PREOPERATE and OPERATE modes

Table A.7 lists the M-sequence types for the STARTUP mode together with the minimum recovery time (T_{initcyc}) that shall be observed for Master implementations (see A.3.9). The M-sequence code refers to the coding in B.1.4.

Table A.7 – M-sequence types for the STARTUP mode

STARTUP M-sequence code	On-request Data Octets	M-sequence type	Minimum recovery time T_{BIT}
n/a	1	TYPE_0	100

Table A.8 lists the M-sequence types for the PREOPERATE mode together with the minimum recovery time (T_{initcyc}) that shall be observed for Master implementations.

Table A.8 – M-sequence types for the PREOPERATE mode

PREOPERATE M-sequence code	On-request Data Octets	M-sequence type	Minimum recovery time [a] T_{BIT}
0 [b]	1	TYPE_0	100
1	2	TYPE_1_2	100
2	8	TYPE_1_V	210
3	32	TYPE_1_V	550

[a] The minimum recovery time in PREOPERATE mode is a requirement for the Master.

[b] It is highly recommended for Devices not to use TYPE_0 thus improving error discovery when Master restarts communication.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table A.9 lists the M-sequence types for the OPERATE mode for legacy Devices. The minimum cycle time for Master in OPERATE mode is specified by the parameter "MinCycleTime" of the Device (see B.1.3).

Table A.9 – M-sequence types for the OPERATE mode (legacy protocol)

OPERATE M-sequence code	On-request Data	Process Data (PD)		M-sequence type
	Octets	PDin	PDout	Legacy protocol (see [8])
0	1	0	0	TYPE_0 [a]
1	2	0	0	TYPE_1_2
don't care	2	PDin + PDout > 2 octets		TYPE_1_1/1_2 (interleaved)
don't care	1	1…8 bit	0	TYPE_2_1
don't care	1	9…16 bit	0	TYPE_2_2
don't care	1	0	1…8 bit	TYPE_2_3
don't care	1	0	9…16 bit	TYPE_2_4
don't care	1	1…8 bit	1…8 bit	TYPE_2_5

[a] It is highly recommended for Devices not to use TYPE_0 thus improving error discovery when Master restarts communication.

Table A.10 lists the M-sequence types for the OPERATE mode for Devices according to this document. The minimum cycle time for Master in OPERATE mode is specified by the parameter MinCycleTime of the Device (see B.1.3).

Table A.10 – M-sequence types for the OPERATE mode

OPERATE M-sequence code	On-request Data	Process Data (PD)		M-sequence type
	Octets	PDin	PDout	
0	1	0	0	TYPE_0 [a]
1	2	0	0	TYPE_1_2
6	8	0	0	TYPE_1_V
7	32	0	0	TYPE_1_V
0	1	1…8 bit	0	TYPE_2_1
0	1	9…16 bit	0	TYPE_2_2
0	1	0	1…8 bit	TYPE_2_3
0	1	0	9…16 bit	TYPE_2_4
0	1	1…8 bit	1…8 bit	TYPE_2_5
0	1	9…16 bit	1…16 bit	TYPE_2_V [b]
0	1	1…16 bit	9…16 bit	TYPE_2_V [b]
4	1	0…32 octets	3…32 octets	TYPE_2_V
4	1	3…32 octets	0…32 octets	TYPE_2_V
5	2	> 0 bit, octets	≥ 0 bit, octets	TYPE_2_V
5	2	≥ 0 bit, octets	> 0 bit, octets	TYPE_2_V
6	8	> 0 bit, octets	≥ 0 bit, octets	TYPE_2_V
6	8	≥ 0 bit, octets	> 0 bit, octets	TYPE_2_V
7	32	> 0 bit, octets	≥ 0 bit, octets	TYPE_2_V
7	32	≥ 0 bit, octets	> 0 bit, octets	TYPE_2_V

[a] It is highly recommended for Devices not to use TYPE_0 thus improving error discovery when Master restarts communication.

[b] Former TYPE_2_6 has been replaced in support of TYPE_2_V due to inefficiency.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.3 Timing constraints

A.3.1 General

The interactions of a Master and its Device are characterized by several time constraints that apply to the UART frame, Master and Device message transmission times, supplemented by response, cycle, delay, and recovery times.

A.3.2 Bit time

The bit time T_{BIT} is the time it takes to transmit a single bit. It is the inverse value of the transmission rate (see Formula (A.2)).

$$T_{BIT} = 1/(\text{transmission rate}) \quad \text{(A.2)}$$

Values for T_{BIT} are specified in Table 9.

A.3.3 UART frame transmission delay of Master (Ports)

The UART frame transmission delay t_1 of a Port is the duration between the end of the stop bit of a UART frame and the beginning of the start bit of the next UART frame. The Port shall transmit the UART frames within a maximum delay of one bit time (see Formula (A.3)).

$$0 \le t_1 \le 1\ T_{BIT} \quad \text{(A.3)}$$

A.3.4 UART frame transmission delay of Devices

The Device's UART frame transmission delay t_2 is the duration between the end of the stop bit of a UART frame and the beginning of the start bit of the next UART frame. The Device shall transmit the UART frames within a maximum delay of 3 bit times (see Formula (A.4)).

$$0 \le t_2 \le 3\ T_{BIT} \quad \text{(A.4)}$$

A.3.5 Response time of Devices

The Device's response time t_A is the duration between the end of the stop bit of a Port's last UART frame being received and the beginning of the start bit of the first UART frame being sent. The Device shall observe a delay of at least one bit time but no more than 10 bit times (see Formula (A.5)).

$$1\ T_{BIT} \le t_A \le 10\ T_{BIT} \quad \text{(A.5)}$$

A.3.6 M-sequence time

Communication between a Port and and its associated Device takes place in a fixed schedule, called the M-sequence time (see Formula (A.6)).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

$$t_{\text{M-sequence}} = (m+n) \times 11 \times T_{\text{BIT}} + t_{\text{A}} + (m\text{-}1) \times t_1 + (n\text{-}1) \times t_2 \quad \text{(A.6)}$$

In this formula, m is the number of UART frames sent by the Port to the Device and n is the number of UART frames sent by the Device to the Port. The formula can only be used for estimates as the times t_1 and t_2 could not be constant.

Figure A.15 demonstrates the timings of an M-sequence consisting of a Master (Port) message and a Device message.

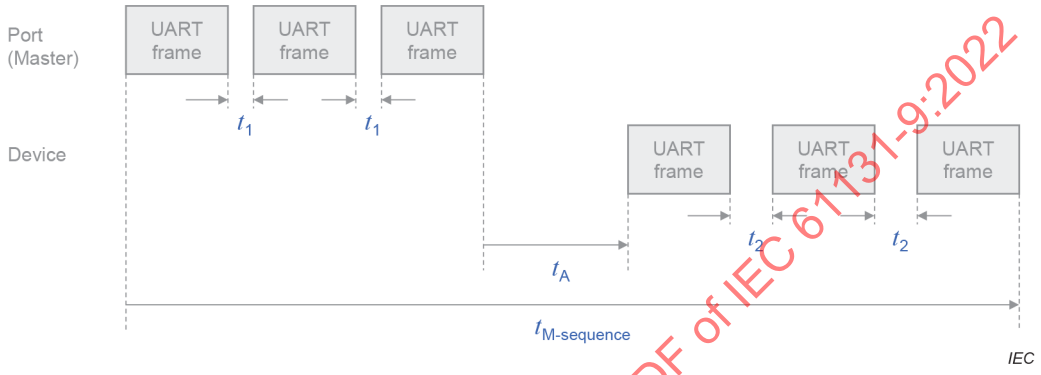


Figure A.15 – M-sequence timing

A.3.7 Cycle time

The cycle time t_{CYC} (see Formula (A.7)) depends on the Device's parameter “MinCycleTime” and the design and implementation of a Master and the number of Ports.

$$t_{\text{CYC}} = t_{\text{M-sequence}} + t_{\text{idle}} \quad \text{(A.7)}$$

The adjustable Device parameter "MasterCycleTime" can be used for the design of a Device specific technology such as an actuator to derive the timing conditions for a default appropriate action such as de-activate or de-energize the actuator (see 7.3.3.5 "MaxCycleTime", 10.2, and 10.8.3).

Table A.11 lists recommended minimum cycle time values for the specified transmission mode of a Port. The values are calculated based on M-sequence Type_2_1.

Table A.11 – Recommended MinCycleTimes

Transmission mode	t_{CYC}
COM1	18,0 ms
COM2	2,3 ms
COM3	0,4 ms

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.3.8 Idle time

The idle time t_{idle} results from the configured cycle time t_{CYC} and the M-sequence time $t_{M\text{-}sequence}$. With reference to a Port, it comprises the time between the end of the message of a Device and the beginning of the next message from the Master (Port).

The idle time shall be long enough for the Device to become ready to receive the next message.

A.3.9 Recovery time

The Master shall wait for a recovery time $t_{initcyc}$ between any two subsequent acyclic Device accesses while in the STARTUP or PREOPERATE phase (see A.2.6). Recovery time is defined between the beginnings of two subsequent Master requests. Calculations shall refer to Formula (A.7).

A.4 Errors and remedies

A.4.1 UART errors

A.4.1.1 Parity errors

The UART parity bit (see Figure 21) and the checksum (see A.1.6) are two independent mechanisms to secure the data transfer. This means that for example two bit errors in different octets of a message, which are resulting in the correct checksum, can also be detected. Both mechanisms lead to the same error processing.

Remedy: The Master shall repeat the Master message two times (see 7.2.2.1). Devices shall reject all data with detected errors and create no reaction.

A.4.1.2 UART framing errors

The conditions for the correct detection of a UART frame are specified in 5.3.3.2. Error processing shall take place whenever perturbed signal shapes or incorrect timings lead to an invalid UART stop bit.

Remedy: See A.4.1.1.

A.4.2 Wake-up errors

The wake-up current pulse is specified in 5.3.3.3 and the wake-up procedures in 7.3.2.1. Several faults are possible during the attempts to establish communication.

Remedy: Retries are possible. See 7.3.2.1 for details.

A.4.3 Transmission errors

A.4.3.1 Checksum errors

The checksum mechanism is specified in A.1.6. Any checksum error leads to an error processing.

Remedy: See A.4.1.1.

A.4.3.2 Timeout errors

The diverse timing constraints with M-sequences are specified in Clause A.3. Master (Ports) and Devices are checking several critical timings such as lack of synchronism within messages.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Remedy: See A.4.1.1.

A.4.3.3 Collisions

A collision occurs whenever the Master and Device are sending simultaneously due to an error. This error is interpreted as a faulty M-sequence.

Remedy: See A.4.1.1.

A.4.4 Protocol errors

A protocol error occurs for example whenever the sequence of the segmented transmission of an ISDU is wrong (see flow control case in A.1.2).

Remedy: Abort of service with ErrorType information (see Annex C).

A.5 General structure and encoding of ISDUs

A.5.1 Overview

The purpose and general structure of an ISDU is specified in 7.3.6.1. Subclauses A.5.2 to A.5.7 provide a detailed description of the individual elements of an ISDU and some examples.

A.5.2 I-Service

Figure A.16 shows the structure of the I-Service octet.

Figure A.16 – I-Service octet

Bits 0 to 3: Length

The encoding of the nibble Length of the ISDU is specified in Table A.14 .

Bits 4 to 7: I-Service

The encoding of the nibble I-Service of the ISDU is specified in Table A.12.

All other elements of the structure specified in 7.3.6.1 are transmitted as independent octets.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table A.12 – Definition of the nibble "I-Service"

I-Service (binary)	Definition		Index format
	Master	Device	
0000	No Service	No Service	n/a
0001	Write Request	Reserved[a]	8-bit Index
0010	Write Request	Reserved	8-bit Index and Subindex
0011	Write Request	Reserved	16-bit Index and Subindex
0100	Reserved	Write Response (-)	none
0101	Reserved	Write Response (+)	none
0110	Reserved	Reserved	
0111	Reserved	Reserved	
1000	Reserved	Reserved	
1001	Read Request	Reserved	8-bit Index
1010	Read Request	Reserved	8-bit Index and Subindex
1011	Read Request	Reserved	16-bit Index and Subindex
1100	Reserved	Read Response (−)	none
1101	Reserved	Read Response (+)	none
1110	Reserved	Reserved	
1111	Reserved	Reserved	

[a] See 3.3.6.

Table A.13 specifies the syntax of the ISDUs. ErrorType can be found in Annex C.

Table A.13 – ISDU syntax

ISDU name	ISDU structure
Write Request	{I-Service(0x1), LEN, Index, [Data*], CHKPDU} ^ {I-Service(0x2), LEN, Index, Subindex, [Data*], CHKPDU} ^ {I-Service(0x3), LEN, Index, Index, Subindex, [Data*], CHKPDU}
Write Response (+)	I-Service(0x5), Length(0x2), CHKPDU
Write Response (−)	I-Service(0x4), Length(0x4), ErrorType, CHKPDU
Read Request	{I-Service(0x9), Length(0x3), Index, CHKPDU} ^ {I-Service(0xA), Length(0x4), Index, Subindex, CHKPDU} ^ {I-Service(0xB), Length(0x5), Index, Index, Subindex, CHKPDU}
Read Response (+)	I-Service(0xD), LEN, [Data*], CHKPDU
Read Response (−)	I-Service(0xC), Length(0x4), ErrorType, CHKPDU
Key LEN = {Length(0x1), ExtLength} ^ {Length}	

A.5.3 Extended length (ExtLength)

The number of octets transmitted in this I-Service, including all protocol information (6 octets), is specified in the "Length" element of an ISDU. If the total length is more than 15 octets, the length is specified using extended length information ("ExtLength"). Permissible values for "Length" and "ExtLength" are listed in Table A.14.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table A.14 – Definition of nibble Length and octet ExtLength

I-Service	Length	ExtLength	Definition
0	0	n/a	No service, ISDU length is 1. Protocol use.
0	1	n/a	Device busy, ISDU length is 1. Protocol use.
0	2 to 15	n/a	Reserved and shall not be used
1 to 15	0	n/a	Reserved and shall not be used
1 to 15	1	0 to 16	Reserved and shall not be used
1 to 15	1	17 to 238	Length of ISDU in "ExtLength"
1 to 15	1	239 to 255	Reserved and shall not be used
1 to 15	2 to 15	n/a	Length of ISDU

A.5.4 Index and Subindex

The parameter address of the data object to be transmitted using the ISDU is specified in the "Index" element. "Index" has a range of values from 0 to 65 535 (see B.2.1 for constraints). Index values 0 and 1 shall be rejected by the Device.

There is no requirement for the Device to support all Index and Subindex values. The Device shall send a negative response to Index or Subindex values not supported.

The data element address of a structured parameter of the data object to be transmitted using the ISDU is specified in the "Subindex" element. "Subindex" has a range of values from 0 to 255, whereby a value of "0" is used to reference the entire data object (see Figure 6).

Table A.15 lists the Index formats used in the ISDU depending on the parameters transmitted.

Table A.15 – Use of Index formats

Index	Subindex	Index format of ISDU
0 to 255	0	8 bit Index
0 to 255	1 to 255	8 bit Index and 8 bit Subindex
256 to 65 535	0 to 255	16 bit Index and 8 bit Subindex (see NOTE)
NOTE See B.2.1 for constraints on the Index range.		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.5.5 Data

The "Data" element can contain the data objects specified in Annex B or Device specific data objects respectively. The data length corresponds to the entries in the "Length" element minus the ISDU protocol elements.

A.5.6 Check ISDU (CHKPDU)

The "CHKPDU" element provides data integrity protection. The sender calculates the value of "CHKPDU" by XOR processing all of the octets of an ISDU, including "CHKPDU" with a preliminary value "0", which is then replaced by the result of the calculation (see Figure A.17).

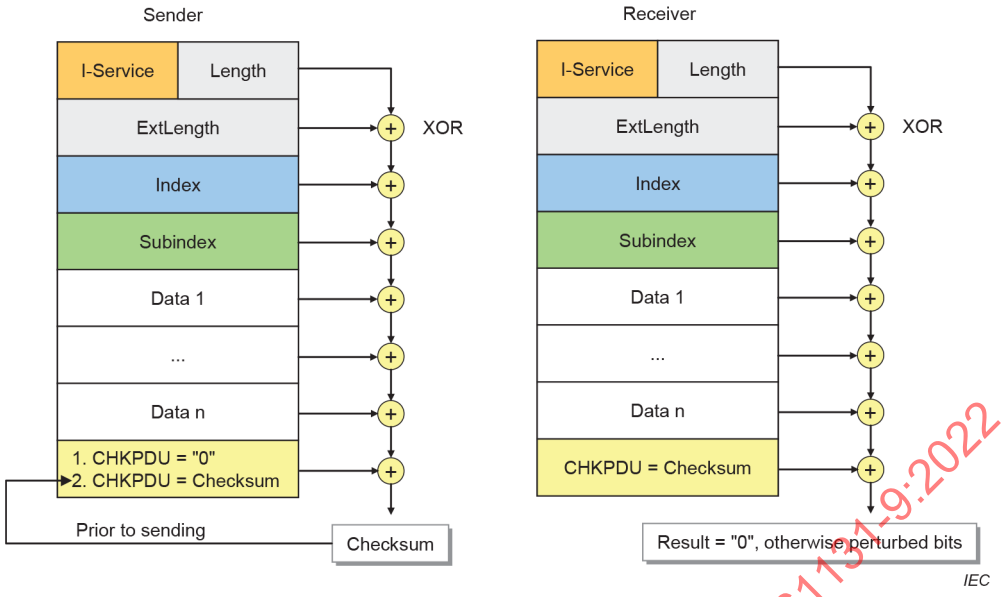
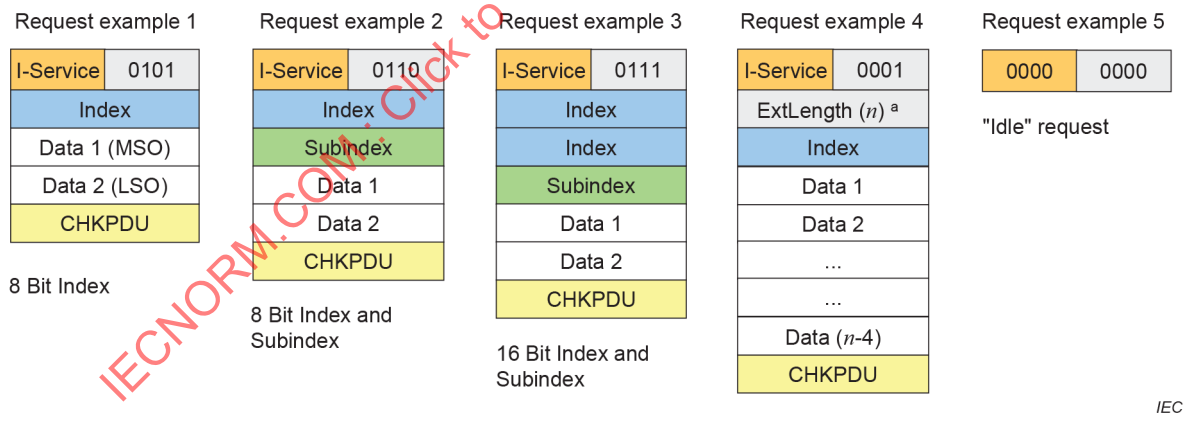


Figure A.17 – Check of ISDU integrity via CHKPDU

The receiver checks whether XOR processing of all of the octets of the ISDU will lead to the result "0" (see Figure A.17). If the result is different from "0", error processing shall take place. See also A.1.6.

A.5.7 ISDU examples

Figure A.18 demonstrates typical examples of request formats for ISDUs, which are explained in the following paragraphs.



[a] Overall ISDU ExtLength = n (1 to 238); Length = 1 ("0001")

Figure A.18 – Examples of request formats for ISDUs

The ISDU request in example 1 comprises one Index element allowing addressing from 0 to 255 (see Table A.15 and Table B.8 for restrictions). In this example the Subindex is "0" and the whole content of Index is Data 1 with the most significant octet (MSO) and Data 2 with the least significant octet (LSO). The total length is 5 ("0101").

The ISDU request in example 2 comprises one Index element allowing addressing from 0 to 255 and the Subindex element allowing addressing an element of a data structure. The total length is 6 ("0110").

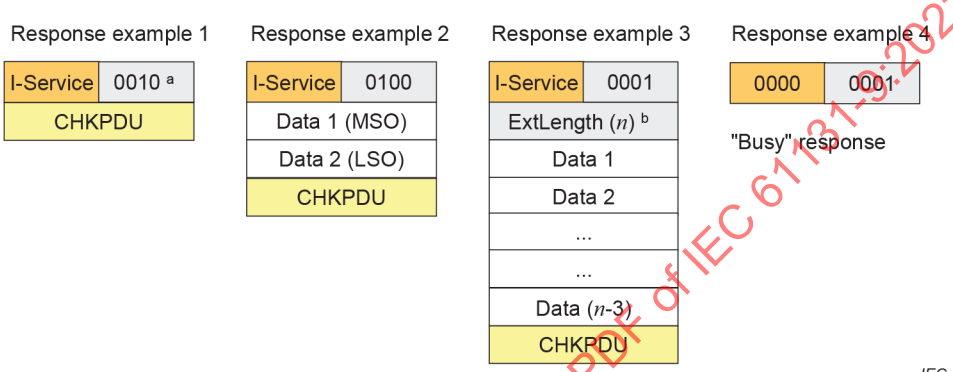
IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The ISDU request in example 3 comprises two Index elements allowing to address from 256 to 65 535 (see Table A.15) and the Subindex element allowing to address an element of a data structure. The total length is 7 ("0111").

The ISDU request in example 4 comprises one Index element and the ExtLength element indicating the number of ISDU elements (n), permitting numbers from 17 to 238. In this case the Length element has the value "1".

The ISDU request "Idle" in example 5 is used to indicate that no service is pending.

Figure A.19 demonstrates typical examples of response ISDUs, which are explained in the following paragraphs.



[a] Minimum length = 2 ("0010")

[b] Overall ISDU ExtLength = n (17 to 238);
Length = 1 ("0001")

Figure A.19 – Examples of response ISDUs

The ISDU response in example 1 shows the minimum value 2 for the Length element ("0010").

The ISDU response in example 2 shows two Data elements and a total number of four elements in the Length element ("0100"). Data 1 carries the most significant octet (MSO) and Data 2 the least significant octet (LSO).

The ISDU response in example 3 shows the ExtLength element indicating the number of ISDU elements (n), permitting numbers from 17 to 238. In this case the Length element has the value "1".

The ISDU response "Busy" in example 4 is used when a Device is currently not able to respond to the read request of the Master due to the necessary preparation time for the response.

Figure A.20 shows a typical example of both a read and a write request ISDU, which are explained in the following paragraphs.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

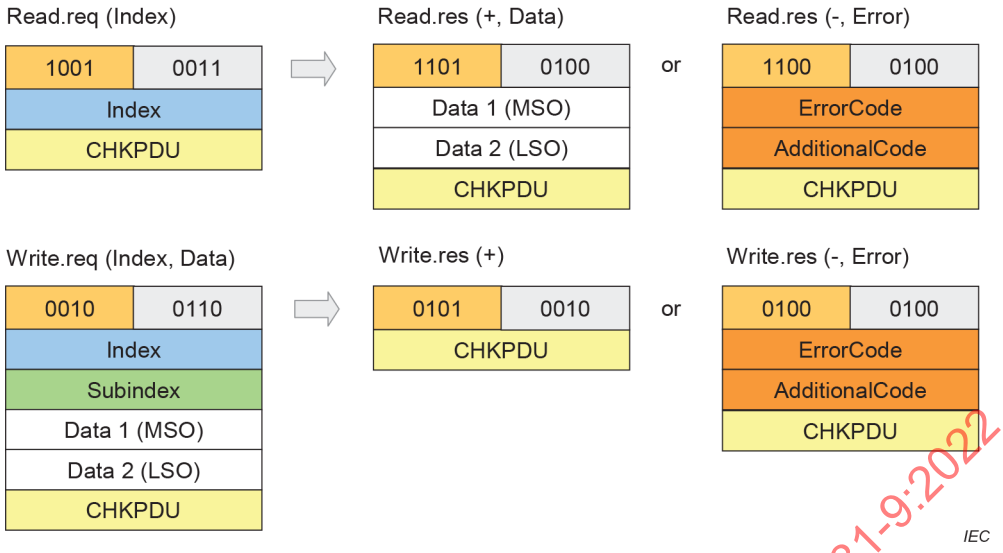


Figure A.20 – Examples of read and write request ISDUs

The code of the read request I-Service is "1001". According to Table A.13 this comprises an Index element. A successful read response (+) of the Device with code "1101" is shown next to the request with two Data elements. Total length is 4 ("0100"). An unsuccessful read response (−) of the Device with code "1100" is shown next in line. It carries the ErrorType with the two Data elements ErrorCode and AdditionalCode (see Annex C).

The code of the write request I-Service is "0010". According to Table A.13 this comprises an Index and a Subindex element. A successful write response (+) of the Device with code "0101" is shown next to the request with no Data elements. Total length is 2 ("0010"). An unsuccessful read response (−) of the Device with code "0100" is shown next in line. It carries the ErrorType with the two Data elements ErrorCode and AdditionalCode (see Annex C).

A.6 General structure and encoding of Events

A.6.1 General

In 7.3.8.1 and Table 58 the purpose and general structure of the Event memory is specified. This memory accommodates a StatusCode, several EventQualifiers and their associated EventCodes. The coding of these memory elements is specified in A.6.2 to A.6.5.

A.6.2 StatusCode type 1 (no details)

Figure A.21 shows the structure of this StatusCode.

NOTE 1 StatusCode type 1 is only used in Events generated by legacy devices (see 7.3.8.1).

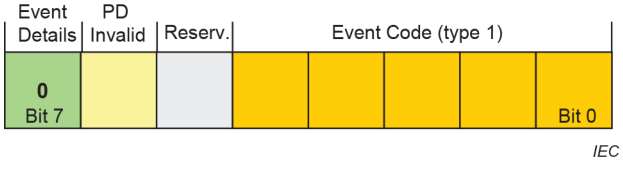


Figure A.21 – Structure of StatusCode type 1

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Bits 0 to 4: EventCode (type 1)

The coding of this data structure is listed in Table A.16. The EventCodes are mapped into EventCodes (type 2) as listed in Annex D. See 7.3.8.2 for additional information.

Table A.16 – Mapping of EventCodes (type 1)

EventCode (type 1)	EventCode (type2)	Instance	Type	Mode
****1	0xFF80	Application	Notification	Event single shot
***1*	0xFF80	Application	Notification	Event single shot
1	0x6320	Application	Notification	Event single shot
*1***	0xFF80	Application	Notification	Event single shot
1****	0xFF10	Application	Notification	Event single shot
Key * Don't care				

Bit 5: Reserved

This bit is reserved and shall be set to zero in StatusCode type 1.

Bit 6: Reserved

NOTE 2 This bit is used in legacy protocol (see [8]) for PDinvalid indication.

Bit 7: Event Details

This bit indicates that no detailed Event information is available. It shall always be set to zero in StatusCode type 1.

A.6.3 StatusCode type 2 (with details)

Figure A.22 shows the structure of the StatusCode type 2.

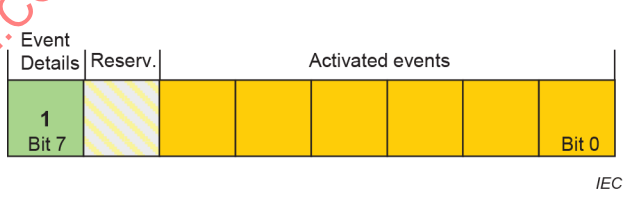


Figure A.22 – Structure of StatusCode type 2

Bits 0 to 5: Activated Events

Each bit is linked to an Event in the memory (see 7.3.8.1) as demonstrated in Figure A.23. Bit 0 is linked to Event 1, bit 1 to Event 2, etc. A bit with value "1" indicates that the corresponding EventQualifier and the EventCode have been entered in valid formats in the memory. A bit with value "0" indicates an invalid entry.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

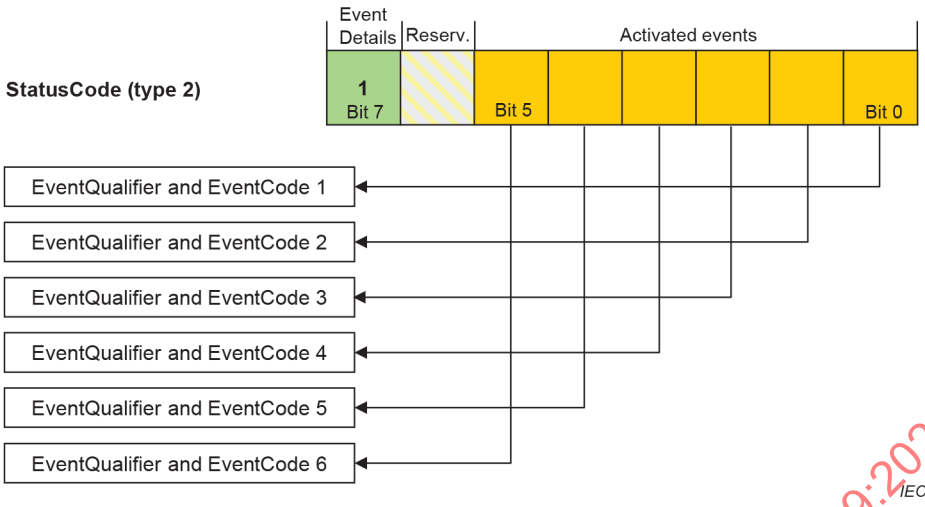


Figure A.23 – Indication of activated Events

Bit 6: Reserved

This bit is reserved and shall be set to zero.

NOTE This bit is used in the legacy protocol version according to [8] for PDinvalid indication.

Bit 7: Event Details

This bit indicates that detailed Event information is available. It shall always be set in StatusCode type 2.

A.6.4 EventQualifier

The structure of the EventQualifier is shown in Figure A.24.

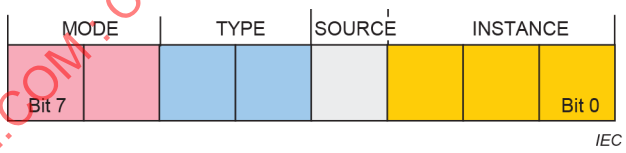


Figure A.24 – Structure of the EventQualifier

Bits 0 to 2: INSTANCE

These bits indicate the particular source (instance) of an Event thus refining its evaluation on the receiver side. Permissible values for INSTANCE are listed in Table A.17.

Table A.17 – Values of INSTANCE

Value	Definition
0	Unknown
1 to 3	Reserved
4	Application (APP)
5	System (SYS)
6 to 7	Reserved

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Bit 3: SOURCE

This bit indicates the source of the Event. Permissible values for SOURCE are listed in Table A.18.

Table A.18 – Values of SOURCE

Value	Definition
0	Device (remote)
1	Master/Port

Bits 4 to 5: TYPE

These bits indicate the Event category. Permissible values for TYPE are listed in Table A.19.

Table A.19 – Values of TYPE

Value	Definition
0	Reserved
1	Notification
2	Warning
3	Error

Bits 6 to 7: MODE

These bits indicate the Event mode. Permissible values for MODE are listed in Table A.20.

Table A.20 – Values of MODE

Value	Definition
0	reserved
1	Event single shot
2	Event disappears
3	Event appears

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.6.5 EventCode

The EventCode entry contains the identifier of an actual Event. Permissible values for EventCode are listed in Annex D.

Annex B
(normative)

Parameters and commands

B.1 Direct Parameter pages 1 and 2

B.1.1 Overview

In principle, the designer of a Device has a large amount of space for parameters and commands as shown in Figure 6. SDCI offers the so-called Direct Parameter pages 1 and 2 with a simplified access method (page communication channel according to Table A.1).

The range of Direct Parameters is structured as shown in Figure B.1. It is split into page 1 and page 2.

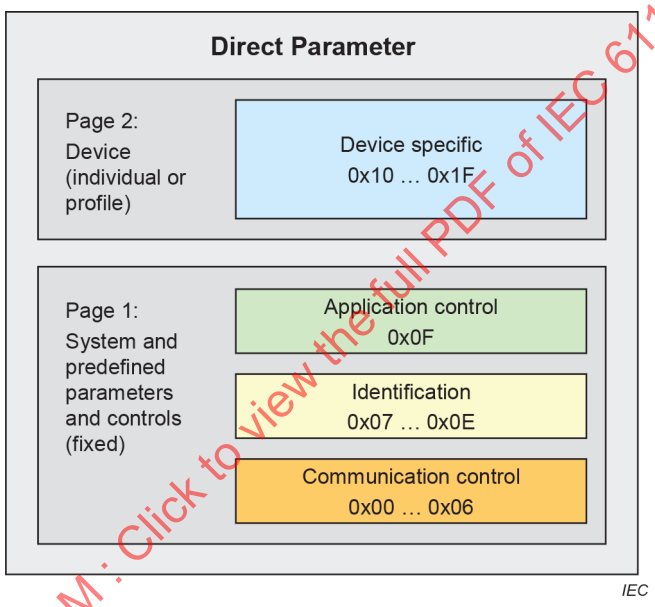


Figure B.1 – Classification and mapping of Direct Parameters

Page 1 ranges from 0x00 to 0x0F. It comprises the following categories of parameters:

– communication control;
– identification parameter;
– application control.

The Master application layer (AL) provides read only access to Direct Parameter page 1 as data objects (see 8.2.1) via Index 0. Single octets can be read via Index 0 and the corresponding Subindex. Subindex 1 indicates address 0x00 and Subindex 16 address 0x0F.

Page 2 ranges from 0x10 to 0x1F. This page comprises parameters optionally used by the individual Device technology. The Master application layer (AL) provides read/write access to Direct Parameter page 2 in form of data objects (see 8.2.1) via Index 1. Single octets can be written or read via Index 1 and the corresponding Subindex. Subindex 1 indicates address 0x10 and Subindex 16 address 0x1F.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A Device shall always return the value "0" upon a read access to Direct Parameter addresses, which are not implemented (for example in case of reserved parameter addresses or not supported optional parameters). The Device shall ignore a write access to not implemented parameters.

The structure of the Direct Parameter pages 1 and 2 is specified in Table B.1.

Table B.1 – Direct Parameter pages 1 and 2

Address	Parameter name	Access	Implementation /reference	Description
Direct Parameter page 1				
0x00	Master-Command	W	Mandatory/ see B.1.2	Master command to switch to operating states (see NOTE 1)
0x01	MasterCycle-Time	R/W	Mandatory/ see B.1.3	Actual cycle duration used by the Master to address the Device. Can be used as a parameter to monitor Process Data transfer.
0x02	MinCycleTime	R	Mandatory/ see B.1.3	Minimum cycle duration supported by a Device. This is a performance feature of the Device and depends on its technology and implementation.
0x03	M-sequence Capability	R	Mandatory/ see B.1.4	Information about implemented options related to M-sequences and physical configuration
0x04	RevisionID	R/W	Mandatory/ see B.1.5	ID of the used protocol version for implementation (shall be set to 0x11)
0x05	ProcessDataIn	R	Mandatory/ see B.1.6	Type and length of input data (Process Data from Device to Master)
0x06	ProcessData-Out	R	Mandatory/ see B.1.7	Type and length of output data (Process Data from Master to Device)
0x07	VendorID 1 (MSB)	R	Mandatory/ see B.1.8	Unique vendor identification (see NOTE 2)
0x08	VendorID 2 (LSB)			
0x09	DeviceID 1 (Octet 2, MSB)	R/W	Mandatory/ see B.1.9	Unique Device identification allocated by a vendor
0x0A	DeviceID 2 (Octet 1)			
0x0B	DeviceID 3 (Octet 0, LSB)			
0x0C	FunctionID 1 (MSB)	R	see B.1.10	Reserved (see Table 102)
0x0D	FunctionID 2 (LSB)			
0x0E		R	reserved	
0x0F	System-Command	W	Optional/ see B.1.11	Command interface for end user applications only and Devices without ISDU support
Direct Parameter page 2				
0x10… 0x1F	Vendor specific	Optional	Optional/ see B.1.12	Device specific parameters
NOTE 1 A read operation returns unspecified values. NOTE 2 VendorIDs are assigned by the IO-Link community.				

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

B.1.2 MasterCommand

The Master application is able to check the status of a Device or to control its behavior with the help of MasterCommands (see 7.3.7).

Permissible values for these parameters are specified in Table B.2.

Table B.2 – Types of MasterCommands

Value	MasterCommand	Description
0x00 to 0x59	Reserved	
0x5A	Fallback	Transition from communication to SIO mode. The Device shall execute this transition after 3 Master-CycleTimes and before 500 ms elapsed after the MasterCommand.
0x5B to 0x94	Reserved	
0x95	MasterIdent	Indicates a Master revision higher than 1.0
0x96	DeviceIdent	Start check of Direct Parameter page for changed entries
0x97	DeviceStartup	Switches the Device from OPERATE or PREOPERATE to STARTUP
0x98	ProcessDataOutputOperate	Process output data valid
0x99	DeviceOperate	Process output data invalid or not available. Switches the Device from STARTUP or PREOPERATE to OPERATE
0x9A	DevicePreoperate	Switches the Device from STARTUP to state PREOPERATE
0x9B to 0xFF	Reserved	

B.1.3 MasterCycleTime and MinCycleTime

The MasterCycleTime is a Master parameter and sets up the actual cycle time of a particular Port.

The MinCycleTime is a Device parameter to inform the Master about the shortest cycle time supported by this Device.

See A.3.7 for the application of the MasterCycleTime and the MinCycleTime. The structure of these two parameters is shown in Figure B.2.

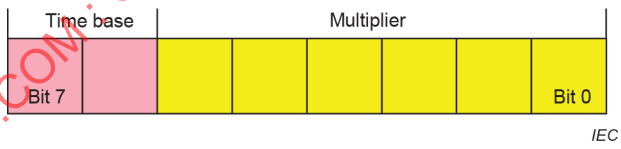


Figure B.2 – MinCycleTime

Bits 0 to 5: Multiplier

These bits contain a 6-bit multiplier for the calculation of MasterCycleTime or MinCycleTime. Permissible values for the multiplier are 0 to 63.

Bits 6 to 7: Time Base

These bits specify the time base for the calculation of MasterCycleTime or MinCycleTime.

When all bits are zero, (binary code 0x00), the Device has no MinCycleTime. In this case the Master shall use the calculated worst case M-sequence timing, that is with the M-sequence type used by the Device, and the maximum times for t_A and t_2 (see A.3.4 to A.3.6).

The permissible combinations for time base and multiplier are listed in Table B.3 along with the resulting values for MasterCycleTime or MinCycleTime.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table B.3 – Possible values of MasterCycleTime and MinCycleTime

Time base encoding	Time Base value	Calculation	Cycle Time
00	0,1 ms	Multiplier × Time Base	0,4 ms to 6,3 ms
01	0,4 ms	6,4 ms + Multiplier × Time Base	6,4 ms to 31,6 ms
10	1,6 ms	32,0 ms + Multiplier × Time Base	32,0 ms to 132,8 ms
11	Reserved	Reserved	Reserved
NOTE The value 0,4 results from the minimum possible transmission time according to A.3.7.			

B.1.4 M-sequenceCapability

The structure of the M-sequenceCapability parameter is shown in Figure B.3.

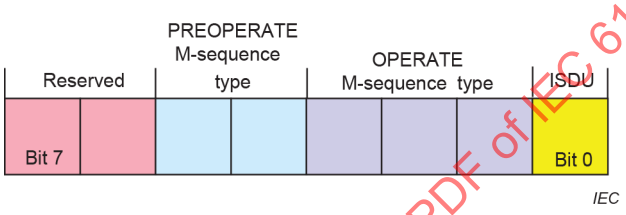


Figure B.3 – M-sequenceCapability

Bit 0: ISDU

This bit indicates whether or not the ISDU communication channel is supported. Permissible values for ISDU are listed in Table B.4.

Table B.4 – Values of ISDU

Value	Definition
0	ISDU not supported
1	ISDU supported

Bits 1 to 3: Coding of the OPERATE M-sequences

This parameter indicates the available M-sequence type during the OPERATE state. Permissible codes for the OPERATE M-sequence type are listed in Table A.9 for legacy Devices and in Table A.10 for Devices according to this document.

Bits 4 to 5: Coding of the PREOPERATE M-sequences

This parameter indicates the available M-sequence type during the PREOPERATE state. Permissible codes for the PREOPERATE M-sequence type are listed in Table A.8.

Bits 6 to 7: Reserved

These bits are reserved and shall be set to zero.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

B.1.5 RevisionID (RID)

The RevisionID parameter is the two-digit version number of the SDCI protocol currently used within the Device. Its structure is shown in Figure B.4. The initial value of RevisionID at powerup is the inherent value for protocol RevisionID. It can be overwritten (see 10.6.3 and Table 101) until the next powerup.

This document specifies protocol version 1.1.

NOTE The legacy protocol version 1.0 is specified in [8].

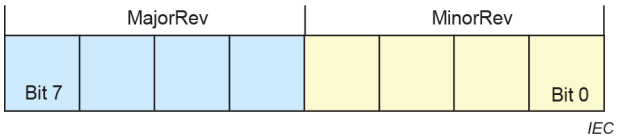


Figure B.4 – RevisionID

Bits 0 to 3: MinorRev

These bits contain the minor digit of the version number, for example 0 for the protocol version 1.0. Permissible values for MinorRev are 0x0 to 0xF.

Bits 4 to 7: MajorRev

These bits contain the major digit of the version number, for example 1 for the protocol version 1.0. Permissible values for MajorRev are 0x0 to 0xF.

B.1.6 ProcessDataIn

The structure of the ProcessDataIn parameter is shown in Figure B.5.

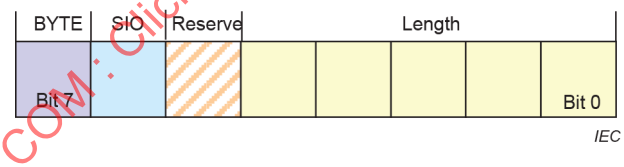


Figure B.5 – ProcessDataIn

Bits 0 to 4: Length

These bits contain the length of the input data (Process Data from Device to Master) in the length unit designated in the BYTE parameter bit. Permissible codes for Length are specified in Table B.6.

Bit 5: Reserved

This bit is reserved and shall be set to zero in this version of this document.

Bit 6: SIO

This bit indicates whether the Device provides a switching signal in SIO mode. Permissible values for SIO are listed in Table B.5.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table B.5 – Values of SIO

Value	Definition
0	SIO mode not supported
1	SIO mode supported

Bit 7: BYTE

This bit indicates the length unit for Length. Permissible values for BYTE and the resulting definition of the Process Data length in conjunction with Length are listed in Table B.6.

Table B.6 – Permitted combinations of BYTE and Length

BYTE	Length	Definition
0	0	no Process Data
0	1	1 bit Process Data, structured in bits
0	2 to 15	*n* bit Process Data, structured in bits
0	16	16 bit Process Data, structured in bits
0	17 to 31	Reserved[a]
1	0, 1	Reserved
1	2	3 octets Process Data, structured in octets
1	3 to 30	(*n* + 1) octets Process Data, structured in octets
1	31	32 octets Process Data, structured in octets
[a] See 3.3.6		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

B.1.7 ProcessDataOut

The structure of the ProcessDataOut parameter is the same as with ProcessDataIn, except with bit 6 ("SIO") reserved.

B.1.8 VendorID (VID)

These octets contain a worldwide unique value per vendor.

NOTE VendorIDs are assigned by the IO-Link community.

B.1.9 DeviceID (DID)

These octets contain the currently used DeviceID. A value of "0" is not permitted. It is highly recommended to store the value of DeviceID in non-volatile memory after a compatibility switch until a reset to the initial value through SystemCommand "Restore factory settings". The value can be overwritten during StartUp (see 10.6.2).

NOTE The communication parameters MinCycleTime, M-sequence Capability, Process Data In and Process Data Out can be changed to achieve compatibility to the requested DeviceID.

B.1.10 FunctionID (FID)

This parameter will be defined in a later version.

B.1.11 SystemCommand

Only Devices without ISDU support shall use the parameter SystemCommand in the Direct Parameter page 1. The implementation of SystemCommand is optional. See Table B.9 for a detailed description of the SystemCommand functions.

NOTE The SystemCommand on the Direct Parameter page 1 does not provide a positive or negative response upon execution of a selected function.

B.1.12 Device specific Direct Parameter page 2

The Device specific Direct Parameters are a set of parameters available to the Device specific technology. The implementation of Device specific Direct Parameters is optional. It is highly recommended for Devices (with ISDU) not to use parameters on Direct Parameter page 2.

NOTE The complete parameter list of the Direct Parameter page 2 is read or write accessible via Index 1 (see B.1.1).

B.2 Predefined Device parameters

B.2.1 Overview

The many different technologies and designs of sensors and actuators require individual and easy access to complex parameters and commands beyond the capabilities of the Direct Parameter page 2. From a Master's point of view, these complex parameters and commands are called application data objects.

Figure B.6 shows the general mapping of data objects for the ISDU transmission.

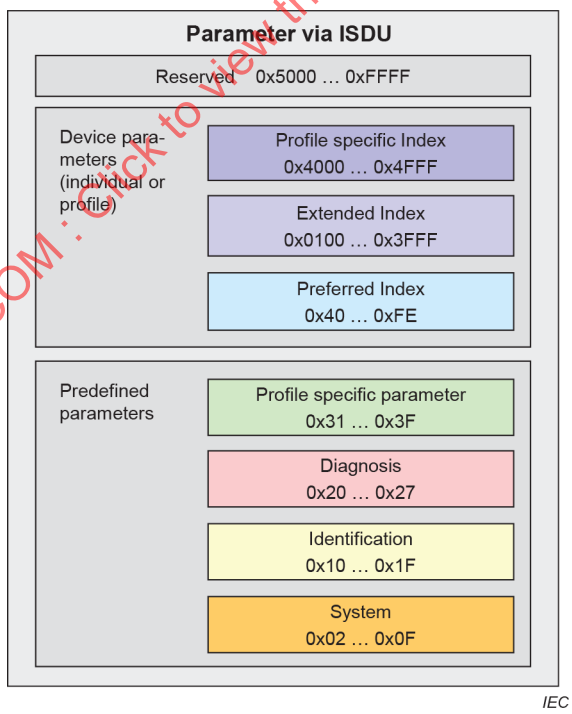


Figure B.6 – Index space for ISDU data objects

So-called ISDU "containers" are the transfer means to exchange application data objects or short data objects. The index of the ISDU is used to address the data objects.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Clause B.2 contains definitions and requirements for the implementation of technology specific Device applications. Implementation rules for parameters and commands are specified in Table B.7.

Table B.7 – Implementation rules for parameters and commands

Rule number	Rule specification
1	All parameters of an Index shall be readable and/or writeable as an entire data object via Subindex 0.
2	The technology specific Device application shall resolve inconsistencies of dependent parameter sets during parameterization.
3	The duration of an ISDU service request is limited (see Table 102). A master application can abort ISDU services after this timeout.
4	Application commands (for example teach-in, reset to factory settings, etc.) are treated like parameters.

Table B.8 specifies the assignment of data objects (parameters and commands) to the Index range of ISDUs. All Indices above 2 are ISDU related.

Table B.8 – Index assignment of data objects (Device parameter)

Index (dec)	Object name	Access	Length	Data type	M/O/C	Remark
0x0000 (0)	Direct Parameter Page 1[a]	R		RecordT	M	Redirected to the page communication channel, see 10.8.5
0x0001 (1)	Direct Parameter Page 2[a]	R/W		RecordT	M	Redirected to the page communication channel, see 10.8.5
0x0002 (2)	System-Command	W	1 octet	UIntegerT	M/O	Command Code Definition (See B.2.2)
0x0003 (3)	Data-Storage-Index	R/W	variable	RecordT	M	Set of data objects for storage (See B.2.3)
0x0004-0x000B (4-11)	Reserved[b]					Reserved for exceptional operations
0x000C (12)	Device-Access-Locks	R/W	2 octets	RecordT	O	Standardized Device locking functions (See B.2.4)
0x000D (13)	Profile-Charac-teristic	R	variable	ArrayT of UIntegerT16	C	Reserved for Common Profile [7] (see B.2.5)
0x000E (14)	PDInput-Descriptor	R	variable	ArrayT of OctetStringT3	C	Reserved for Common Profile [7] (see B.2.6)
0x000F (15)	PDOutput-Descriptor	R	variable	ArrayT of OctetStringT3	C	Reserved for Common Profile [7] (see B.2.7)
0x0010 (16)	Vendor-Name	R	max. 64 octets	StringT NOTE	M	Vendor information (See B.2.8)
0x0011 (17)	Vendor-Text	R	max. 64 octets	StringT NOTE	O	Additional vendor information (See B.2.9)
0x0012 (18)	Product-Name	R	max. 64 octets	StringT NOTE	M	Detailed product or type name (See B.2.10)
0x0013 (19)	ProductID	R	max. 64 octets	StringT NOTE	O	Product or type identification (See B.2.11)
0x0014 (20)	Product-Text	R	max. 64 octets	StringT NOTE	O	Description of Device function or characteristic (See B.2.12)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Index (dec)	Object name	Access	Length	Data type	M/O/C	Remark
0x0015 (21)	Serial-Number	R	max. 16 octets	StringT NOTE	O	Vendor specific serial number (See B.2.13)
0x0016 (22)	Hardware-Revision	R	max. 64 octets	StringT NOTE	O	Vendor specific format (See B.2.14)
0x0017 (23)	Firmware-Revision	R	max. 64 octets	StringT NOTE	O	Vendor specific format (See B.2.15)
0x0018 (24)	Application-Specific-Tag	R/W	min. 16, max. 32 octets	StringT NOTE	C	Tag defined by user (See B.2.16)
0x0019 (25)	Function-Tag	R/W	max. 32 octets	StringT NOTE	C	Reserved for Common Profile [7] (See B.2.17)
0x001A (26)	Location-Tag	R/W	max. 32 octets	StringT NOTE	C	Reserved for Common Profile [7] (See B.2.18)
0x001B-0x001F (27-31)	Reserved					
0x0020 (32)	ErrorCount	R	2 octets	UIntegerT	O	Errors since power-on or reset (See B.2.19)
0x0021-0x0023 (33-35)	Reserved					
0x0024 (36)	Device-Status	R	1 octet	UIntegerT	O	Contains current status of the Device (See B.2.20)
0x0025 (37)	Detailed-Device-Status	R	variable	ArrayT of OctetStringT3	O	See B.2.21
0x0026-0x0027 (38-39)	Reserved					
0x0028 (40)	Process-DataInput	R	PD length	Device specific	O	Read last valid Process Data from PDin channel (See B.2.22)
0x0029 (41)	Process-DataOutput	R	PD length	Device specific	O	Read last valid Process Data from PDout channel (See B.2.23)
0x002-0x002F (42-47)	Reserved					
0x0030 (48)	Offset-Time	R/W	1 octet	RecordT	O	Synchronization of Device application timing to M-sequence timing (See B.2.24)
0x0031-0x003F (49-63)	Reserved for profiles					
0x0040-0x00FE (64-254)	Preferred Index					Device specific (8 bit)
0x00FF (255)	Reserved					
0x0100-0x3FFF (256-16 383)	Extended Index					Device specific (16 bit)
0x4000-0x41FF (16 384-16 895)	Profile specific Index					Reserved for Device profile
0x4200-0x42FF (16 896-17 151)	Safety specific Index					Reserved for Safety system extensions [10]

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Index (dec)	Object name	Access	Length	Data type	M/O/C	Remark
0x4300-0x4FFF (17 152-20 479)	Profile specific Index					Reserved for Device profile
0x5000-0x50FF (20 480-20 735)	Wireless specific Index					Reserved for Wireless system extensions [11]
0x5100-0xFFFF (20 736-65 535)	Reserved					

Key M = mandatory; O = optional; C = conditional (only when using Common Profile [7])

NOTE UTF8 coding required for StringT.

[a] These grayed fields indicate Indices of the special page communication channel

[b] See 3.3.6

B.2.2 SystemCommand

Devices with ISDU support shall use the ISDU Index 0x0002 to receive the SystemCommand. The commands shall be acknowledged. The possible responses are defined in 10.3.7. The timing of the appropriate response is defined together with the SystemCommand functionality.

Implementation of the SystemCommand feature is optional for Devices. The coding of SystemCommands is specified in Table B.9.

Table B.9 – Coding of SystemCommand (ISDU)

Command (hex)	Command (dec)	Command name	H/M/O	Definition
0x00	0	Reserved[a]		
0x01	1	ParamUploadStart	O	Start parameter upload
0x02	2	ParamUploadEnd	O	Stop parameter upload
0x03	3	ParamDownloadStart	O	Start parameter download
0x04	4	ParamDownloadEnd	O	Stop parameter download
0x05	5	ParamDownloadStore	O	Finalize parameterization and start Data Storage
0x06	6	ParamBreak	O	Cancel all Param commands
0x07 to 0x3F	7 to 63	Reserved		
0x40 to 0x7F	64 to 127	Reserved for profiles		
0x80	128	Device reset	O	See 10.7.2
0x81	129	Application reset	H	See 10.7.3
0x82	130	Restore factory settings	O	See 10.7.4
0x83	131	Back-to-box	M	See 10.7.5
0x84 to 0x9F	132 to 159	Reserved		
0xA0 to 0xFF	160 to 255	Vendor specific		

Key H = highly recommended; M = mandatory; O = optional

NOTE See 10.3.

[a] See 3.3.6

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The SystemCommand 0x05 (ParamDownloadStore) shall be implemented according to 10.4.2, whenever the Device provides parameters to be stored via the Data Storage mechanism, i.e. parameter "Index_List" in Index 0x0003 is not empty (see Table B.10).

The implementation of the SystemCommands 0x01 to 0x06 required for Block Parameterization according to 10.3.5 is optional. However, all these commands or none of them shall be implemented (for SystemCommand 0x05 the rule for Data Storage dominates).

See B.1.11 for SystemCommand options on the Direct Parameter page 1.

B.2.3 DataStorageIndex

Table B.10 specifies the DataStorageIndex assignments. Record items shall not be separated by offset gaps. Offsets shall be built according to Table F.19.

Table B.10 – DataStorageIndex assignments

Index	Sub-index	Offset	Access	Parameter Name	Coding	Data type
0x0003	01	N + 72	R/W	DS_Command	0x00: Reserved 0x01: DS_UploadStart 0x02: DS_UploadEnd 0x03: DS_DownloadStart 0x04: DS_DownloadEnd 0x05: DS_Break 0x06 to 0xFF: Reserved	UIntegerT8 (8 bit)
	02	N + 64	R	State_Property	Bit 0: Reserved Bit 1 and 2: State of Data Storage 0b00: Inactive 0b01: Upload 0b10: Download 0b11: Data Storage locked Bit 3 to 6: Reserved Bit 7: DS_UPLOAD_FLAG "1": DS_UPLOAD_REQ pending "0": no DS_UPLOAD_REQ	UIntegerT8 (8 bit)
	03	N + 32	R	Data_Storage_ Size	Number of octets for storing all the necessary information for the Device replacement (see 10.4.5). Maximum size is 2 048 octets.	UIntegerT32 (32 bit)
	04	N	R	Parameter_ Checksum	Parameter set revision indication: CRC signature or Revision Counter (see 10.4.8)	UIntegerT32 (32 bit)
	05	0	R	Index_List	List of parameter Indices to be saved (see Table B.11)	OctetStringT (variable)
NOTE $N = (n \times 3 + 2) \times 8$; for n see Table B.11.						

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The parameter DataStorageIndex 0x0003 contains all the information to be used for the Data Storage handling. This parameter is reserved for private exchanges between the Master and the Device; the Master shall block any write access request from a gateway application to this Index (see Figure 5). The parameters within this Index 0x0003 are specified as follows.

DS_Command

This octet carries the Data Storage commands for the Device.

State_Property

This octet indicates the current status of the Data Storage mechanism. Bit 7 shall be stored in non-volatile memory. The Master checks this bit at start-up and performs a parameter upload if requested.

Data_Storage_Size

These four octets provide the requested memory size as number of octets for storing all the information required for the replacement of a Device including the structural information (Index, Subindex). Data type is UIntegerT32 (32 bit). The maximum size is 2 048 octets. See Table G.1 for the elements to be taken into account in the size calculation.

Parameter_Checksum

This checksum is used to detect changes in the parameter set without reading all parameters. The value of the checksum is calculated according to the procedure in 10.4.8. The Device shall change the checksum whenever a parameter out of the parameter set has been altered. Different parameter sets shall hold different checksums. It is recommended that the Device stores this parameter locally in non-volatile memory.

Index_List

Table B.11 specifies the structure of the Index_List. Each Index_List can carry up to 70 entries (see Table 102).

Table B.11 – Structure of Index_List

Entry	Address	Definition	Data type
X1	Index	Index of first parameter to be saved	Unsigned16
	Subindex	Subindex of first parameter to be saved	Unsigned8
X2	Index	Index of next parameter to be saved	Unsigned16
	Subindex	Subindex of next parameter to be saved	Unsigned8
.....	.....	.....	.....
X*n*	Index	Index of last parameter to be saved	Unsigned16
	Subindex	Subindex of last parameter to be saved	Unsigned8
X*n*+1	Index	Termination_Marker 0x0000: End of Index_List >0x0000: Next Index containing an Index_List	Unsigned16

Large sets of parameters can be handled via concatenated Index_Lists. The last two octets of the Index_List shall carry the Termination Marker. A value "0" indicates the end of the Index List. In case of concatenation the Termination Marker is set to the next Index containing an Index List. The structure of the following Index List is the same as specified in Table B.11. Thus, the concatenation of lists ends if a Termination Marker with the value "0" is found.

B.2.4 DeviceAccessLocks

The parameter DeviceAccessLocks allows control of the Device behavior. Standardized Device functions can independently be configured via defined flags in this parameter. The DeviceAccessLocks configuration can be changed by overwriting the parameter. The actual configuration setting is available per read access to this parameter. The data type is RecordT of BooleanT. Access is only permitted via Subindex 0.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

This parameter is optional. If implemented, it shall be non-volatile.

The following Device access lock categories are specified:

– Parameter write access (obsolete);
– Data Storage (obsolete);
– Local parameterization (optional);
– Local user interface operation (optional).

Table B.12 lists the Device locking possibilities.

Table B.12 – Device locking possibilities

Bit	Category	Definition
0	Parameter (write) access	0: unlocked (default) 1: locked (highly recommended not to implement/use)
1	Data Storage	0: unlocked (default) NOTE 1: locked (highly recommended not to implement/use)
2	Local parameterization (optional)	0: unlocked (default) 1: locked
3	Local user interface (optional)	0: unlocked (default) 1: locked
4 to 15	Reserved	
NOTE For compatibility reasons, the Master still reads the parameter State_Property/State of Data Storage (see Table B.10).		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Parameter (write) access:

If this bit is set, write access to all Device parameters over the SDCI communication interface is inhibited for all read/write parameters of the Device except the parameter Device Access Locks. Read access is not affected. The Device shall respond with the negative service response – access denied – to a write access if the parameter access is locked.

The parameter (write) access lock mechanism shall not block downloads of the Data Storage mechanism (between DS_DownloadStart and DS_DownloadEnd or DS_Break).

Data Storage:

If this bit is set in the Device, the Data Storage mechanism is disabled (see 10.4.2 and 11.4.4). In this case, the Device shall respond to a write access (within its Data Storage Index) with a negative service response – access denied – (see B.2.3). Read access to its DataStorageIndex is not affected.

This setting is also indicated in the State Property within Data Storage Index.

Local parameterization:

If this bit is set, the parameterization via local control elements on the Device is inhibited (write protection). Read only is possible (see 10.6.7).

Local user interface:

If this bit is set, operation of the human machine interface on the Device is disabled (see 10.6.8).

B.2.5 ProfileCharacteristic

This parameter contains the list of ProfileIdentifiers (PID's) corresponding to the Device Profile implemented in the Device.

NOTE Details are provided in [7].

B.2.6 PDInputDescriptor

This parameter contains the description of the data structure of the process input data for a profile Device.

NOTE Details are provided in [7].

B.2.7 PDOutputDescriptor

This parameter contains the description of the data structure of the process output data for a profile Device.

NOTE Details are provided in [7].

B.2.8 VendorName

The parameter VendorName contains only one of the vendor names listed for the assigned VendorID. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is mandatory.

NOTE The list of vendor names associated with a given VendorID is maintained by the IO-Link community.

B.2.9 VendorText

The parameter VendorText contains additional information about the vendor. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is optional.

B.2.10 ProductName

The parameter ProductName contains the complete product name. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is mandatory.

NOTE The corresponding entry in the IODD Device variant list is expected to match this parameter.

B.2.11 ProductID

The parameter ProductID shall contain the vendor specific product or type identification of the Device. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is optional.

B.2.12 ProductText

The parameter ProductText shall contain additional product information for the Device, such as product category (for example photoelectric background suppression, ultrasonic distance sensor, pressure sensor, etc.). The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is optional.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

B.2.13 SerialNumber

The parameter SerialNumber shall contain a unique vendor specific notation for each individual Device. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 16. This parameter is optional.

B.2.14 HardwareRevision

The parameter HardwareRevision shall contain a vendor specific notation for the hardware revision of the Device. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is optional.

B.2.15 FirmwareRevision

The parameter FirmwareRevision shall contain a vendor specific notation for the firmware revision of the Device. The parameter is a read-only data object. The data type is StringT with a maximum fixedLength of 64. This parameter is optional.

B.2.16 ApplicationSpecificTag

The parameter ApplicationSpecificTag shall be provided as read/write data object for the user application. It can serve as a free user specific tag. The data type is StringT with a minimum fixedLength of 16, and a preferred fixedLength of 32 octets (see [7]). As default it is recommended to fill this parameter with "***". This parameter is conditional.

B.2.17 FunctionTag

The parameter FunctionTag contains the description of the specific function of a profile Device within an application. As default it is recommended to fill this parameter with "***". This parameter is conditional.

NOTE Details are provided in [7].

B.2.18 LocationTag

The parameter LocationTag contains the description of the location of a profile Device within an application. As default it is recommended to fill this parameter with "***". This parameter is conditional.

NOTE Details are provided in [7].

B.2.19 ErrorCount

The parameter ErrorCount provides information on errors occurred in the Device application since power-on or reset. Usage of this parameter is vendor or Device specific. The data type is UIntegerT with a bitLength of 16. The parameter is a read-only data object. This parameter is optional.

B.2.20 DeviceStatus

B.2.20.1 Overview

The parameter DeviceStatus shall provide information about the Device condition (diagnosis) by the Device's technology. The data type is UIntegerT with a bitLength of 8. The parameter is a read-only data object. This parameter is optional.

The following Device conditions in Table B.13 are specified. They shall be generated by the Device applications. The parameter DeviceStatus can be read by any PLC program or tools such as Asset Management (see Clause 11).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table B.13 lists the different DeviceStatus information. The criteria for these indications are specified in B.2.20.2 through B.2.20.5.

Table B.13 – DeviceStatus parameter

Value	Definition
0	Device is operating properly
1	Maintenance-Required (see B.2.20.2)
2	Out-of-Specification (see B.2.20.3)
3	Functional-Check (see B.2.20.4)
4	Failure (see B.2.20.5)
5 to 255	Reserved

B.2.20.2 Maintenance-Required

Although the Process Data are valid, internal diagnostics indicate that the Device is close to lose its ability of correct functioning.

EXAMPLE Optical lenses getting dusty, build-up of deposits, lubricant level low.

B.2.20.3 Out-of-Specification

Although the Process Data are valid, internal diagnostics indicate that the Device is operating outside its specified measuring range or environmental conditions.

EXAMPLE Power supply, auxiliary energy, temperature, pneumatic pressure, magnetic interference, vibrations, acceleration, interfering light, bubble formation in liquids.

B.2.20.4 Functional-Check

Process Data are temporarily invalid due to intended manipulations on the Device.

EXAMPLE Calibrations, teach-in, position adjustments, and simulation.

B.2.20.5 Failure

Process Data are invalid due to malfunction in the Device or its peripherals. The Device is unable to perform its intended function.

B.2.21 DetailedDeviceStatus

The parameter DetailedDeviceStatus shall provide information about currently pending Events in the Device. Events of TYPE "Error" or "Warning" and MODE "Event appears" (see A.6.4) shall be entered into the list of DetailedDeviceStatus with EventQualifier and EventCode. Upon occurrence of an Event with MODE "Event disappears", the corresponding entry in DetailedDeviceStatus shall be set to EventQualifier "0x00" and EventCode "0x0000". This way this parameter always provides the current diagnosis status of the Device. The parameter is a read-only data object. The data type is ArrayT with a maximum number of 64 array elements (Event entries). The number of array elements of this parameter is Device specific. Upon power-off or reset of the Device the contents of all array elements are set to initial settings – EventQualifier "0x00", EventCode "0x0000". This parameter is optional.

Table B.14 specifies the structure of the parameter DetailedDeviceStatus.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table B.14 – DetailedDeviceStatus (Index 0x0025)

Sub-index	Object name	Data Type	Comment
1	Error_Warning_1	3 octets	All octets 0x00: no Error/ Warning Octet 1: EventQualifier Octet 2,3: EventCode
2	Error_Warning_2	3 octets	
3	Error_Warning_3	3 octets	
4	Error_Warning_4	3 octets	
…			
n	Error_Warning_n	3 octets	

The designer can choose the implementation of a static list, i.e. one fix array position for each Event with a specific EventCode, or a dynamic list, i.e. each Event entry is stored into the next free array position. Subindex access is not supported.

B.2.22 ProcessDataInput

The parameter ProcessDataInput shall provide the last valid process input data from the Device application. The data type and structure are identical to the Process Data In transferred in the process communication channel. The parameter is a read-only data object. This parameter is optional.

B.2.23 ProcessDataOutput

The parameter ProcessDataOutput shall provide the last valid process output data written to the Device application. The data type and structure are identical to the Process Data Out transferred in the process communication channel. The parameter is a read-only data object. This parameter is optional.

B.2.24 OffsetTime

The parameter OffsetTime (t_{offset}) allows a Device application to synchronize on M-sequence cycles of the data link layer via adjustable offset times. The data type is RecordT. Access is only possible via Subindex "0". The parameter is a read/write data object. This parameter is optional.

The structure of the parameter OffsetTime is shown in Figure B.7:

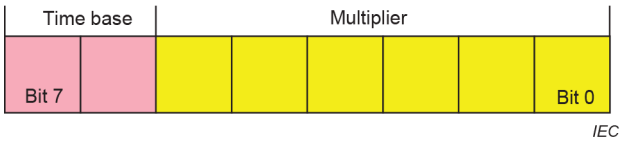


Figure B.7 – Structure of the OffsetTime

Bits 0 to 5: Multiplier

These bits contain a 6-bit factor for the calculation of the OffsetTime. Permissible values for the multiplier are 0 to 63.

Bits 6 to 7: Time Base

These bits contain the time base for the calculation of the OffsetTime.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The permissible combinations for Time Base and Multiplier are listed in Table B.15 along with the resulting values for OffsetTime. Setting both Multiplier and Time Base to zero deactivates synchronization with the help of an OffsetTime. The value of OffsetTime shall not exceed the MasterCycleTime (see B.1.3)

Table B.15 – Time base coding and values of OffsetTime

Time base encoding	Time Base value	Calculation	OffsetTime
00	0,01 ms	Multiplier × Time Base	0,01 ms to 0,63 ms
01	0,04 ms	0,64 ms + Multiplier × Time Base	0,64 ms to 3,16 ms
10	0,64 ms	3,20 ms + Multiplier × Time Base	3,20 ms to 43,52 ms
11	2,56 ms	44,16 ms + Multiplier × Time Base	44,16 ms to 126,08 ms

B.2.25 Profile parameter (reserved)

Indices 0x0031 to 0x003F are reserved for Device profiles.

NOTE Details are provided in [7].

B.2.26 Preferred Index

Preferred Indices (0x0040 to 0x00FE) can be used for vendor specific Device functions. This range of Indices is considered preferred due to lower protocol overhead within the ISDU and thus higher data throughput for small data objects as compared to the Extended Index (see B.2.27).

B.2.27 Extended Index

Extended Indices (0x0100 to 0x3FFF) can be used for vendor specific Device functions.

B.2.28 Profile specific Index (reserved)

Indices 0x4000 to 0x4FFF are reserved for Device profiles.

NOTE Details are provided in [7].

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex C
(normative)

ErrorTypes (ISDU errors)

C.1 General

An ErrorType is used within negative service confirmations of ISDUs (see A.5.2 and Table A.13). It indicates the cause of a negative confirmation of a Read or Write service. The origin of the error could be located in the Master (local) or in the Device (remote).

The ErrorType consists of two octets, the main error cause and more specific information:

– ErrorCode (high order octet);
– AdditionalCode (low order octet).

The ErrorType represents information about the incident, the origin, and the instance. The permissible ErrorTypes and the criteria for their deployment are listed in Clause C.2 and Clause C.3. All other ErrorType values are reserved and shall not be used.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

C.2 Application related ErrorTypes

C.2.1 Overview

The permissible ErrorTypes resulting from the Device application are listed in Table C.1.

Table C.1 – ErrorTypes

Incident	Error Code	Additional Code	Name	Definition
Device application error – no details	0x80	0x00	APP_DEV	See C.2.2
Index not available	0x80	0x11	IDX_NOTAVAIL	See C.2.3
Subindex not available	0x80	0x12	SUBIDX_NOTAVAIL	See C.2.4
Service temporarily not available	0x80	0x20	SERV_NOTAVAIL	See C.2.5
Service temporarily not available – local control	0x80	0x21	SERV_NOTAVAIL_LOCCTRL	See C.2.6
Service temporarily not available – Device control	0x80	0x22	SERV_NOTAVAIL_DEVCTRL	See C.2.7
Access denied	0x80	0x23	IDX_NOT_ACCESSIBLE	See C.2.8
Parameter value out of range	0x80	0x30	PAR_VALOUTOFRNG	See C.2.9
Parameter value above limit	0x80	0x31	PAR_VALGTLIM	See C.2.10
Parameter value below limit	0x80	0x32	PAR_VALLTLIM	See C.2.11
Parameter length overrun	0x80	0x33	VAL_LENOVRRUN	See C.2.12
Parameter length underrun	0x80	0x34	VAL_LENUNDRUN	See C.2.13
Function not available	0x80	0x35	FUNC_NOTAVAIL	See C.2.14
Function temporarily unavailable	0x80	0x36	FUNC_UNAVAILTEMP	See C.2.15
Invalid parameter set	0x80	0x40	PAR_SETINVALID	See C.2.16
Inconsistent parameter set	0x80	0x41	PAR_SETINCONSIST	See C.2.17
Application not ready	0x80	0x82	APP_DEVNOTRDY	See C.2.18

Incident	Error Code	Additional Code	Name	Definition
Vendor specific	0x81	0x00	UNSPECIFIC	See C.2.19
Vendor specific	0x81	0x01 to 0xFF	VENDOR_SPECIFIC	See C.2.19

C.2.2 Device application error – no details

This ErrorType shall be used if the requested service has been refused by the Device application and no detailed information of the incident is available.

C.2.3 Index not available

This ErrorType shall be used whenever a read or write access occurs to a non-existing Index with or without Subindex access.

C.2.4 Subindex not available

This ErrorType shall be used whenever a read or write access occurs to a non-existing Subindex of an existing Index.

C.2.5 Service temporarily not available

This ErrorType shall be used if a parameter is not accessible for a read or write service due to the current state of the Device application.

C.2.6 Service temporarily not available – local control

This ErrorType shall be used if a parameter is not accessible for a read or write service due to an ongoing local operation at the Device (for example operation or parameterization via an on-board Device control panel).

C.2.7 Service temporarily not available – device control

This ErrorType shall be used if a read or write service is not accessible due to a remote triggered state of the device application (for example parameterization during a remote triggered teach-in operation or calibration).

C.2.8 Access denied

This ErrorType shall be used if a Write service tries to access a read-only parameter or if a Read service tries to access a write-only parameter.

C.2.9 Parameter value out of range

This ErrorType shall be used for a write service to a parameter outside its permitted range of values. Example: enumerations (list of single values), combination of value ranges and enumeration.

C.2.10 Parameter value above limit

This ErrorType shall be used for a write service to a parameter above its specified value range.

C.2.11 Parameter value below limit

This ErrorType shall be used for a write service to a parameter below its specified value range.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

C.2.12 Parameter length overrun

This ErrorType shall be used when the content of a write service to a parameter is greater than the parameter specified length. This ErrorType shall also be used if a data object is too large to be processed by the Device application (for example ISDU buffer restriction).

C.2.13 Parameter length underrun

This ErrorType shall be used when the content of a write service to a parameter is less than the parameter specified length (for example write access of an Unsigned16 value to an Unsigned32 parameter).

C.2.14 Function not available

This ErrorType shall be used for a write service with a command value not supported by the Device application (for example a SystemCommand with a value not implemented).

C.2.15 Function temporarily unavailable

This ErrorType shall be used for a write service with a command value calling a Device function not available due to the current state of the Device application (for example a SystemCommand).

C.2.16 Invalid parameter set

This ErrorType shall be used if values sent via single parameter transfer are not consistent with other actual parameter settings (for example overlapping set points for a binary data setting; see 10.3.4).

C.2.17 Inconsistent parameter set

This ErrorType shall be used at the termination of a Block Parameter transfer with ParamDownloadEnd or ParamDownloadStore if the plausibility check shows inconsistencies (see 10.3.5 and B.2.2).

C.2.18 Application not ready

This ErrorType shall be used if a read or write service is refused due to a temporarily unavailable application (for example peripheral controllers during startup).

C.2.19 Vendor specific

This ErrorType will be propagated directly to upper-level processing elements as an error (no warning) by the Master.

C.3 Derived ErrorTypes

C.3.1 Overview

Derived ErrorTypes are generated in the Master AL and are caused by internal incidents or those received from the Device. Table C.2 lists the specified Derived ErrorTypes.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table C.2 – Derived ErrorTypes

Incident	Error Code	Additional Code	Name	Definition
Master – Communication error	0x10	0x00	COM_ERR	See C.3.2
Master – ISDU timeout	0x11	0x00	I-SERVICE_TIMEOUT	See C.3.3
Device Event – ISDU error (DL, Error, single shot, 0x5600)	0x11	0x00	I-SERVICE_TIMEOUT	See C.3.4
Device Event – ISDU illegal service primitive (AL, Error, single shot, 0x5800)	0x11	0x00	I-SERVICE_TIMEOUT	See C.3.5
Master – ISDU checksum error	0x56	0x00	M_ ISDU_CHECKSUM	See C.3.6
Master – ISDU illegal service primitive	0x57	0x00	M_ ISDU_ILLEGAL	See C.3.7
Device Event – ISDU buffer overflow (DL, Error, single shot, 0x5200)	0x80	0x33	VAL_LENOVRRUN	See C.3.8 and C.2.12 Events from legacy Devices shall be redirected in compatibility mode to this derived ErrorType

C.3.2 Master – Communication error

The Master generates a negative service response with this ErrorType if a communication error occurred during a read or write service, for example the SDCI connection is interrupted.

C.3.3 Master – ISDU timeout

The Master generates a negative service response with this ErrorType, if a Read or Write service is pending longer than the specified I-Service timeout (see Table 102) in the Master.

C.3.4 Device Event – ISDU error

If the Master received an Event with the EventQualifier (see A.6.4: DL, Error, Event single shot) and the EventCode 0x5600, a negative service response indicating a service timeout is generated and returned to the requester (see C.3.3).

C.3.5 Device Event – ISDU illegal service primitive

If the Master received an Event with the EventQualifier (see A.6.4: AL, Error, Event single shot) and the EventCode 0x5800, a negative service response indicating a service timeout is generated and returned to the requester (see C.3.3).

C.3.6 Master – ISDU checksum error

The Master generates a negative service response with this ErrorType, if its data link layer detects an ISDU checksum error.

C.3.7 Master – ISDU illegal service primitive

The Master generates a negative service response with this ErrorType, if its data link layer detects an ISDU illegal service primitive.

C.3.8 Device Event – ISDU buffer overflow

If the Master received an Event with the EventQualifier (see A.6.4: DL, Error, Event single shot) and the EventCode 0x5200, a negative service response indicating a parameter length overrun is generated and returned to the requester (see C.2.12).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

C.4 SMI related ErrorTypes

C.4.1 Overview

The Master returns SMI related ErrorTypes within a negative response (Result (-) while performing an SMI service (see 11.2). Table C.3 lists the SMI related ErrorTypes.

Table C.3 – SMI related ErrorTypes

Incident	Error Code	Additional Code	Name
ArgBlock unknown	0x40	0x01	ARGBLOCK_NOT_SUPPORTED
Incorrect ArgBlock content type	0x40	0x02	ARGBLOCK_INCONSISTENT
Device not communicating	0x40	0x03	DEVICE_NOT_ACCESSIBLE
Service unknown	0x40	0x04	SERVICE_NOT_SUPPORTED
Process Data not accessible	0x40	0x05	DEVICE_NOT_IN_OPERATE
Insufficient memory	0x40	0x06	MEMORY_OVERRUN
Incorrect Port number	0x40	0x11	PORT_NUM_INVALID
Incorrect ArgBlock length	0x40	0x34	ARGBLOCK_LENGTH_INVALID
Master busy	0x40	0x36	SERVICE_TEMP_UNAVAILABLE
Inconsistent DS data	0x40	0x39	INCONSISTENT_DS_DATA
Device or Master error	ee	aa	Propagated error, for "ee" and "aa" see Clauses C.2 and C.3
Reserved	0x40	0x80 to 0xFF	Vendor specific

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

C.4.2 ArgBlock unknown

This ErrorType shall be used if the requested ArgBlockID is unknown to the SMI.

C.4.3 Incorrect ArgBlock content type

This ErrorType shall be used if the SMI service detects errors in the structure of the provided ArgBlock.

C.4.4 Device not communicating

This ErrorType shall be used if the Port is not communicating with the Device.

C.4.5 Service unknown

This ErrorType shall be used if a requested SMI service is not supported by the Master.

C.4.6 Process Data not accessible

This ErrorType shall be used if the requested Process Data cannot be accessed in the current state of communication.

C.4.7 Insufficient memory

This ErrorType shall be used if the requested SMI service requires more memory space.

C.4.8 Incorrect Port number

This ErrorType shall be used if the requested Port number is invalid.

C.4.9 Incorrect ArgBlock length

This ErrorType shall be used if the actual ArgBlock length does not correspond to the ArgBlockID.

C.4.10 Master busy

This ErrorType shall be used if the SMI service is blocked due to other running processes.

C.4.11 Inconsistent DS data

This ErrorType shall be used if Data Storage is not supported or Data Storage is not activated on this Port or Data Storage content is not consistent with Port configuration, for example VendorID does not match.

C.4.12 Device or Master error

These ErrorTypes from Device or Master Port are propagated if the requested SMI service has been denied by the Device.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex D
(normative)

EventCodes (diagnosis information)

D.1 General

The concept of Events is described in 7.3.8.1 and the general structure and encoding of Events are specified in Clause A.6. Whenever the StatusCode indicates an Event in case of a Device or a Master incident, the associated EventCode shall be provided as diagnosis information. As specified in Clause A.6, the Event entry contains an EventCode in addition to the EventQualifier. The EventCode identifies an actual incident. Permissible values for EventCode are listed in Table D.1; all other EventCode values are reserved and shall not be used.

D.2 EventCodes for Devices

Table D.1 lists the specified EventCode identifiers and their definitions for Devices (Source = "REMOTE"). The EventCodes are created by the technology specific Device application (instance = APP).

Table D.1 – EventCodes for Devices

EventCode ID	Definition and recommended maintenance action	DeviceStatus Value (NOTE 1)	Type (NOTE 2)
0x0000	No malfunction	0	Notification
0x1000	General malfunction – unknown error	4	Error
0x1001 to 0x17FF	Reserved		
0x1800 to 0x18FF	Vendor specific		
0x1900 to 0x3FFF	Reserved		
0x4000	Temperature fault – Overload	4	Error
0x4001 to 0x420F	Reserved		
0x4210	Device temperature overrun – Clear source of heat	2	Warning
0x4211 to 0x421F	Reserved		
0x4220	Device temperature underrun – Insulate Device	2	Warning
0x4221 to 0x4FFF	Reserved		
0x5000	Device hardware fault – Device exchange	4	Error
0x5001 to 0x500F	Reserved		
0x5010	Component malfunction – Repair or exchange	4	Error
0x5011	Non volatile memory loss – Check batteries	4	Error
0x5012	Batteries low – Exchange batteries	2	Warning
0x5013 to 0x50FF	Reserved		
0x5100	General power supply fault – Check availability	4	Error
0x5101	Fuse blown/open – Exchange fuse	4	Error

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

EventCode ID	Definition and recommended maintenance action	DeviceStatus Value (NOTE 1)	Type (NOTE 2)
0x5102 to 0x510F	Reserved		
0x5110	Primary supply voltage overrun – Check tolerance	2	Warning
0x5111	Primary supply voltage underrun – Check tolerance	2	Warning
0x5112	Secondary supply voltage fault (Port Class B) – Check tolerance	2	Warning
0x5113 to 0x5FFF	Reserved		
0x6000	Device software fault – Check firmware revision	4	Error
0x6001 to 0x631F	Reserved		
0x6320	Parameter error – Check data sheet and values	4	Error
0x6321	Parameter missing – Check data sheet	4	Error
0x6322 to 0x634F	Reserved		
0x6350	Reserved		
0x6351 to 0x76FF	Reserved		
0x7700	Wire break of a subordinate device – Check installation	4	Error
0x7701 to 0x770F	Wire break of subordinate device 1 …device 15 – Check installation	4	Error
0x7710	Short circuit – Check installation	4	Error
0x7711	Ground fault – Check installation	4	Error
0x7712 to 0x8BFF	Reserved		
0x8C00	Technology specific application fault – Reset Device	4	Error
0x8C01	Simulation active – Check operational mode	3	Warning
0x8C02 to 0x8C0F	Reserved		
0x8C10	Process variable range overrun – Process Data uncertain	2	Warning
0x8C11 to 0x8C1F	Reserved		
0x8C20	Measurement range exceeded – Check application	4	Error
0x8C21 to 0x8C2F	Reserved		
0x8C30	Process variable range underrun – Process Data uncertain	2	Warning
0x8C31 to 0x8C3F	Reserved		
0x8C40	Maintenance required – Cleaning	1	Warning
0x8C41	Maintenance required – Refill	1	Warning
0x8C42	Maintenance required – Exchange wear and tear parts	1	Warning
0x8C43 to 0x8C9F	Reserved		
0x8CA0 to 0x8DFF	Vendor specific		
0x8E00 to 0xAFFF	Reserved		
0xB000 to 0xB0FF	Reserved for Safety extensions	See [10]	See [10]

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

EventCode ID	Definition and recommended maintenance action	DeviceStatus Value (NOTE 1)	Type (NOTE 2)
0xB100 to 0xBFFF	Reserved for profiles		
0xC000 to 0xFF90	Reserved		
0xFF91	Data Storage upload request ("DS_UPLOAD_REQ") – internal, not visible to user	0	Notification (single shot)
0xFF92 to 0xFFAF	Reserved		
0xFFB0 to 0xFFB7	Reserved for Wireless extensions	See [11]	See [11]
0xFFB8 to 0xFFFF	Reserved		
NOTE 1 See B.2.20 for a description of this parameter. NOTE 2 See Table A.19 for a description of Event types.			

D.3 EventCodes for Ports

Table D.2 lists the specified EventCode identifiers and their definitions for Ports. The EventCodes are created by the Master (SOURCE = "Master/Port", see Table A.18, and "application" (APP) or "communication system" (SYS) as INSTANCE, see Table A.17). EventCode identifiers 0xFF21 to 0xFFFF are internal system information and shall not be visible to users. The following rules apply:

– Port Events referring to SDCI communication are mandatory (exceptions 0xFF26/0xFF27) and are specified in detail (Event INSTANCE = SYS). The other Port Events (Event INSTANCE = APP) are optional.
– Each appearing Port Event of Type "Error" requires a disappearing Port Event whenever the cause of the Error has been fixed.
– Occurring PortStatusInfo "PORT_DIAG" leads to an appearing EventCode 0x180x or 0x600x depending on "SYS" Error (see Table 126).
– Leaving PortStatusInfo "PORT_DIAG" to others leads to disappearing EventCodes for each pending Error (0x180x).
– Every appearing/disappearing Event leads to an update of the DiagEntry section in the PortStatusList (see Table E.4).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table D.2 – EventCodes for Ports

EventCode ID	Definition and recommended maintenance action	Event INSTANCE	Type
0x0000 to 0x17FF	Reserved		
0x1800	No Device (communication) – Occurring PortStatusInfo "NO_Device" leads to an appearing EventCode 0x1800 – Appearing EventCode 0x1800 causes disappearing of all pending EventCodes of INSTANCE "SYS". – Leaving PortStatusInfo "NO_DEVICE" to others leads to a disappearing EventCode 0x1800	SYS	Error
0x1801	Startup parametrization error – check parameter	APP	Error
0x1802	Incorrect VendorID – Inspection Level mismatch	SYS	Error
0x1803	Incorrect DeviceID – Inspection Level mismatch Trigger: SM_PortMode (COMP_FAULT)	SYS	Error

EventCode ID	Definition and recommended maintenance action	Event INSTANCE	Type
0x1804	Short circuit at C/Q – check wire connection	APP	Error
0x1805	Overtemperature – check Master temperature and load	APP	Error
0x1806	Short circuit at L+ – check wire connection	APP	Error
0x1807	Overcurrent at L+ – check power supply (e.g. L1+)	APP	Error
0x1808	Reserved		
0x1809	Backup inconsistency – memory out of range (2 048 octets)	SYS	Error
0x180A	Backup inconsistency – identity fault	SYS	Error
0x180B	Backup inconsistency – Data Storage unspecific error	SYS	Error
0x180C	Backup inconsistency – upload fault	SYS	Error
0x180D	Parameter inconsistency – download fault	SYS	Error
0x180E	P24 (Class B) missing or undervoltage	APP	Error
0x180F	Short circuit at P24 (Class B) – check wire connection (e.g. L2+)	APP	Error
0x1810	Short circuit at I/Q – check wiring	APP	Error
0x1811	Short circuit at C/Q (if digital output) – check wiring	APP	Error
0x1812	Overcurrent at I/Q – check load	APP	Error
0x1813	Overcurrent at C/Q (if digital output) – check load	APP	Error
0x1814 to 0x1EFF	Reserved		
0x1F00 to 0x1FFF	Vendor specific		–
0x2000 to 0x2FFF	Safety extensions		See [10]
0x3000 to 0x3FFF	Wireless extensions		See [11]
0x4000 to 0x5FFF	Reserved		
0x6000	Invalid cycle time	SYS	Error
0x6001	Revision fault – incompatible protocol version	SYS	Error
0x6002	ISDU batch failed – parameter inconsistency?	SYS	Error
0x6003 to 0xFF20	Reserved		
0xFF21 [a]	DL: Device plugged in ("NEW_SLAVE") – PD stop		Notification
0xFF22 [a]	Device communication lost ("DEV_COM_LOST")		Notification
0xFF23 [a]	Data Storage identification mismatch ("DS_IDENT_MISMATCH")		Notification
0xFF24 [a]	Data Storage buffer overflow ("DS_BUFFER_OVERFLOW")		Notification
0xFF25 [a]	Data Storage parameter access denied ("DS_ACCESS_DENIED")		Notification
0xFF26 [b]	Port status changed – Use "SMI_PortStatus" service for Port status in detail. Each change of "PortStatusInfo" causes this Event via SMI_PortEvent	SYS	Notification
0xFF27 [b]	Data Storage upload completed and new data object available. Each completion of a Data Storage upload causes this Event via SMI_PortEvent	SYS	Notification
0xFF28 to 0xFF30	Reserved		
0xFF31 [a]	DL: Incorrect Event signalling ("EVENT")		Notification
0xFF32 to 0xFFFF	Reserved		

[a] No more required due to SMI Event concept. Not recommended for implementations.

[b] These Events are optional.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex E
(normative)

Coding of ArgBlocks

E.1 General

The purpose of ArgBlocks is explained in 11.2.2. Each ArgBlock is uniquely defined by its ArgBlock identifier (ArgBlockID) and its ArgBlock length (ArgBlockLength). Extension of ArgBlocks by just using a larger ArgBlock length is not permitted. Manufacturer specific ArgBlocks are possible by using the service groups B to E (see Figure E.1).

Transmission of ArgBlocks follows the convention in 3.3.6 as octet stream beginning with octet offset 0.

The four-nibble structure of the ArgBlockID is shown in Figure E.1. The ArgBlockID "0x0000" is reserved. The fourth nibble (N4) is assigned to SMI service groups. The third nibble (N3) is assigned to domains and to SMI management. Nibble 1 (N1) and nibble 2 (N2) define ArgBlocks within the particular domain.

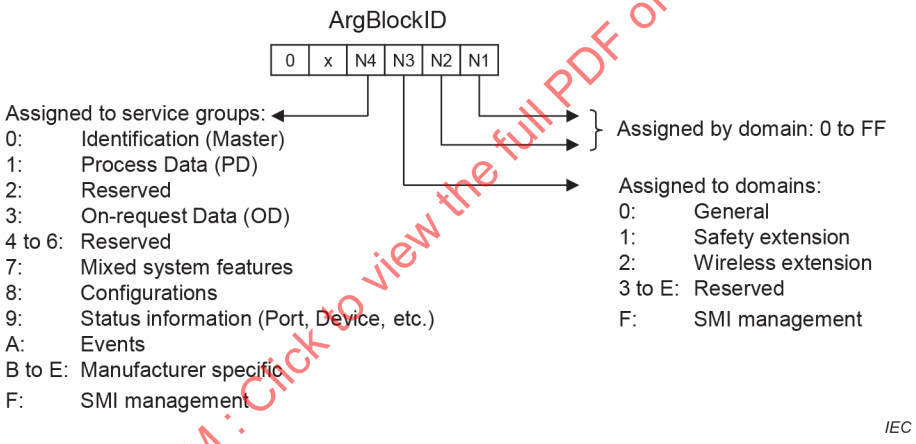


Figure E.1 – Assignment of ArgBlock identifiers

Table E.1 shows all defined ArgBlock types and their IDs including those for system extensions in order to avoid ambiguities. ArgBlockIDs are assigned by the IO-Link Community.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table E.1 – ArgBlock types and their ArgBlockIDs

ArgBlock type	ArgBlockID	Definition	Used by SMI_xxx services
MasterIdent	0x0001	Clause E.2	SMI_MasterIdentification (see 11.2.4)
FSMasterAccess	0x0100	[10]	–
WMasterConfig	0x0200	[11]	–
PDIn	0x1001	Clause E.10	SMI_PDIn (see 11.2.17)
PDOut	0x1002	Clause E.11	SMI_PDOut (see 11.2.18)
PDInOut	0x1003	Clause E.12	SMI_PDInOut (see 11.2.19)
SPDUIn	0x1101	[10]	–
SPDUOut	0x1102	[10]	–
PDInIQ	0x1FFE	Clause E.13	SMI_PDInIQ (see 11.2.20)
PDOutIQ	0x1FFF	Clause E.14	SMI_PDOutIQ (see 11.2.21) SMI_PDReadbackOutIQ (see 11.2.22)
On-requestData	0x3000	Clause E.5	SMI_DeviceWrite (see 11.2.10)
	0x3001		SMI_DeviceRead (see 11.2.11)
DS_Data	0x7000	Clause E.6	SMI_DSToParServ (see 11.2.8) SMI_ParServToDS (see 11.2.9)
DeviceParBatch	0x7001	Clause E.7	SMI_ParamWriteBatch (see 11.2.12) SMI_ParamReadBatch (see 11.2.13)
IndexList	0x7002	Clause E.8	SMI_ParamReadBatch (see 11.2.13)
PortPowerOffOn	0x7003	Clause E.9	SMI_PortPowerOffOn (see 11.2.14)
PortConfigList	0x8000	Clause E.3	SMI_PortConfiguration (see 11.2.5) SMI_ReadBackPortConfiguration (see 11.2.6)
FSPortConfigList	0x8100	[10]	–
WTrackConfigList	0x8200	[11]	–
PortStatusList	0x9000	Clause E.4	SMI_PortStatus (see 11.2.7)
FSPortStatusList	0x9100	[10]	–
WTrackStatusList	0x9200	[11]	–
WTrackScanResult	0x9201	[11]	–
DeviceEvent	0xA000	Clause E.15	SMI_DeviceEvent (see 11.2.15)
PortEvent	0xA001	Clause E.16	SMI_PortEvent (11.2.16)
VoidBlock	0xFFF0	Clause E.17	SMI service management
JobError	0xFFFF	Clause E.18	SMI service management

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

E.2 MasterIdent

This ArgBlock is used by the service SMI_MasterIdentification (see 11.2.4). Table E.2 shows the coding of the MasterIdent ArgBlock.

Table E.2 – MasterIdent

Octet Off-set	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x0001
2	VendorID	Unique VendorID of the Master (see B.1.8)	Unsigned16	1 to 0xFFFF
4	MasterID	4 octets long vendor specific unique identification of the Master	Unsigned32	1 to 0xFFFFFFFF
8	MasterType	0: Unspecific (manufacturer specific) 1: Reserved 2: Master acc. to this document or later 3: FS_Master; see [10] 4: W_Master; see [11] 5 to 255: Reserved	Unsigned8	0 to 0xFF
9	Features_1	7 6 5 4 3 2 1 0 Bit 0: DeviceParBatch (SMI_ParamWriteBatch) 0 = not supported 1 = supported Bit 1: DeviceParBatch (SMI_ParamReadBatch) 0 = not supported 1 = supported Bit 2: PortPowerOffOn (SMI_PortPowerOffOn) 0 = not supported 1 = supported Bit 3 to 7: Reserved (= 0)	Unsigned8	0 to 0xFF
10	Features_2	7 6 5 4 3 2 1 0 Reserved for future use (= 0)	Unsigned8	0 to 0xFF
11	MaxNumberOfPorts	Maximum number (n) of Ports of this Master	Unsigned8	1 to 0xFF
12	PortTypes	Array indicating for all n Ports the type of Port 0: Class A 1: Class A with PortPowerOffOn 2: Class B; see 5.4.2 3: FS_Port_A without OSSDe; see [10] 4: FS_Port_A with OSSDe; see [10] 5: FS_Port_B; see [10] 6: W_Master; see [11] 7 to 255: Reserved	Array [1 to n] of Unsigned8	1 to 6

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

E.3 PortConfigList

This ArgBlock is used by the services SMI_PortConfiguration (see 11.2.5) and SMI_Readback-PortConfiguration (see 11.2.6). Table E.3 shows the coding of the PortConfigList ArgBlock.

Table E.3 – PortConfigList

Octet Off-set	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x8000
2	PortMode	This element contains the Port mode expected by the SMI client, e.g. gateway application. All modes are mandatory. They shall be mapped to the Target Modes of "SM_SetPortConfig" (see 9.2.2.2). 0: DEACTIVATED (SM: INACTIVE → Port is deactivated; input and output Process Data are "0"; Master shall not perform activities at this Port) 1: IOL_MANUAL (SM: CFGCOM → Target Mode based on user defined configuration including validation of RID, VID, DID) 2: IOL_AUTOSTART [a] (SM: AUTOCOM →Target Mode w/o configuration and w/o validation of VID/DID; RID gets highest revision the Master is supporting; Validation: NO_CHECK) 3: DI_C/Q (Pin 4 at M12) [b] (SM: DI → Port in input mode SIO) 4: DO_C/Q (Pin 4 at M12) [b] (SM: DO → Port in output mode SIO) 5 to 48: Reserved for future versions 49 to 96: Reserved for extensions (see [10], [11]) 97 to 255: Manufacturer specific	Unsigned8	0 to 0xFF
3	Validation&Backup	This element contains the InspectionLevel to be performed by the Device and the Backup/Restore behavior. 0: No Device check 1: Type compatible Device V1.0 2: Type compatible Device V1.1 3: Type compatible Device V1.1, Backup + Restore 4: Type compatible Device V1.1, Restore 5 to 255: Reserved	Unsigned8	0 to 0xFF
4	I/Q behavior (manufacturer or profile specific, see [10], [11])	This element defines the behavior of the I/Q signal (Pin 2 at M12 connector) 0: Not supported 1: Digital Input 2: Digital Output 3: Reserved 4: Reserved 5: Power 2 (Port Class B) 6 to 255: Reserved	Unsigned8	0 to 0xFF

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Octet Off-set	Element name	Definition	Data type	Values
5	PortCycleTime	This element contains the Port cycle time expected by the SMI client. AFAP is default. They shall be mapped to the ConfiguredCycleTime of "SM_SetPortConfig" (see 9.2.2.2) 0: AFAP (As fast as possible – SM: FreeRunning → Port cycle timing is not restricted. Default value in Port mode IOL_MANUAL) 1 to 255: TIME (SM: For coding see Table B.3. Device shall achieve the indicated Port cycle time. An error shall be created if this value is below MinCycleTime of the Device or in case of other misfits)	Unsigned8	0 to 0xFF
6	VendorID	This element contains the 2 octets long VendorID expected by the SMI client (see B.1.8)	Unsigned16	1 to 0xFFFF
8	DeviceID	This element contains the 3 octets long DeviceID expected by the SMI client (see B.1.9)	Unsigned32	1 to 0xFFFFFF
[a] In PortMode "IOL_Autostart" parameters VendorID, DeviceID, and Validation&Backup are treated "don't care". [b] In PortModes "DI_C/Q" and "DO_C/Q" all parameters are treated "don't care".				

E.4 PortStatusList

This ArgBlock is used by the service SMI_PortStatus (see 11.2.7). Table E.4 shows the coding of the ArgBlock "PortStatusList". It refers to the state machine of the Configuration Manager in Figure 101 and shows its current states.

Content of "PortStatusInfo" is derived from "PortMode" in 9.2.2.4. Values not available shall be set to "0".

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table E.4 – PortStatusList

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x9000
2	PortStatusInfo	This element contains status information of the Port. 0: NO_DEVICE No communication (COMLOST). However, Port configuration IOL_MANUAL or IOL_AUTOSTART was set (see Table E.3). 1: DEACTIVATED Port configuration DEACTIVATED was set (see Table E.3). 2: PORT_DIAG This value to be set if a DiagEntry indicates an upcoming diagnosis of the Port during startup, validation, and Data Storage (group error). Device is in PREOPERATE and DiagEntry contains the diagnosis cause. 3: Reserved 4: OPERATE This value to be set if the Device is in OPERATE, even in case of Device error. 5: DI_C/Q Port configuration "DI" was set (see Table E.3). 6: DO_C/Q Port configuration "DO" was set (see Table E.3). 7 to 8: Reserved for SDCI-FS [10] 9 to 253: Reserved 254: PORT_POWER_OFF Shutdown of Port is active caused by SMI_PortPowerOffOn 255: NOT_AVAILABLE PortStatusInfo currently not available	Unsigned8 (enum)	0 to 0xFF
3	PortQualityInfo	This element contains status information on Process Data (see 8.2.2.12). Bit0: 0 = VALID 1 = INVALID Bit1: 0 = PDOUTVALID 1 = PDOUTINVALID Bit2 to Bit7: Reserved	Unsigned8	–
4	RevisionID	This element contains information of the SDCI protocol revision of the Device (see B.1.5) 0: NOT_DETECTED (No communication at that Port) <>0: Copied from Direct parameter page, address 4	Unsigned8	0 to 0xFF

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Octet Offset	Element name	Definition	Data type	Values
5	TransmissionRate	This element contains information on the effective Port transmission rate. 0: NOT_DETECTED (No communication at that Port) 1: COM1 (transmission rate 4,8 kbit/s) 2: COM2 (transmission rate 38,4 kbit/s) 3: COM3 (transmission rate 230,4 kbit/s) 4 to 255: Reserved for future use	Unsigned8	0 to 0xFF
6	MasterCycleTime	This element contains information on the Master cycle time. For coding see B.1.3.	Unsigned8	–
7	InputDataLength	This element contains the input data length as number of octets of the Device provided by the PDIn service (see Clause E.10)	Unsigned8	0 to 0x20
8	OutputDataLength	This element contains the output data length as number of octets for the Device accepted by the PDOut service (see Clause E.11	Unsigned8	0 to 0x20
9	VendorID	This element contains the 2 octets long VendorID expected by the SMI client	Unsigned16	1 to 0xFFFF
11	DeviceID	This element contains the 3 octets long DeviceID expected by the SMI client	Unsigned32	1 to 0xFFFFFF
15	NumberOfDiags	This element contains the number x of diagnosis entries (DiagEntry 0 to DiagEntry x)	Unsigned8	0 to 0xFF
16	DiagEntry0	This element contains the "EventQualifier" and "EventCode" of a diagnosis (Event). See B.2.21 for coding and how to deal with "Event appears/disappears".	Struct Unsigned8/16	–
19	DiagEntry1	Further entries up to x if applicable…	…	–

E.5 On-request_Data

This ArgBlock with ArgBlockID 0x3000 is used by the service SMI_DeviceWrite (see 11.2.10) and with ArgBlockID 0x3001 (Index only) by the service SMI_DeviceRead (see 11.2.11). Table E.5 shows the coding of the ArgBlockType "On-request_Data".

Table E.5 – On-request_Data

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x3000 (Write) 0x3001 (Read)
2	Index	This element contains the Index to be used for the AL_Write or AL_Read service	Unsigned16	0 to 0xFFFF
4	Subindex	This element contains the Subindex to be used for the AL_Write or AL_Read service	Unsigned8	0 to 0xFF
5 to n	On-request Data	This element contains the On-request Data for ArgBlock 0x3000 if available.	Octet string	–

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

E.6 DS_Data

This ArgBlock is used by the services SMI_DSToParServ (see 11.2.8) and SMI_ParServToDS (see 11.2.9). Table E.6 shows the coding of the ArgBlockType "DS_Data".

Table E.6 – DS_Data

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x7000
2 to n	DataStorageObject	This element contains the Device parameter set coded according to 11.4.2 (Table G.2 followed by Table G.1)	Record (octet string)	1 to 2×2^{10} +12

E.7 DeviceParBatch

This ArgBlock provides means to transfer a large number of Device parameters via a number of ISDU write or read requests to the Device. It is used by the services SMI_ParamWriteBatch (see 11.2.12) or SMI_ParamReadBatch (see 11.2.13). Table E.7 shows the coding of the ArgBlockType "DeviceParBatch".

Table E.7 – DeviceParBatch

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x7001
2	Object1_Index	Index of 1st parameter	Unsigned16	0 to 0xFFFF
4	Object1_Subindex	Subindex of 1st parameter	Unsigned8	0 to 0xFF
5	Object1_Length	Length of parameter record or	Unsigned8	0 to 0xE8
		ISDU error (implicitly 2 octets)	Unsigned8	0xFF (error)
6	Object1_Data	Parameter record or	Record	0 to r
		ISDU ErrorType (return value)	Unsigned16	ErrorType
6+r	Object2_Index	Index of 2nd parameter	Unsigned16	0 to 0xFFFF
6+r+2	Object2_Subindex	Subindex of 2nd parameter	Unsigned8	0 to 0xFF
6+r+3	Object2_Length	Length of parameter record or	Unsigned8	0 to 0xE8
		ISDU error (implicitly 2 octets)	Unsigned8	0xFF (error)
6+r+4	Object2_Data	Parameter record or	Record	0 to s
		ISDU ErrorType (return value)	Unsigned16	ErrorType
…		…		
…	Object$x$_Index	Index of x^{th} parameter	Unsigned16	0 to 0xFFFF
…	Object$x$_Subindex	Subindex of x^{th} parameter	Unsigned8	0 to 0xFF
…	Object$x$_Length	Length of parameter record or	Unsigned8	0 to 0xE8
		ISDU error (implicitly 2 octets)	Unsigned8	0xFF (error)
…	Object$x$_Data	Parameter record or	Record	0 to t
		ISDU ErrorType (return value)	Unsigned16	ErrorType
In case of SMI_ParamWriteBatch, this ArgBlock will return ErrorType "0x0000" for each successfully written object				

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

E.8 IndexList

This ArgBlock provides a list of the Indices of several requested Device parameters to be retrieved from a Device via the service SMI_ParamReadBatch (see 11.2.13). Table E.8 shows the coding of the ArgBlockType "IndexList".

Table E.8 – IndexList

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x7002
2	Object1_Index	Index of 1st object	Unsigned16	0 to 0xFFFF
4	Object1_Subindex	Subindex of 1st object	Unsigned8	0 to 0xFF
5	Object2_Index	Index of 2nd object	Unsigned16	0 to 0xFFFF
7	Object2_Subindex	Subindex of 2nd object	Unsigned8	0 to 0xFF
8	Object3_Index	Index of 3rd object	Unsigned16	0 to 0xFFFF
10	Object3_Subindex	Subindex of 3rd object	Unsigned8	0 to 0xFF
…				

E.9 PortPowerOffOn

Table E.9 shows the ArgBlockType "PortPowerOffOn". The service "SMI_PortPowerOffOn" (see 11.2.14) together with this ArgBlock can be used for energy saving purposes during production stops or alike. Minimum PowerOffTime shall be 500 ms.

Table E.9 – PortPowerOffOn

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x7003
2	PortPowerMode	0: One time switch off (PowerOffTime) 1: Switch PortPowerOff (permanent) 2: Switch PortPowerOn (permanent)	Unsigned8	–
3	PowerOffTime	Duration of Master Port power off (ms). See also [10].	Unsigned16	0x01F4 to 0xFFFF

E.10 PDIn

This ArgBlock provides means to retrieve input Process Data from the InBuffer within the Master. It is used by the service SMI_PDIn (see 11.2.17). Table E.10 shows the coding of the "PDIn" ArgBlockType.

Mapping principles of input Process Data (PD) are specified in 11.7.2. The following rules apply for the ArgBlock PDIn:

– the first 2 octets are occupied by the ArgBlockID (0x1001),

– the third octet (offset = 2) carries the Port Qualifier Information (PQI),

– the fourth octet specifies the length of input Process Data (cyclic values or the DI bit on the C/Q line),

– subsequent octets are occupied by the input Process Data of the Device.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table E.10 – PDIn

Octet offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x1001
2	PQI	Port Qualifier Information	Unsigned8	–
3	InputDataLength	This element contains the length of the Device's input Process Data contained in the following elements.	Unsigned8	0 to 0x20
4	PDI0	Input Process Data (octet 0)	Unsigned8	0 to 0xFF
5	PDI1	Input Process Data (octet 1)	Unsigned8	0 to 0xFF
…				
InputDataLength + 4	PDIn	Input Process Data (octet n)	Unsigned8	0 to 0xFF

E.11 PDOut

This ArgBlock provides means to transfer output Process Data to the OutBuffer within the Master. It is used by the service SMI_PDOut (see 11.2.18). Table E.11 shows coding of the "PDOut" ArgBlockType.

Mapping principles of output Process Data (PD) are specified in 11.7.3. The following rules apply for the ArgBlock PDOut:

– the first 2 octets are occupied by the ArgBlockID (0x1002),
– the third octet (offset = 2) carries the Port qualifier (OE),
– the fourth octet specifies the length of output Process Data (cyclic values or the DO bit on the C/Q line),
– subsequent octets are occupied by the output Process Data, which are propagated to the Device.

Table E.11 – PDOut

Octet offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x1002
2	OE	Output Enable	Unsigned8	–
3	OutputDataLength	This element contains the length of the output Process Data for the Device contained in the following elements.	Unsigned8	0 to 0x20
4	PDO0	Output Process Data (octet 0)	Unsigned8	0 to 0xFF
5	PDO1	Output Process Data (octet 1)	Unsigned8	0 to 0xFF
…				
OutputDataLength + 4	PDOm	Output Process Data (octet m)	Unsigned8	0 to 0xFF

E.12 PDInOut

This ArgBlock provides means to retrieve input Process Data from the InBuffer and output Process Data from the OutBuffer within the Master. It is used by the service SMI_PDInOut (see 11.2.19). Table E.12 shows the coding of the "PDInOut" ArgBlockType using mapping principles of Clause E.10 and Clause E.11.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table E.12 – PDInOut

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x1003
2	PQI	Port Qualifier Information	Unsigned8	–
3	OE	Output Enable	Unsigned8	–
4	InputDataLength	This element contains the length of the Device's input Process Data contained in the following elements.	Unsigned8	0 to 0x20
5	PDI0	Input Process Data (octet 0)	Unsigned8	0 to 0xFF
6	PDI1	Input Process Data (octet 1)	Unsigned8	0 to 0xFF
…				
InputDataLength + 5	PDIn	Input Process Data (octet n)	Unsigned8	0 to 0xFF
InputDataLength + 6	OutputDataLength	This element contains the length of the output Process Data for the Device contained in the following elements.	Unsigned8	0 to 0x20
InputDataLength + 6	PDO0	Output Process Data (octet 0)	Unsigned8	0 to 0xFF
InputDataLength + 7	PDO1	Output Process Data (octet 1)	Unsigned8	0 to 0xFF
…				
InputDataLength + OutputDataLength + 6	PDOm	Output Process Data (octet m)	Unsigned8	0 to 0xFF

E.13 PDInIQ

This ArgBlock provides means to retrieve input Process Data (I/Q signal) from the InBuffer within the Master. It is used by the service SMI_PDInIQ (see 11.2.20). Table E.13 shows the coding of the "PDInIQ" ArgBlockType.

Mapping principles of input Process Data (PD) are specified in 11.7.2. The following rules apply for the ArgBlock PDInIQ:

– the first 2 octets are occupied by the ArgBlockID (0x1FFE),
– the subsequent octet is occupied by the input Process Data of the signal line,
– padding (unused) bits shall be filled with "0".

Table E.13 – PDInIQ

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x1FFE
2	PDI0	Input Process Data I/Q signal (octet 0)	Unsigned8	0 to 0x01

E.14 PDOutIQ

This ArgBlock provides means to transfer output Process Data (I/Q signal) to the OutBuffer within the Master. It is used by the services SMI_PDOutIQ (see 11.2.21) and SMI_PDReadbackOutIQ (see 11.2.22). Table E.14 shows the coding of the "PDOutIQ" ArgBlockType.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Mapping principles of output Process Data (PD) are specified in 11.7.3. The following rules apply for the ArgBlock PDOutIQ:

– the first 2 octets are occupied by the ArgBlockID (0x1FFF),
– the subsequent octet is occupied by the output Process Data that is propagated to the signal line,
– padding (unused) bits shall be filled with "0".

Table E.14 – PDOutIQ

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x1FFF
2	PDO0	Output Process Data I/Q signal (octet 0)	Unsigned8	0 to 0x01

E.15 DeviceEvent

This ArgBlock is used by the services SMI_DeviceEvent (see 11.2.15). Table E.15 shows the coding of the ArgBlockType "DeviceEvent".

Table E.15 – DeviceEvent

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0xA000
2	EventQualifier	EventQualifier according to A.6.4	Unsigned8	0 to 0xFF
3,4	EventCode	EventCode according to Table D.1	Unsigned16	0 to 0xFFFF

E.16 PortEvent

This ArgBlock is used by the services SMI_PortEvent (see 11.2.16). Table E.16 shows the coding of the ArgBlockType "PortEvent".

Table E.16 – PortEvent

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0xA001
2	EventQualifier	EventQualifier according to A.6.4	Unsigned8	0 to 0xFF
3,4	EventCode	EventCode according to Table D.2	Unsigned16	0 to 0xFFFF

E.17 VoidBlock

This ArgBlock is used in SMI services to indicate read requests within the argument. Table E.17 shows the coding of the ArgBlockType "VoidBlock".

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table E.17 – VoidBlock

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0xFFF0

E.18 JobError

This ArgBlock is used in SMI services to indicate negative acknowledgments "Result (-)" together with an ErrorType according to Table C.3. Table E.18 shows the coding of the ArgBlockType "JobError".

Table E.18 – JobError

Octet Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0xFFFF
2	ExpArgBlockID	Expected ArgBlockID of the service result	Unsigned16	0x0001 to 0xFFFF
4	ErrorCode	SMI service related ErrorType or propagated Device/Master error (upper value)	Unsigned8	Table C.3
5	AdditionalCode	SMI service related ErrorType or propagated Device/Master error (lower value)	Unsigned8	

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex F
(normative)

Data types

F.1 General

This annex specifies basic and composite data types. Examples demonstrate the structures and the transmission aspects of data types for singular use or in a packed manner.

NOTE More examples are available in [6].

F.2 Basic data types

F.2.1 General

The coding of basic data types is shown only for singular use, which is characterized by

– Process Data consisting of one basic data type,
– Parameter consisting of one basic data type,
– Subindex (> 0) access on individual data items of parameters of composite data types (arrays, records).

F.2.2 BooleanT

A BooleanT represents a data type that can have only two different values i.e. TRUE and FALSE. The data type is specified in Table F.1. For singular use the coding is shown in Table F.2. A sender shall always use 0xFF for 'TRUE' or 0x00 for 'FALSE'. Since some upper-level software tools are not used to this restricted use of Booleans, a receiver can interpret the range from 0x01 through 0xFE for 'TRUE' or reject with an error message. The packed form is demonstrated in Table F.22 and Figure F.9.

Table F.1 – BooleanT

Data type name	Value range	Resolution	Length
BooleanT	TRUE/FALSE	-	1 bit or 1 octet

Table F.2 – BooleanT coding

Bit	7	6	5	4	3	2	1	0	Values
TRUE	1	1	1	1	1	1	1	1	0xFF
FALSE	0	0	0	0	0	0	0	0	0x00

F.2.3 UIntegerT

A UIntegerT represents an unsigned number depicted by 2 up to 64 bits ("enumerated"). The number is accommodated and right aligned within the following permitted octet containers: 1, 2, 4, or 8. High order padding bits are filled with "0". Coding examples are shown in Figure F.1 and Figure F.2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

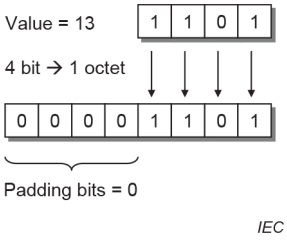


Figure F.1 – Coding example of small UIntegerT

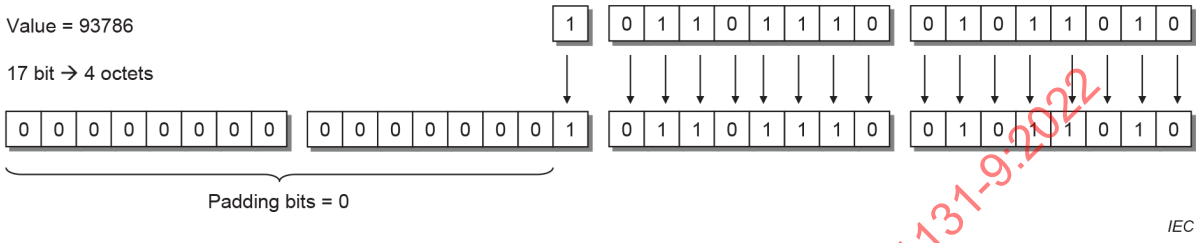


Figure F.2 – Coding example of large UIntegerT

The data type UIntegerT is specified in Table F.3 for singular use.

Table F.3 – UIntegerT

Data type name	Value range	Resolution	Length
UIntegerT	$0 \ldots 2^{\text{bitlength}} - 1$	1	1 octet, or 2 octets, or 4 octets, or 8 octets
NOTE 1 High order padding bits are filled with "0". NOTE 2 Most significant octet (MSO) sent first.			

F.2.4 IntegerT

An IntegerT represents a signed number depicted by 2 up to 64 bits. The number is accommodated within the following permitted octet containers: 1, 2, 4, or 8 and right aligned and extended correctly signed to the chosen number of bits. The data type is specified in Table F.4 for singular use. SN represents the sign with "0" for all positive numbers and zero, and "1" for all negative numbers. Padding bits are filled with the content of the sign bit (SN).

Table F.4 – IntegerT

Data type name	Value range	Resolution	Length
IntegerT	$-2^{\text{bitlength}-1} \ldots 2^{\text{bitlength}-1} - 1$	1	1 octet, or 2 octets, or 4 octets, or 8 octets
NOTE 1 High order padding bits are filled with the value of the sign bit (SN). NOTE 2 Most significant octet (MSO) sent first (lowest respective octet number in Table F.5).			

The four coding possibilities in containers are listed in Table F.5 through Table F.8.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table F.5 – IntegerT coding (8 octets)

Bit	7	6	5	4	3	2	1	0	Container
Octet 1	SN	2^{62}	2^{61}	2^{60}	2^{59}	2^{58}	2^{57}	2^{56}	8 octets
Octet 2	2^{55}	2^{54}	2^{53}	2^{52}	2^{51}	2^{50}	2^{49}	2^{48}	
Octet 3	2^{47}	2^{46}	2^{45}	2^{44}	2^{43}	2^{42}	2^{41}	2^{40}	
Octet 4	2^{39}	2^{38}	2^{37}	2^{36}	2^{35}	2^{34}	2^{33}	2^{32}	
Octet 5	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	
Octet 6	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
Octet 7	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	
Octet 8	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	

Table F.6 – IntegerT coding (4 octets)

Bit	7	6	5	4	3	2	1	0	Container
Octet 1	SN	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	4 octets
Octet 2	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
Octet 3	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	
Octet 4	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	

Table F.7 – IntegerT coding (2 octets)

Bit	7	6	5	4	3	2	1	0	Container
Octet 1	SN	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2 octets
Octet 2	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	

Table F.8 – IntegerT coding (1 octet)

Bit	7	6	5	4	3	2	1	0	Container
Octet 1	SN	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	1 octet

Coding examples within containers are shown in Figure F.3

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

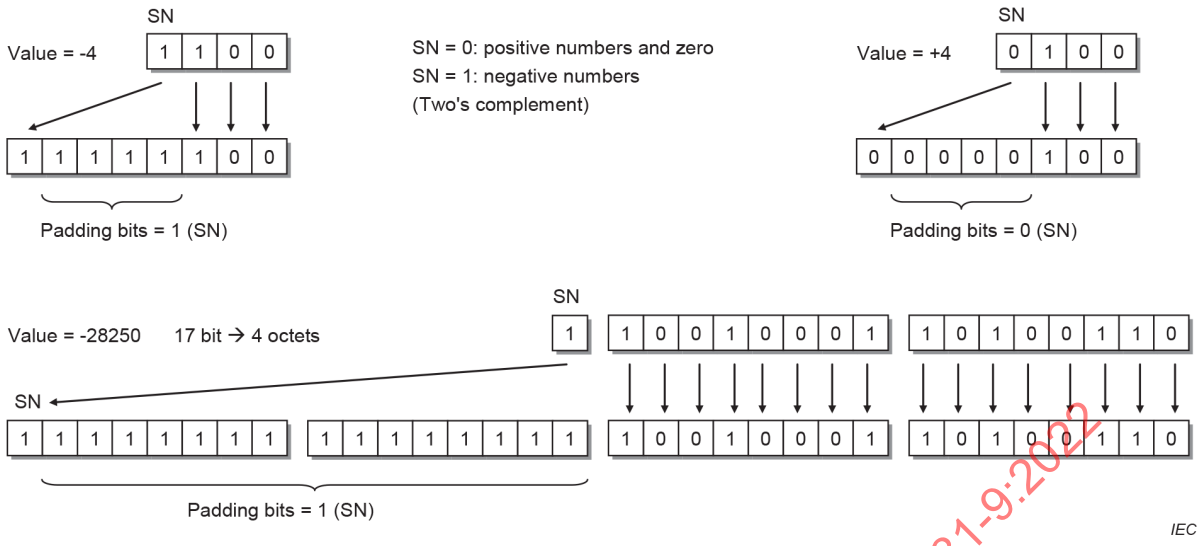


Figure F.3 – Coding examples of IntegerT

F.2.5 Float32T

A Float32T represents a number specified by IEEE Std 754 [21] as single precision (32 bit). Table F.9 gives the definition and Table F.10 the coding. SN represents the sign with "0" for all positive numbers and zero, and "1" for all negative numbers.

Table F.9 – Float32T

Data type name	Value range	Resolution	Length
Float32T	See IEEE Std 754 [21]	See IEEE Std 754 [21]	4 octets

Table F.10 – Coding of Float32T

Bits	7	6	5	4	3	2	1	0
Octet 1	SN	Exponent (E)						
	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1
Octet 2	(E)	Fraction (F)						
	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}
Octet 3	Fraction (F)							
	2^{-8}	2^{-9}	2^{-10}	2^{-11}	2^{-12}	2^{-13}	2^{-14}	2^{-15}
Octet 4	Fraction (F)							
	2^{-16}	2^{-17}	2^{-18}	2^{-19}	2^{-20}	2^{-21}	2^{-22}	2^{-23}

In order to realize negative exponent values a special exponent encoding mechanism is set in place as follows:

The Float32T exponent (E) is encoded using an offset binary representation, with the zero offset being 127, also known as exponent bias in [21].

E_{min} = 0x01 − 0x7F = −126

E_{max} = 0xFE − 0x7F = 127

Exponent bias = 0x7F = 127

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Thus, as defined by the offset binary representation, in order to get the true exponent, the offset of 127 shall be subtracted from the stored exponent.

F.2.6 StringT

A StringT represents an ordered sequence of symbols (characters) with a variable or fixed length of octets (maximum of 232 octets) coded in US-ASCII (7 bit) or UTF-8. UTF-8 uses one octet for all ASCII characters and up to 4 octets for other characters. 0x00 is not permitted as a character. Table F.11 gives the definition.

Table F.11 – StringT

Data type name	Encoding	Standards	Length [a]
StringT	US-ASCII	see ISO/IEC 646 [15]	Any length of character string with a maximum of 232 octets
	UTF-8 [b]	see ISO/IEC 10646 [17]	
[a] The length can be obtained from a Device's IODD via the attribute 'fixedLength'.			
[b] In order to ensure proper handling of client applications it is recommended not to use US-ASCII or UTF-8 codes from 0x00 to 0x1F and 0xFF.			

An instance of StringT can be shorter than defined by the IODD attribute 'fixedLength'. 0x00 shall be used for the padding of unused octets.

A condensed form can be used for optimization, where the character string is transmitted in its actual length and the padding octets are omitted. The receiver can deduce the original length from the length of the ISDU or by searching the first NULL (0x00) character (see A.5.2 and A.5.3). This condensed form can be used in case of singular access (see Figure F.4).

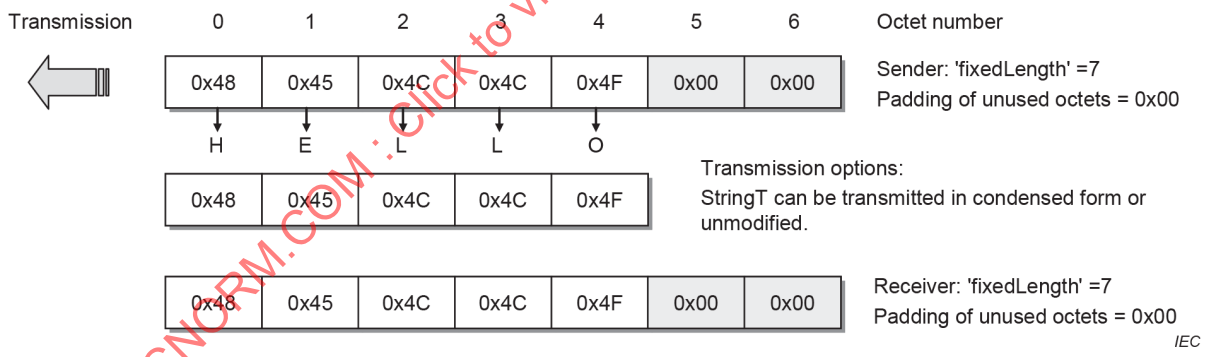


Figure F.4 – Singular access of StringT

F.2.7 OctetStringT

An OctetStringT represents an ordered sequence of octets with a fixed length (maximum of 232 octets). Table F.12 gives the definition and Figure F.5 a coding example for a fixed length of 7.

Table F.12 – OctetStringT

Data type name	Value range	Standards	Length
OctetStringT	0x00 … 0xFF per octet	–	Fixed length with a maximum of 232 octets
NOTE The length can be obtained from a Device's IODD via the attribute 'fixedLength'.			

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

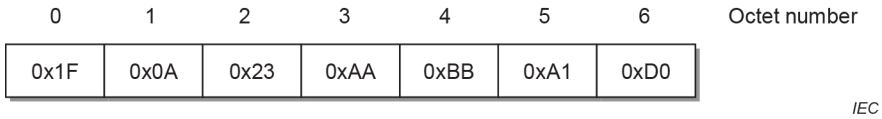


Figure F.5 – Coding example of OctetStringT

F.2.8 TimeT

A TimeT is based on the RFC 1305 [22] standard and composed of two unsigned values that express the network time related to a particular date. Its semantic has changed from RFC 1305 according to Figure F.6. Table F.13 gives the definition and Table F.14 the coding of TimeT.

The first element is a 32-bit unsigned integer data type that provides the network time in seconds since 1900-01-01 0.00,00 (UTC) or since 2036-02-07 6.28,16 (UTC) for time values less than 0x9DFF4400, which represents the 1984-01-01 0:00,00 (UTC). The second element is a 32-bit unsigned integer data type that provides the fractional portion of seconds in $1/2^{32}$ s. Rollovers after 136 years are not automatically detectable and shall be maintained by the application.

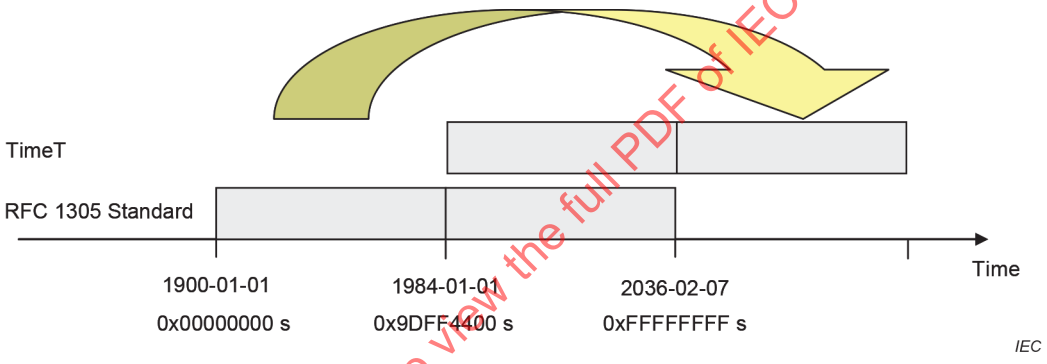


Figure F.6 – Definition of TimeT

Table F.13 – TimeT

Data type name	Value range	Resolution	Length
TimeT	Octet 1 to 4 (see Table F.14): $0 \leq i \leq (2^{32} - 1)$	s (seconds)	8 octets (32 bit unsigned integer + 32 bit unsigned integer)
	Octet 5 to 8 (see Table F.14): $0 \leq i \leq (2^{32} - 1)$	$(1/2^{32})$ s	
NOTE 32-bit unsigned integer are normal computer science data types.			

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table F.14 – Coding of TimeT

Bit	7	6	5	4	3	2	1	0	Definitions
Octet 1	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	Seconds since 1900-01-01 0.00,00 or since 2036-02-07 6.28,16 when time value less than 0x9DFF4400.00000000
Octet 2	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
Octet 3	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	
Octet 4	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	
Octet 5	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	Fractional part of seconds. One unit is $1/(2^{32})$ s
Octet 6	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
Octet 7	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	
Octet 8	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	
	msb							lsb	msb = most significant bit lsb = least significant bit

F.2.9 TimeSpanT

A TimeSpanT is a 64-bit integer value i.e. a two's complement binary number with a length of eight octets, providing the network time difference in fractional portion of seconds in $1/2^{32}$ seconds. Table F.15 gives the definition and Table F.16 the coding of TimeSpanT.

Table F.15 – TimeSpanT

Data type name	Value range	Resolution	Length
TimeSpanT	Octet 1 to 8 (see Table F.16): $-2^{63} \leq i \leq (2^{63}-1)$	$(1/2^{32})$ s	8 octets (64-bit integer)
NOTE 64-bit integer is a normal computer science data type.			

Table F.16 – Coding of TimeSpanT

Bit	7	6	5	4	3	2	1	0	Definitions
Octet 1	2^{63}	2^{62}	2^{61}	2^{60}	2^{59}	2^{58}	2^{57}	2^{56}	Fractional part of seconds as 64-bit integer. One unit is $1/(2^{32})$ s.
Octet 2	2^{55}	2^{54}	2^{53}	2^{52}	2^{51}	2^{50}	2^{49}	2^{48}	
Octet 3	2^{47}	2^{46}	2^{45}	2^{44}	2^{43}	2^{42}	2^{41}	2^{40}	
Octet 4	2^{39}	2^{38}	2^{37}	2^{36}	2^{35}	2^{34}	2^{33}	2^{32}	
Octet 5	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	
Octet 6	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
Octet 7	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	
Octet 8	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	
	msb							lsb	msb = most significant bit lsb = least significant bit

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

F.3 Composite data types

F.3.1 General

Composite data types are combinations of basic data types only. A composite data type consists of several basic data types packed within a sequence of octets. Unused bit space shall be padded with "0".

F.3.2 ArrayT

An ArrayT addressed by an Index is a data structure with data items of the same data type. The individual data items are addressable by the Subindex. Subindex 0 addresses the whole array within the Index space. The structuring rules for arrays are given in Table F.17.

Table F.17 – Structuring rules for ArrayT

Rule number	Rule specification
1	The Subindex data items are packed in a row without gaps describing an octet sequence.
2	The highest Subindex data item *n* starts right aligned within the octet sequence.
3	UIntegerT and IntegerT with a length of ≥ 58 bit and < 64 bit are not permitted.

Table F.18 and Figure F.7 give an example for the access of an array. Its content is a set of parameters of the same basic data type.

Table F.18 – Example for the access of an ArrayT

Index	Subindex	Offset	Data items	Data Type
66	1	12	0x2	IntegerT, 'bitLength' = 3
	2	9	0x6	
	3	6	0x4	
	4	3	0x7	
	5	0	0x5	

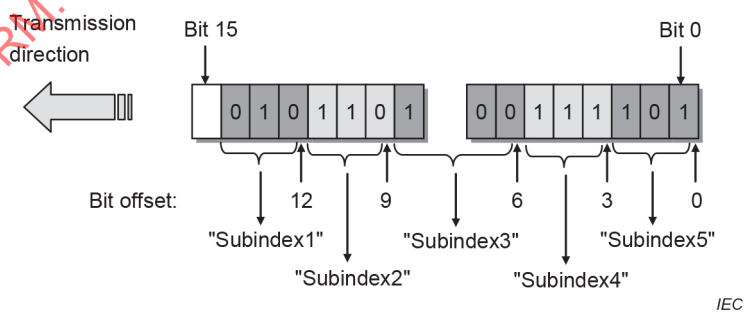


Figure F.7 – Example of an ArrayT data structure

F.3.3 RecordT

A record addressed by an Index is a data structure with data items of different data types. The Subindex allows addressing individual data items within the record on certain bit positions.

NOTE Bit positions within a RecordT can be obtained from the IODD of the particular Device.

The structuring rules for records are given in Table F.19.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table F.19 – Structuring rules for RecordT

Rule number	Rule specification
1	The Subindices within the IODD shall be listed in ascending order from 1 to *n* describing an octet sequence. Gaps within the list of Subindices are allowed.
2	Bit offsets shall always be indicated within this octet sequence (may show no strict order in the IODD).
3	The bit offset starts with the last octet within the sequence; this octet starts with offset 0 for the least significant bit and offset 7 for the most significant bit.
4	The following data types shall always be aligned on octet boundaries: Float32T, StringT, OctetStringT, TimeT, and TimeSpanT.
5	UIntegerT and IntegerT with a length of ≥ 58 bit shall always be aligned on one side of an octet boundary.
6	It is highly recommended for UIntegerT and IntegerT with a length of ≥ 8 bit to align always on one side of an octet boundary.
7	It is highly recommended for UIntegerT and IntegerT with a length of < 8 bit not to cross octet boundaries.
8	A bit position shall not be used by more than one record item.

Table F.20 gives a first example (Example 1) for the access of a RecordT. It consists of varied parameters named "Status", "Text", and "Value".

Table F.20 – Example 1 for the access of a RecordT

Index	Subindex	Offset	Data items							Data Type	Name
47	1	88	0x23	0x45						UIntegerT, 'bitLength' = 16	Status
	2	32	H	E	L	L	O	0x00	0x00	StringT, 'fixedLength' = 7	Text
	3	0	0x56	0x12	0x22	0x34				UIntegerT, 'bitLength' = 32	Value
NOTE 'bitLength' and 'fixedLength' are defined in the IODD of the particular Device.											

Table F.21 gives a second example (Example 2) for the access of a RecordT. It consists of varied parameters named "Level", "Min", and "Max". Figure F.8 shows the corresponding data structure.

Table F.21 – Example 2 for the access of a RecordT

Index	Subindex	Offset	Data items			Data Type	Name
46	1	2	0x32	0xF1		UIntegerT, 'bitLength' = 14	Level
	2	1	FALSE			BooleanT	Min
	3	0	TRUE			BooleanT	Max
NOTE 'bitLength' is defined in the IODD of the particular Device.							

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

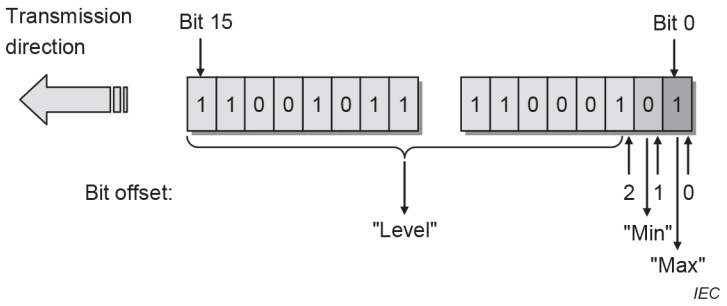


Figure F.8 – Example 2 of a RecordT structure

Table F.22 gives a third example (Example 3) for the access of a RecordT. It consists of varied parameters named "Control" through "Enable". Figure F.9 demonstrates the corresponding RecordT structure of Example 3 with the bit offsets.

Table F.22 – Example 3 for the access of a RecordT

Index	Subindex	Offset	Data items		Data Type	Name
45	1	32	TRUE		BooleanT	NewBit
	2	33	FALSE		BooleanT	DR4
	3	34	FALSE		BooleanT	CR3
	4	35	TRUE		BooleanT	CR2
	5	38	TRUE		BooleanT	Control
	6	16	0xF8	0x23	OctetStringT, 'fixedLength' = 2	Setpoint
	7	8	0x41		StringT, 'fixedLength' = 1	Unit
	8	0	0xC3		OctetStringT, 'fixedLength' = 1	Enable
NOTE 'fixedLength' is defined in the IODD of the particular Device.						

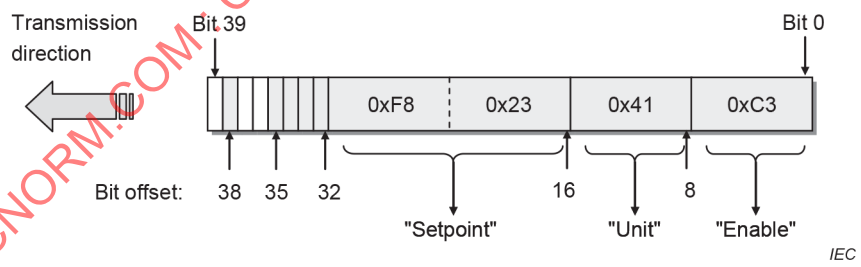


Figure F.9 – Example 3 of a RecordT structure

Figure F.10 shows a selective write request of a variable within the RecordT of Example 3 and a write request of the complete RecordT (see A.5.7).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

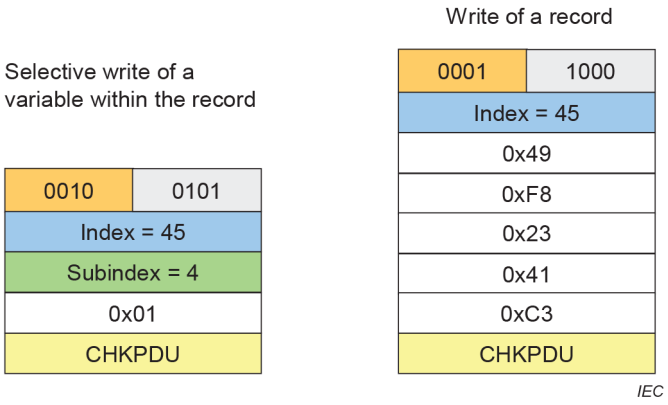


Figure F.10 – Write requests for Example 3

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex G
(normative)

Structure of the Data Storage data object

Table G.1 gives the structure of a Data Storage (DS) data object within the Master (see 11.4.2).

Table G.1 – Structure of the stored DS data object

Part	Parameter name	Definition	Data type
Object 1	ISDU_Index	ISDU Index (0 to 0xFFFF)	Unsigned16
	ISDU_Subindex	ISDU Index (0 to 0xFF)	Unsigned8
	ISDU_Length	Length of the subsequent record	Unsigned8
	ISDU_Data	Record of length ISDU_Length	Record
Object 2	ISDU_Index	ISDU Index (0 to 0xFFFF)	Unsigned16
	ISDU_Subindex	ISDU Index (0 to 0xFF)	Unsigned8
	ISDU_Length	Length of the subsequent record	Unsigned8
	ISDU_Data	Record of length ISDU_Length	Record

Object *n*	ISDU_Index	ISDU Index (0 to 0xFFFF)	Unsigned16
	ISDU_Subindex	ISDU Index (0 to 0xFF)	Unsigned8
	ISDU_Length	Length of the subsequent record	Unsigned8
	ISDU _Data	Record of length ISDU_Length	Record

The Device shall calculate the required memory size by summarizing the objects 1 to *n* (see Table B.10, Subindex 3).

The Master shall locally store in non-volatile memory the header information specified in Table G.2. See Table B.10.

Table G.2 – Associated header information for stored DS data objects

Part	Parameter name	Definition	Data type
Header	Parameter Checksum	32-bit CRC signature or revision counter (see 10.4.8)	Unsigned32
	VendorID	See B.1.8	Unsigned16
	DeviceID	See B.1.9	Unsigned32
	FunctionID	See B.1.10	Unsigned16

In case of empty Data Storage data object, the header shall be set to "0" and the ArgBlockLength shall be set to 12.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex H
(normative)

Master and Device conformity

H.1 Electromagnetic compatibility (EMC) requirements

H.1.1 General

The EMC requirements of this document are only relevant for the SDCI interface part of a particular Master or Device. The technology functions of a Device and its relevant EMC requirements are not in the scope of this document. For this purpose, the Device specific product standards shall apply. For the Master usually the EMC requirements for peripherals are specified in IEC 61131-2 or IEC 61000-6-2.

To ensure proper operating conditions of the SDCI interface, the test configurations specified in H.1.6 (Master) or H.1.7 (Device) shall be maintained during all the EMC tests. The tests required in the product standard of the equipment under test (EUT) can alternatively be performed in SIO mode.

H.1.2 Operating conditions

It is highly recommended to evaluate the SDCI during the startup phase with the cycle times given in Table H.1. In most cases, this leads to the minimal time requirements for the performance of these tests. Alternatively, the SDCI can be evaluated during normal operation of the Device, provided that the required number of M-sequences specified in Table H.1 took place during each test.

In case a test requires longer M-sequences than an M-sequence group specified in Table H.1, the error criteria shall be applied to every M-sequence group.

H.1.3 Performance criteria

a) Performance criterion A

The SDCI operating at an average cycle time as specified in Table H.1 shall not show more than six detected M-sequence errors within the number of M-sequences given in Table H.1. Multiple kinds of errors within one M-sequence shall be counted as one error. No interruption of communication is permitted.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Table H.1 – EMC test conditions for SDCI

Transmission rate	Master		Device		Maximum of M-sequence errors
	t_{CYC}	Number of M-sequences of TYPE_2_5 (read) (6 octets)	t_{CYC}	Number of M-sequences of TYPE_0 (read) (4 octets)	
4,8 kbit/s	18,0 ms	300 (6 000)	100 T_{BIT} (20,84 ms)	350 (7 000)	6
38,4 kbit/s	2,3 ms	450 (9 000)	100 T_{BIT} (2,61 ms)	500 (10 000)	6
230,4 kbit/s	0,4 ms	700 (14 000)	100 T_{BIT} (0,44 ms)	800 (16 000)	6
NOTE 1 The numbers of M-sequences are calculated according to the algorithm in Clause I.2 and rounded up. The larger number of M-sequences (in brackets) are required if a certain test (for example fast transients/burst) applies interferences only with a burst/cycle ratio (see Table H.2). NOTE 2 "Number of M-sequences" is defined as a group for the performance criteria for which the maximum number of detected errors is valid.					

b) Performance Criterion B

The error rate of criterion A shall also be satisfied after but not during the test. No change of actual operating state (e.g. permanent loss of communication) or stored data is allowed.

H.1.4 Required immunity tests

Table H.2 specifies the EMC tests to be performed.

Table H.2 – EMC test levels

Phenomena	Test Level	Performance Criterion	Constraints
Electrostatic discharges (ESD) IEC 61000-4-2	Air discharge: ±8 kV Contact discharge: ±4 kV	B	See H.1.4, a)
Radiofrequency electromagnetic field. Amplitude modulated IEC 61000-4-3	80 MHz to 1 000 MHz 10 V/m 1 400 MHz to 2 000 MHz 3 V/m 2 000 MHz to 6 000 MHz 3 V/m	A	See H.1.4, a), H.1.4, b), H.1.4, e).
Fast transients (burst) IEC 61000-4-4 (see also IEC 60947-5-2)	±1 kV	A	5 kHz or 100 kHz. The number of M-sequences in Table H.1 shall be increased by a factor of 20 due to the burst/cycle ratio 15 ms/300 ms. See H.1.4, c)
	±2 kV	B	
Surge IEC 61000-4-5	Not required for an SDCI link (SDCI link is limited to 20 m)		-
Radio-frequency common mode IEC 61000-4-6	0,15 MHz to 80 MHz 10 VEMF	A	See H.1.4, b) and H.1.4, d)
Voltage dips and interruptions IEC 61000-4-11	Not required for an SDCI link		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

The following requirements also apply as specified in Table H.2.

a) As this phenomenon influences the entire device under test, an existing device specific product standard shall take precedence over the test levels specified here.

b) The test shall be performed with a step size of 1 % and a dwell of 1 s. If a single M-sequence error occurs at a certain frequency, that frequency shall be tested until the number of M-sequences according to Table H.1 has been transmitted or until six M-sequence errors occurred.

c) Depending on the transmission rate the test time varies. The test time shall be at least one minute (with the transmitted M-sequences and the permitted errors increased accordingly).

d) This phenomenon is expected to influence most probably the EUTs internal analog signal processing and only with a very small probability the functionality of the SDCI communication. Therefore, an existing device specific product standard shall take precedence over the test levels specified here.

e) Measurement shall be performed at least for three orthogonal orientations of the Device with respect to the direction of the electromagnetic wave propagation.

H.1.5 Required emission tests

The definition of emission limits is not in the scope of this document. The requirements of the Device specific product family or generic standards apply; usually for general industrial environments this is IEC 61000-6-4.

All emission tests shall be performed at the fastest possible communication rate with the fastest cycle time.

H.1.6 Test configurations for Master

H.1.6.1 General rules

The following rules apply for the test of Masters:

– In the following test setup diagrams only the SDCI and the power supply cables are shown. All other cables shall be treated as required by the relevant product standard.

– Grounding of power supply, Master, and Devices shall be according to the relevant product standard or manual.

– Where not otherwise stated, the SDCI cable shall have an overall length of 20 m. Excess length laid as an inductive coil with a diameter of 0,3 m, where applicable mounted 0,1 m above reference ground.

– Where applicable, the auxiliary Devices shall be placed 10 cm above RefGND.

– A typical test configuration consists of the Master and two Devices, except for the RF common mode test, where only one Device shall be used.

– Each Port shall fulfill the EMC requirements.

H.1.6.2 Electrostatic discharges

Figure H.1 shows the test setup for electrostatic discharge according to IEC 61000-4-2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

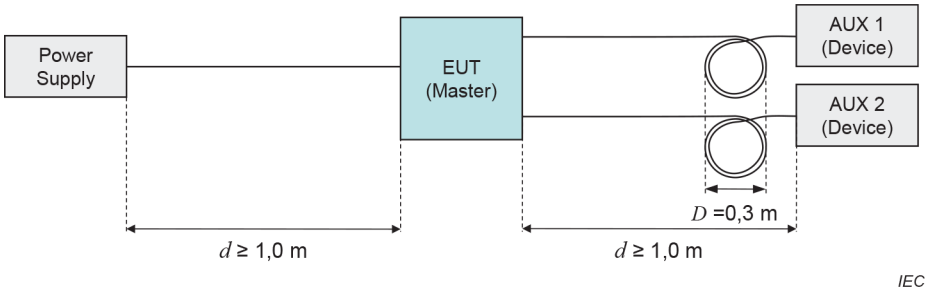


Figure H.1 – Test setup for electrostatic discharge (Master)

H.1.6.3 Radio-frequency electromagnetic field

Figure H.2 shows the test setup for radio-frequency electromagnetic field according to IEC 61000-4-3.

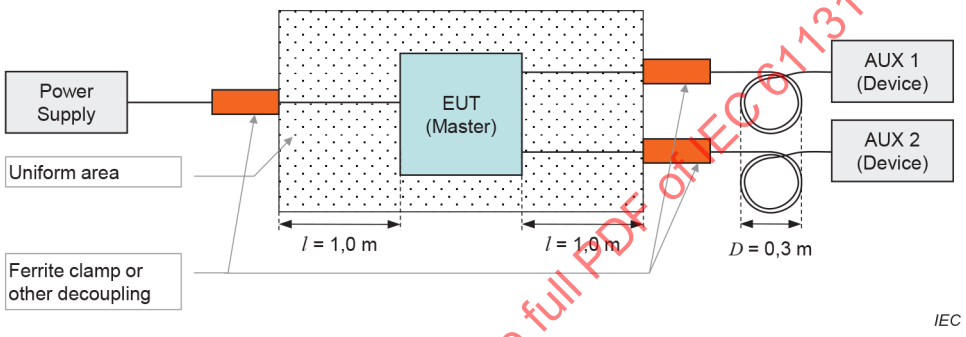


Figure H.2 – Test setup for RF electromagnetic field (Master)

H.1.6.4 Fast transients (burst)

Figure H.3 shows the test setup for fast transients according to IEC 61000-4-4. No coupling into SDCI line to AUX 2 is required.

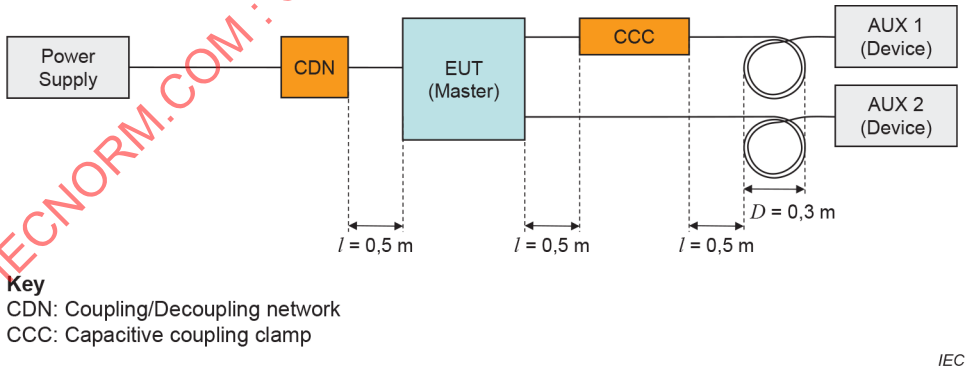


Figure H.3 – Test setup for fast transients (Master)

H.1.6.5 Radio-frequency common mode

Figure H.4 shows the test setup for radio-frequency common mode according to IEC 61000-4-6.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

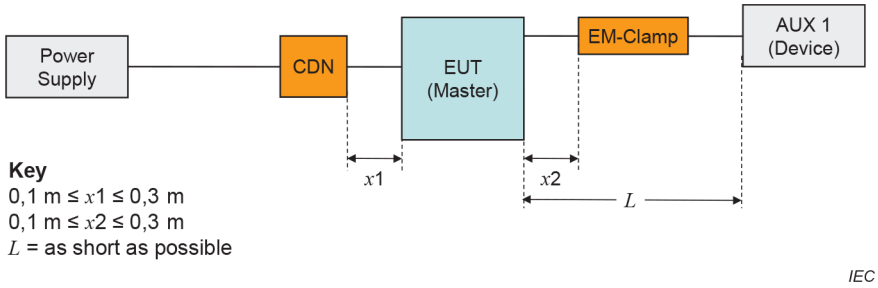


Figure H.4 – Test setup for RF common mode (Master)

H.1.7 Test configurations for Devices

H.1.7.1 General rules

For the test of Devices, the following rules apply:

– In the following test setup diagrams only the SDCI and the power supply cables are shown. All other cables shall be treated as required by the relevant product standard.

– Grounding of the Master and the Devices shall be according to the relevant product standard or user manual.

– Where not otherwise stated, the SDCI cable shall have an overall length of 20 m. Excess length laid as an inductive coil with a diameter of 0,3 m, where applicable mounted 0,1 m above RefGND.

– Where applicable, the auxiliary Devices shall be placed 10 cm above RefGND.

– The test with Device AUX 2 is optional.

H.1.7.2 Electrostatic discharges

Figure H.5 shows the test setup for electrostatic discharge according to IEC 61000-4-2.

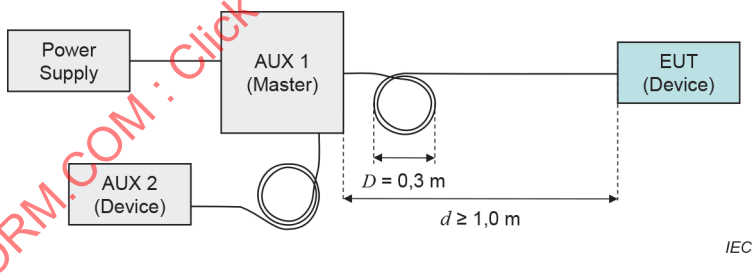


Figure H.5 – Test setup for electrostatic discharges (Device)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

H.1.7.3 Radio-frequency electromagnetic field

Figure H.6 shows the test setup for radio-frequency electromagnetic field according to IEC 61000-4-3.

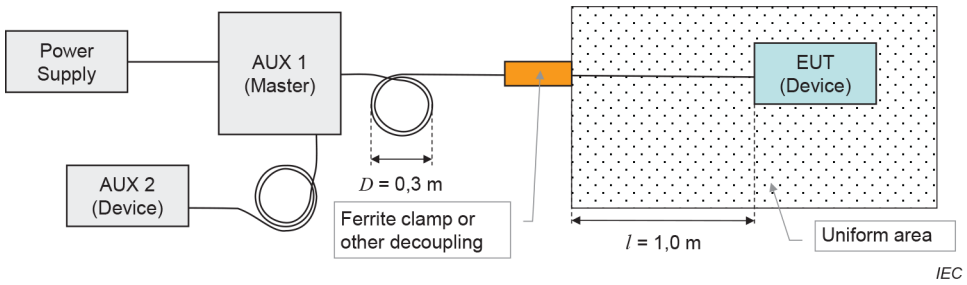


Figure H.6 – Test setup for RF electromagnetic field (Device)

H.1.7.4 Fast transients (burst)

Figure H.7 shows the test setup for fast transients according to IEC 61000-4-4.

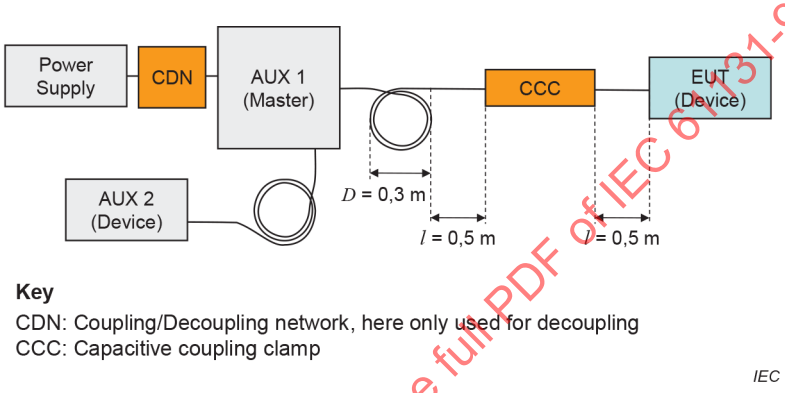


Figure H.7 – Test setup for fast transients (Device)

H.1.7.5 Radio-frequency common mode

Figure H.8 shows the test setup for radio-frequency common mode according to IEC 61000-4-6.

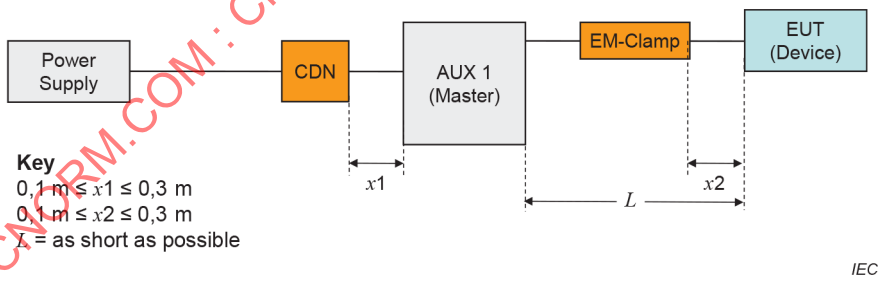


Figure H.8 – Test setup for RF common mode (Device)

H.2 Test strategies for conformity

H.2.1 Test of a Device

The Master AUX 1 (see Figure H.5 to Figure H.8) shall continuously send an M-sequence TYPE_0 (read Direct Parameter page 2) message at the cycle time specified in Table H.1 and count the missing and the erroneous Device responses. Both numbers shall be added and indicated.

NOTE Detailed instructions for the Device tests are specified in [9].

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

H.2.2 Test of a Master

The Device AUX 1 (see Figure H.1 to Figure H.4) shall use M-sequence TYPE_2_5. Its input Process Data shall be generated by an 8-bit random or pseudo random generator. The Master shall copy the input Process Data of any received Device message to the output Process Data of the next Master message to be sent. The cycle time should be according to Table H.1. If not possible, the number of M-sequences for the test shall be calculated according to the algorithm in Clause I.2 and rounded up. The used cycle time and number of M-sequences shall be documented in test records. The Device AUX 1 shall compare the output Process Data with the previously sent input Process Data and count the number of deviations. The Device shall also count the number of missing (not received within the expected cycle time) or received perturbed Master messages. All numbers shall be added and indicated.

NOTE 1 A deviation of sent and received Process Data indicates to the AUX1 that the EUT (Master) did not receive the Device message.

NOTE 2 Detailed instructions for the Master tests are specified in [9].

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex I
(informative)

Residual error probabilities

I.1 Residual error probability of the SDCI data integrity mechanism

Figure I.1 shows the residual error probability (REP) of the SDCI data integrity mechanism consisting of the checksum data integrity procedure ("XOR6") as specified in A.1.6 and the UART parity. The diagram refers to IEC 60870-5-1 [2] with its data integrity class I2 for a minimum Hamming distance of 4 (red dotted line).

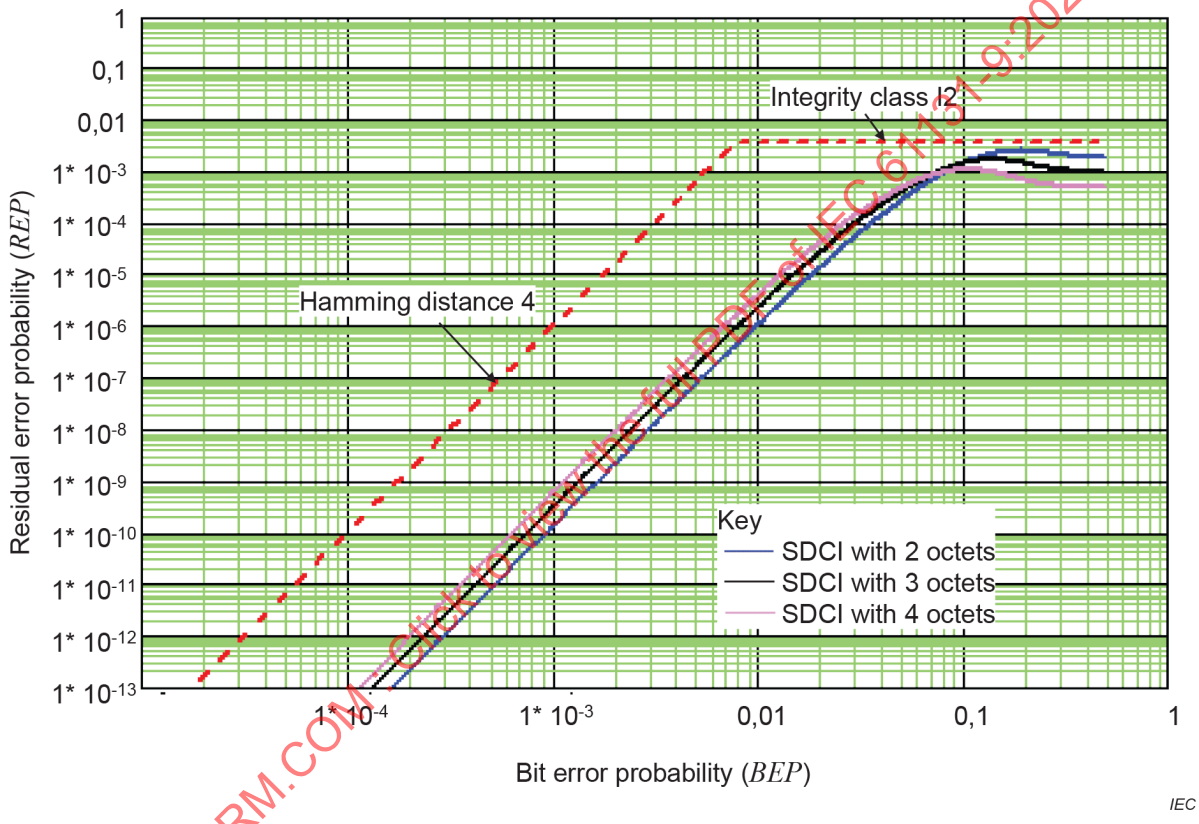


Figure I.1 – Residual error probability for the SDCI data integrity mechanism

The blue line shows the residual error curve for a data length of 2 octets. The black curve shows the residual error curve for a data length of 3 octets. The purple curve shows the residual error curve for a data length of 4 octets.

I.2 Derivation of EMC test conditions

The performance criterion A in H.1.3 is derived from requirements specified in IEC 61158-2 [3] in respect to interference susceptibility and error rates (citation; "*frames*" translates into "messages" within this document):

- *Only 1 undetected erroneous frame in 20 years at 1 600 frames/s*
- *The ratio of undetected to detected frames shall not exceed 10^{-6}*
- *EMC tests shall not show more than 6 erroneous frames within 100 000 frames*

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

With SDCI, the first requirement transforms into Equation (I.1). This equation allows for the determination of a value of BEP. The equation can be resolved in a numerical way.

$$F20 \times R(BEP) \leq 1 \tag{I.1}$$

where

$F20$ is the number of messages in 20 years;

$R(BEP)$ is the residual error probability of the checksum and parity mechanism (Figure I.1);

BEP is the bit error probability from Figure I.1.

The objective of the EMC test is to prove that the BEP of the SDCI communication meets the value determined in the first step. The maximum number of detected perturbed messages is chosen to be 6 here for practical reasons. The number of required SDCI test messages can be determined with the help of Equation (I.2) and the value of BEP determined in the first step.

$$NoTF \geq \frac{1}{BEP} \times \frac{1}{BitPerF} \times NopErr \tag{I.2}$$

where

$NoTF$ is the number of test messages;

$BitPerF$ is the number of bits per message;

$NopErr$ is the maximum number of detected perturbed messages = 6.

Equation (I.2) is only valid under the assumption that messages with 1 bit error are more frequent than messages with more bit errors. An M-sequence consists of two messages. Therefore, the calculated number of test messages has to be divided by 2 to provide the number of M-sequences for Table H.1.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex J
(informative)

Example sequence of an ISDU transmission

Figure J.1 demonstrates an example for the transmission of ISDUs using an AL_Read service with a 16-bit Index and Subindex for 19 octets of user data with mapping to an M-sequence TYPE_2_5 for sensors and with interruption in case of an Event transmission.

Master							Device		
		FC	CKT	PD	OD	OD	PD	CKS	
comment (state, action) (see in Table 46)	cycle nr	R W 1bit	Com Chan. 2bit	Flow CTRL 5bit	Frame Typ 2bit	CHK 6bit	Process Data 8bit	OnReq Data Master 8bit	OnReq Data Device 8bit	Process Data	CHK E PD	comment (state, action)
Idle_1	0	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	OnReq idle
ISDURequest_2, transmission,	1	0111	0000		10	xxxxxx	xxxxxxxx	1011 0101		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	2	0110	0001		10	xxxxxx	xxxxxxxx	Index(hi)		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	3	0110	0010		10	xxxxxx	xxxxxxxx	Index(lo)		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	4	0110	0011		10	xxxxxx	xxxxxxxx	Subindex		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	5	0110	0100		10	xxxxxx	xxxxxxxx	CHKPDU		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDUWait_3, start ISDU Timer	6	1111	0000		10	xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	7	1111	0000		10	xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	8	1111	0000		10	xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	9	1111	0000		10	xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	10	1111	0000		10	xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUResponse_4, reception Stop ISDU Timer	11	1111	0000		10	xxxxxx	xxxxxxxx		1101 0001	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	12	1110	0001		10	xxxxxx	xxxxxxxx		0001 0011	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	13	1110	0010		10	xxxxxx	xxxxxxxx		Data 1	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	14	1110	0011		10	xxxxxx	xxxxxxxx		Data 2	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	15	1110	0100		10	xxxxxx	xxxxxxxx		Data 3	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	16	1110	0101		10	xxxxxx	xxxxxxxx		Data 4	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	17	1110	0110		10	xxxxxx	xxxxxxxx		Data 5	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	18	1110	0111		10	xxxxxx	xxxxxxxx		Data 6	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	19	1110	1000		10	xxxxxx	xxxxxxxx		Data 7	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, no response, retry in next cycle	20	1110	1001		10	Err	xxxxxxxx				xxxxxx	ISDUResponse_4, korrupted CHK, don' t send resp.
ISDUResponse_4, no response, retry in next cycle	21	1110	1001		10	Err	xxxxxxxx				xxxxxx	ISDUResponse_4, corrupted CHK, don' t send resp.
ISDUResponse_4, reception	22	1110	1001		10	xxxxxx	xxxxxxxx		Data 8	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	34	1110	1010		10	xxxxxx	xxxxxxxx		Data 9	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception, start eventhandler	35	1110	1011		10	xxxxxx	xxxxxxxx		Data 10	xxxxxxxx	1 0 xxxxxx	ISDUResponse_4, transmission, freeze event
Read_Event_2, reception	36	1100	0000		10	xxxxxx	xxxxxxxx		Diag State with detail	xxxxxxxx	1 0 xxxxxx	Read_Event_2, transmission
Read_Event_2, reception	37	110x	xxxx		10	xxxxxx	xxxxxxxx		Event qualifier	xxxxxxxx	1 0 xxxxxx	Read_Event_2, transmission
Command handler_2, transmission set PDOutdata state to invalid	38	0010	0000		10	xxxxxx	xxxxxxxx	1001 1001		xxxxxxxx	1 0 xxxxxx	ComandHandler_2, reception, set PDOutdata state to invalid
Read_Event_2, reception	39	110x	xxxx		10	xxxxxx	xxxxxxxx		ErrorCode msb	xxxxxxxx	1 0 xxxxxx	Read_Event_2, transmission
Read_Event_2, reception	40	110x	xxxx		10	xxxxxx	xxxxxxxx		ErrorCode lsb	xxxxxxxx	1 0 xxxxxx	Read_Event_2, transmission
EventConfirmation_4, transmission, event handler idle	41	0100	0000		10	xxxxxx	xxxxxxxx	xxxxxxxx		xxxxxxxx	0 0 xxxxxx	EventConfirmation, reception
ISDUResponse_4, reception	42	1110	1100		10	xxxxxx	xxxxxxxx		Data 11	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	43	1110	1101		10	xxxxxx	xxxxxxxx		Data 12	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	44	1110	1110		10	xxxxxx	xxxxxxxx		Data 13	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	45	1110	1111		10	xxxxxx	xxxxxxxx		Data 14	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	46	1110	0000		10	xxxxxx	xxxxxxxx		Data 15	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	47	1110	0001		10	xxxxxx	xxxxxxxx		Data 16	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	48	1110	0010		10	xxxxxx	xxxxxxxx		CHKPDU	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
Idle_1	49	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	50	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	51	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	52	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Write Parameter, transmission	53	0011	0000		10	xxxxxx	xxxxxxxx	xxxxxxxx		xxxxxxxx	0 0 xxxxxx	Write Parameter, reception
Read Parameter, reception	54	1011	0000		10	xxxxxx	xxxxxxxx		xxxxxxxx	xxxxxxxx	0 0 xxxxxx	Read Parameter, transmission
Idle_1	55	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	56	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	57	1111	0001		10	xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1

IEC

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure J.1 – Example for ISDU transmissions *(1 of 2)*

ISDURequest_2, transmission	58	0111 0000	10 xxxxxx	xxxxxxxx	0001 1011		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	59	0110 0001	10 xxxxxx	xxxxxxxx	Index		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	60	0110 0010	10 xxxxxx	xxxxxxxx	Data 1		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	61	0110 0011	10 xxxxxx	xxxxxxxx	Data 2		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	62	0110 0100	10 xxxxxx	xxxxxxxx	Data 3		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	63	0110 0101	10 xxxxxx	xxxxxxxx	Data 4		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	64	0110 0110	10 xxxxxx	xxxxxxxx	Data 5		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	65	0110 0111	10 xxxxxx	xxxxxxxx	Data 6		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	66	0110 1000	10 xxxxxx	xxxxxxxx	Data 7		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	67	0110 1001	10 xxxxxx	xxxxxxxx	Data 8		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	68	0110 1010	10 xxxxxx	xxxxxxxx	CHKPDU		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDUWait_3, start ISDU Timer	69	1111 0000	10 xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUResponse_4, reception Stop ISDU Timer	70	1111 0000	10 xxxxxx	xxxxxxxx		0101 0010	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	71	1110 0001	10 xxxxxx	xxxxxxxx		CHKPDU	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
Idle_1	72	1111 0001	10 xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	73	1111 0001	10 xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
ISDURequest_2, transmission,	74	0111 0000	10 xxxxxx	xxxxxxxx	1011 0101		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	75	0110 0001	10 xxxxxx	xxxxxxxx	Index(hi)		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	76	0110 0010	10 xxxxxx	xxxxxxxx	Index(lo)		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	77	0110 0011	10 xxxxxx	xxxxxxxx	Subindex		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDURequest_2, transmission	78	0110 0100	10 xxxxxx	xxxxxxxx	CHKPDU		xxxxxxxx	0 0 xxxxxx	ISDURequest_2, reception
ISDUWait_3, start ISDU Timer	79	1111 0000	10 xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	80	1111 0000	10 xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	81	1111 0000	10 xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	82	1111 0000	10 xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUWait_3, inc. ISDU timer	83	1111 0000	10 xxxxxx	xxxxxxxx		0000 0001	xxxxxxxx	0 0 xxxxxx	ISDUWait_3, application busy
ISDUResponse_4, reception Stop ISDU Timer	84	1111 0000	10 xxxxxx	xxxxxxxx		1101 0001	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	85	1110 0001	10 xxxxxx	xxxxxxxx		0001 1110	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, reception	86	1110 0010	10 xxxxxx	xxxxxxxx		Data 1	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, transmission
ISDUResponse_4, ABORT	87	1111 1111	10 xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	ISDUResponse_4, ABORT
Idle_1	88	1111 0001	10 xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1
Idle_1	89	1111 0001	10 xxxxxx	xxxxxxxx		0000 0000	xxxxxxxx	0 0 xxxxxx	Idle_1

IEC

Figure J.1 *(2 of 2)*

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Annex K
(informative)

Recommended methods for detecting parameter changes

K.1 CRC signature

Cyclic Redundancy Checking belongs to the HASH function family. A CRC signature across all changeable parameters can be calculated by the Device with the help of a so-called proper generator polynomial. The calculation results in a different signature whenever the parameter set has been changed. It should be noted that the signature also secures the octet order within the parameter set. Any change in the order when calculating the signature will lead to a different value. The quality of securing (undetected changes) depends heavily on both the CRC generator polynomial and the length (number of octets) of the parameter set. The seed value should be > 0. One calculation method directly uses the formula, another one uses octet shifting and lookup tables. The first one requests less program memory and is a bit slower, the other one requires memory for a lookup table (1×2^{10} octets for a 32-bit signature) and is fast. The parameter data set comparison is performed in state "Checksum_9" of the Data Storage (DS) state machine in Figure 104. Table K.1 lists several possible generator polynomials and their detection level.

Table K.1 – Proper CRC generator polynomials

Generator polynomial	Signature	Data length	Undetected changes
0x9B	8 bits	1 octet	$< 2^{-8}$ (not recommended)
0x4EAB	16 bits	1 < octets < 3	$< 2^{-16}$ (not recommended)
0x5D6DCB	24 bits	1 < octets < 4	$< 2^{-24}$ (not recommended)
0xF4ACFB13	32 bits	1 < octets < 2^{32}	$< 2^{-32}$ (recommended)

K.2 Revision counter

A 32-bit revision counter can be implemented, counting any change of the parameter set. The Device shall use a random initial value for the Revision Counter. The counter itself shall not be stored via the Index List of the Device. After the download the actual counter value is read back from the Device to avoid multiple writing initiated by the download sequence. The parameter data set comparison is performed in state "Checksum_9" of the Data Storage (DS) state machine in Figure 104.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Bibliography

[1] IEC 60050 (all parts), *International Electrotechnical Vocabulary (IEV)* (available at http://www.electropedia.org)

[2] IEC 60870-5-1, *Telecontrol equipment and systems. Part 5: Transmission protocols – Section One: Transmission frame formats*

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

[4] IEC TR 62453-61, *Field device tool interface specification – Device Type Manager (DTM) Styleguide for common object model*

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

[6] IO-Link Community, *IO Device Description (IODD),* Order No. 10.012 (available at <http://www.io-link.com>)

[7] IO-Link Community, *IO-Link Common Profile,* Order No. 10.072 (available at <http://www.io-link.com>)

[8] IO-Link Community, *IO-Link Communication, V1.0, January 2009, Order No. 10.002* (available at <http://www.io-link.com>)

[9] IO-Link Community, *IO-Link Test Specification,* Order No. 10.032 (available at <http://www.io-link.com>)

[10] IO-Link Community, *IO-Link Safety System Extensions,* Order No. 10.092 (available at <http://www.io-link.com>)

[11] IO-Link Community, *IO-Link Wireless System Extensions,* Order No. 10.112 (available at <http://www.io-link.com>)

[12] IO-Link Community, *IO-Link Common Gateway Profile*, work in progress

[13] IEC 62443 (all parts), *Industrial communication networks – Network and system security*

[14] IEC TR 62390, *Common automation device – Profile guideline*

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

[16] ISO/IEC 2022, *Information technology – Character code structure and extension techniques*

[17] ISO/IEC 10646, *Information technology – Universal coded character set (UCS)*

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

[19] ISO/IEC 19505 (all parts), *Information technology – Object Management Group Unified Modeling Language (OMG UML)*

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

[20] ISO 1177, *Information processing – Character structure for start/stop and synchronous character oriented transmission*

[21] ANSI/IEEE Std 754, *IEEE Standard for Floating-point Arithmetic*

[22] Internet Engineering Task Force (IETF): RFC 1305 – *Network Time Protocol Version 4: Specification, Implementation, and Analysis;* available at < www.ietf.org >

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

SOMMAIRE

AVANT-PROPOS 349
INTRODUCTION 351
0.1 Généralités 351
0.2 Déclaration de propriété 352
1 Domaine d'application 353
2 Références normatives 353
3 Termes, définitions, symboles, termes abrégés et conventions 354
3.1 Termes et définitions 354
3.2 Symboles et termes abrégés 359
3.3 Conventions 361
3.3.1 Généralités 361
3.3.2 Paramètres de service 361
3.3.3 Procédures de services 362
3.3.4 Attributs de service 362
3.3.5 Figures 363
3.3.6 Tableaux 363
3.3.7 Ordre de transmission des octets 363
3.3.8 Descriptifs comportementaux 363
4 Vue d'ensemble de l'interface SDCI (IO-Link™) 365
4.1 Objectif de la technologie 365
4.2 Positionnement dans la hiérarchie de l'automatisation 365
4.3 Câblage, connecteurs et alimentation 366
4.4 Fonctionnalités de communication de l'interface SDCI 367
4.5 Rôle d'un Maître 369
4.6 Configuration SDCI 370
4.7 Mise en correspondance avec les bus de terrain 370
4.8 Structure de la norme 370
5 Couche Physique (PL) 371
5.1 Généralités 371
5.1.1 Principes de base 371
5.1.2 Topologie 372
5.2 Services de Couche Physique 372
5.2.1 Vue d'ensemble 372
5.2.2 Services de PL 373
5.3 Émetteur/Récepteur 375
5.3.1 Méthode de description 375
5.3.2 Exigences électriques 375
5.3.3 Exigences temporelles 381
5.4 Alimentation 386
5.4.1 Options d'alimentation 386
5.4.2 Classe d'Accès B 386
5.4.3 Exigences de mise sous tension 387
5.5 Support 387
5.5.1 Connecteurs 387
5.5.2 Câblage 389
6 Entrées et sorties normalisées (SIO) 390

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7 Couche de liaison de données (DL) 390
7.1 Généralités 390
7.2 Services de la couche liaison de données 392
7.2.1 Services de la section B de couche DL 392
7.2.2 Services de la section A de couche DL 403
7.3 Protocole de la couche liaison de données 407
7.3.1 Vue d'ensemble 407
7.3.2 Module de traitement de mode DL 408
7.3.3 Module de traitement des messages 418
7.3.4 Module de traitement des Données de Processus 427
7.3.5 Module de traitement de Données à la Demande 431
7.3.6 Module de traitement des ISDU 434
7.3.7 Module de traitement des commandes 439
7.3.8 Module de traitement d'Événements 441
8 Couche d'application (AL) 445
8.1 Généralités 445
8.2 Services de la Couche d'Application 446
8.2.1 Services AL au sein du Maître et du Dispositif 446
8.2.2 Services d'AL 447
8.3 Protocole de couche d'application 455
8.3.1 Vue d'ensemble 455
8.3.2 Transfert de Données à la Demande 455
8.3.3 Traitement des Événements 461
8.3.4 Cycles de Données de Processus 464
9 Gestion Système (SM) 465
9.1 Généralités 465
9.2 Gestion Système du Maître 465
9.2.1 Vue d'ensemble 465
9.2.2 Services SM du Maître 468
9.2.3 Protocole SM du Maître 472
9.3 Gestion Système du Dispositif 481
9.3.1 Vue d'ensemble 481
9.3.2 Services SM du Dispositif 483
9.3.3 Protocole SM du Dispositif 489
10 Dispositif 496
10.1 Aperçu 496
10.2 Échange de Données de Processus (PDE) 497
10.3 Gestionnaire de Paramètres (PM) 497
10.3.1 Généralités 497
10.3.2 Diagramme d'états du Gestionnaire de Paramètres 498
10.3.3 Paramètre dynamique 500
10.3.4 Paramètre unique 500
10.3.5 Paramètre de bloc 502
10.3.6 Accès concurrent au paramétrage 505
10.3.7 Traitement des commandes 505
10.4 Stockage des Données (DS) 506
10.4.1 Généralités 506
10.4.2 Diagramme d'états du Stockage des Données 506

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.4.3 Configuration du DS 509
10.4.4 Espace mémoire du DS 509
10.4.5 Index_List de DS 509
10.4.6 Disponibilité des paramètres de DS 509
10.4.7 DS sans ISDU 509
10.4.8 Indication de modification de paramètres de DS 509
10.5 Répartiteur d'Événements (ED) 509
10.6 Fonctionnalités de Dispositif 510
10.6.1 Généralités 510
10.6.2 Rétrocompatibilité d'un Dispositif 510
10.6.3 Compatibilité de versions du protocole 510
10.6.4 Indication visuelle d'état de l'interface SDCI 511
10.6.5 Verrouillage de l'Accès aux paramètres 511
10.6.6 Verrouillage du Stockage des Données 511
10.6.7 Verrouillage des entrées locales du paramètre 511
10.6.8 Verrouillage de l'interface utilisateur 511
10.6.9 Décalage temporel 511
10.6.10 Concept de Stockage des Données 512
10.6.11 Paramètre de bloc 512
10.7 Options de réinitialisation du Dispositif 512
10.7.1 Vue d'ensemble 512
10.7.2 Réinitialisation du Dispositif 513
10.7.3 Réinitialisation de l'application 514
10.7.4 Restauration des réglages d'usine 514
10.7.5 Retour à la boîte 514
10.8 Règles et contraintes de conception d'un Dispositif 515
10.8.1 Généralités 515
10.8.2 Données de Processus 515
10.8.3 Perte de communication 515
10.8.4 Paramètre Direct 515
10.8.5 Canal de communication des ISDU 515
10.8.6 Règles de DeviceID relatives aux variantes de Dispositifs 516
10.8.7 Constantes de protocole 516
10.9 Descriptif de Dispositif d'E/S (IODD) 517
10.10 Diagnostic d'un Dispositif 517
10.10.1 Concepts 517
10.10.2 Événements 518
10.10.3 Indicateurs visuels 519
10.11 Connectivité d'un Dispositif 520
11 Maître 520
11.1 Vue d'ensemble 520
11.1.1 Positionnement d'un Maître et des Applications de Passerelle 520
11.1.2 Structure, applications et services d'un Maître 521
11.1.3 Vue d'objet d'un Maître et ses Accès 523
11.2 Services de l'interface normalisée du Maître (SMI) 523
11.2.1 Vue d'ensemble 523
11.2.2 Structure des arguments de service SMI 525
11.2.3 Simultanéité et hiérarchisation des services SMI 526
11.2.4 SMI_MasterIdentification 526

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.2.5 SMI_PortConfiguration 528
11.2.6 SMI_ReadbackPortConfiguration 529
11.2.7 SMI_PortStatus 530
11.2.8 SMI_DSToParServ 532
11.2.9 SMI_ParServToDS 533
11.2.10 SMI_DeviceWrite 535
11.2.11 SMI_DeviceRead 536
11.2.12 SMI_ParamWriteBatch 537
11.2.13 SMI_ParamReadBatch 539
11.2.14 SMI_PortPowerOffOn 541
11.2.15 SMI_DeviceEvent 542
11.2.16 SMI_PortEvent 543
11.2.17 SMI_PDIn 544
11.2.18 SMI_PDOut 545
11.2.19 SMI_PDInOut 547
11.2.20 SMI_PDInIQ 548
11.2.21 SMI_PDOutIQ 549
11.2.22 SMI_PDReadbackOutIQ 551
11.3 Gestionnaire de configuration (CM) 552
11.3.1 Coordination des applications d'un Maître 552
11.3.2 Diagramme d'états du Gestionnaire de Configuration 555
11.4 Stockage de Données (DS) 560
11.4.1 Vue d'ensemble 560
11.4.2 Objet de données DS 560
11.4.3 Sauvegarde et restauration 560
11.4.4 Diagramme d'états DS 561
11.4.5 Sélection de paramètres pour Stockage des Données 567
11.5 Échange de Données à la Demande (ODE) 567
11.6 Unité de Diagnostic (DU) 569
11.6.1 Généralités 569
11.6.2 Événements spécifiques au Dispositif 569
11.6.3 Événements spécifiques à l'Accès 569
11.6.4 État de diagnostic dynamique 570
11.6.5 Recommandations en matière de meilleure pratique 570
11.7 Échange de PD (PDE) 571
11.7.1 Généralités 571
11.7.2 Mapping d'entrée de Données de Processus 571
11.7.3 Mapping de sortie de Données de Processus 573
11.7.4 État du qualificatif non valide/valide des Données de Processus 574
12 Vue holistique du Stockage de Données 575
12.1 Point de vue de l'utilisateur 575
12.2 Opérations et conditions préalables 575
12.2.1 Objet et objectifs 575
12.2.2 Conditions préalables pour l'activation du mécanisme de Stockage de Données 575
12.2.3 Conditions préalables pour les types de Dispositifs à remplacer 575
12.2.4 Conditions préalables pour les jeux de paramètres 576
12.3 Mise en service 576
12.3.1 Mise en service en ligne 576

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

12.3.2 Mise en service hors site 576
12.4 Niveaux de sauvegarde 577
12.4.1 Objet 577
12.4.2 Vue d'ensemble 577
12.4.3 Mise en service ("Désactivation") 578
12.4.4 Production ("Sauvegarde/Restauration") 578
12.4.5 Production ("Restauration") 579
12.5 Cas d'utilisation 580
12.5.1 Remplacement du Dispositif ("Sauvegarde/Restauration") 580
12.5.2 Remplacement du Dispositif ("Restauration") 580
12.5.3 Remplacement du Maître 580
12.5.4 Réplication de projet 581
13 Intégration 581
13.1 Modèle générique de Maître pour l'intégration système 581
13.2 Rôle des applications passerelles 582
13.2.1 Clients 582
13.2.2 Coordination 582
13.3 Sécurité 582
13.4 Applications passerelles particulières 583
13.4.1 Modification de la configuration du Dispositif, y compris le Stockage des Données 583
13.4.2 Serveur de paramètres et contrôle des recettes 583
13.5 Outil de configuration des Accès et Dispositifs (PDCT) 583
13.5.1 Stratégie 583
13.5.2 Accès au Maître par l'intermédiaire de la SMI 583
13.5.3 Exemples de présentation de base 584
Annexe A (normative) Codages, contraintes temporelles et erreurs 586
A.1 Structure générale et codage de séquences M 586
A.1.1 Vue d'ensemble 586
A.1.2 Commande de séquence M (MC) 586
A.1.3 Somme de contrôle/Type de séquence M (CKT) 587
A.1.4 Données utilisateur (PD ou OD) 587
A.1.5 Somme de contrôle/état (CKS) 588
A.1.6 Calcul de la somme de contrôle 589
A.2 Types de séquence M 590
A.2.1 Vue d'ensemble 590
A.2.2 Séquence M de TYPE _0 590
A.2.3 Séquence M de TYPE_1_x 590
A.2.4 Séquence M de TYPE_2_x 592
A.2.5 Type de séquence M 3 594
A.2.6 Utilisation des types de séquence M pour les modes STARTUP, PREOPERATE et OPERATE 594
A.3 Contraintes temporelles 596
A.3.1 Généralités 596
A.3.2 Temps binaire 596
A.3.3 Retard de transmission de trame UART de Maître (Accès) 597
A.3.4 Retard de transmission de trame UART de Dispositifs 597
A.3.5 Temps de réponse de Dispositifs 597
A.3.6 Temps de séquence M 597

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

A.3.7 Durée de cycle 598
A.3.8 Temps au repos 598
A.3.9 Temps de récupération 598
A.4 Erreurs et mesures correctives 599
A.4.1 Erreurs d'UART 599
A.4.2 Erreurs de réactivation 599
A.4.3 Erreurs de transmission 599
A.4.4 Erreurs de protocole 600
A.5 Structure générale et codage d'ISDU 600
A.5.1 Vue d'ensemble 600
A.5.2 "Service I" 600
A.5.3 Longueur étendue (ExtLength) 601
A.5.4 Index et Sous-index 602
A.5.5 Données 602
A.5.6 Vérification des ISDU (CHKPDU) 602
A.5.7 Exemples d'ISDU 603
A.6 Structure générale et codage d'Événements 605
A.6.1 Généralités 605
A.6.2 StatusCode de type 1 (sans détails) 605
A.6.3 StatusCode de type 2 (avec détails) 606
A.6.4 EventQualifier 607
A.6.5 EventCode 608
Annexe B (normative) Paramètres et commandes 609
B.1 Pages 1 et 2 de Paramètres Directs 609
B.1.1 Vue d'ensemble 609
B.1.2 MasterCommand 611
B.1.3 MasterCycleTime et MinCycleTime 611
B.1.4 Capacité de séquence M 612
B.1.5 RevisionID (RID) 613
B.1.6 ProcessDataIn 613
B.1.7 ProcessDataOut 614
B.1.8 VendorID (VID) 614
B.1.9 DeviceID (DID) 614
B.1.10 FunctionID (FID) 614
B.1.11 SystemCommand 615
B.1.12 Page 2 de Paramètres Directs spécifiques au Dispositif 615
B.2 Paramètres de Dispositif prédéfinis 615
B.2.1 Vue d'ensemble 615
B.2.2 SystemCommand 618
B.2.3 DataStorageIndex 619
B.2.4 DeviceAccessLocks 621
B.2.5 ProfileCharacteristic 622
B.2.6 PDInputDescriptor 623
B.2.7 PDOutputDescriptor 623
B.2.8 VendorName 623
B.2.9 VendorText 623
B.2.10 ProductName 623
B.2.11 ProductID 623
B.2.12 ProductText 623

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

B.2.13 SerialNumber 623
B.2.14 HardwareRevision 624
B.2.15 FirmwareRevision 624
B.2.16 ApplicationSpecificTag 624
B.2.17 FunctionTag 624
B.2.18 LocationTag 624
B.2.19 ErrorCount 624
B.2.20 DeviceStatus 624
B.2.21 DetailedDeviceStatus 625
B.2.22 ProcessDataInput 626
B.2.23 ProcessDataOutput 626
B.2.24 OffsetTime 626
B.2.25 Paramètre de profil (réservé) 627
B.2.26 Index préférentiel 627
B.2.27 Index étendu 627
B.2.28 Index spécifique au profil (réservé) 627
Annexe C (normative) ErrorTypes (erreurs d'ISDU) 628
C.1 Généralités 628
C.2 ErrorTypes relatifs à l'application 628
C.2.1 Vue d'ensemble 628
C.2.2 Erreur d'application du Dispositif – pas de détails 629
C.2.3 Index non disponible 629
C.2.4 Sous-index non disponible 629
C.2.5 Service temporairement non disponible 629
C.2.6 Service temporairement non disponible – contrôle local 629
C.2.7 Service temporairement non disponible – contrôle du Dispositif 629
C.2.8 Accès refusé 629
C.2.9 Valeur de paramètre hors limites 630
C.2.10 Valeur de paramètre supérieure à la limite 630
C.2.11 Valeur de paramètre inférieure à la limite 630
C.2.12 Dépassement de longueur de paramètre 630
C.2.13 Longueur de paramètre insuffisante 630
C.2.14 Fonction non disponible 630
C.2.15 Fonction temporairement indisponible 630
C.2.16 Jeu de paramètres non valide 630
C.2.17 Jeu de paramètres incohérent 630
C.2.18 Application non prête 630
C.2.19 Spécifique au fournisseur 631
C.3 ErrorTypes dérivés 631
C.3.1 Vue d'ensemble 631
C.3.2 Maître – Erreur de communication 631
C.3.3 Maître – Expiration du délai d'ISDU 631
C.3.4 Événement de Dispositif – Erreur d'ISDU 632
C.3.5 Événement de Dispositif – Primitive de service ISDU illicite 632
C.3.6 Maître – Erreur de somme de contrôle d'ISDU 632
C.3.7 Maître – Primitive de service ISDU illicite 632
C.3.8 Événement de Dispositif – Débordement de tampon d'ISDU 632
C.4 ErrorTypes relatifs à la SMI 632
C.4.1 Vue d'ensemble 632

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

C.4.2 ArgBlock inconnu 633
C.4.3 Type de contenu ArgBlock incorrect 633
C.4.4 Dispositif ne communique pas 633
C.4.5 Service inconnu 633
C.4.6 Données de Processus non accessibles 633
C.4.7 Mémoire insuffisante 633
C.4.8 Numéro d'Accès incorrect 633
C.4.9 Longueur ArgBlock incorrecte 633
C.4.10 Maître occupé 633
C.4.11 Données DS incohérente 633
C.4.12 Erreur Dispositif ou Maître 633
Annexe D (normative) EventCodes (informations de diagnostic) 634
D.1 Généralités 634
D.2 EventCodes pour des Dispositifs 634
D.3 EventCodes pour les Accès 636
Annexe E (normative) Codage des ArgBlocks 639
E.1 Généralités 639
E.2 MasterIdent 640
E.3 PortConfigList 641
E.4 PortStatusList 643
E.5 On-request_Data 645
E.6 DS_Data 646
E.7 DeviceParBatch 646
E.8 IndexList 647
E.9 PortPowerOffOn 647
E.10 PDIn 647
E.11 PDOut 648
E.12 PDInOut 649
E.13 PDInIQ 649
E.14 PDOutIQ 650
E.15 DeviceEvent 650
E.16 PortEvent 650
E.17 VoidBlock 651
E.18 JobError 651
Annexe F (normative) Types de données 652
F.1 Généralités 652
F.2 Types de données de base 652
F.2.1 Généralités 652
F.2.2 BooleanT 652
F.2.3 UIntegerT 652
F.2.4 IntegerT 653
F.2.5 Float32T 655
F.2.6 StringT 656
F.2.7 OctetStringT 656
F.2.8 TimeT 657
F.2.9 TimeSpanT 658
F.3 Types de données composés 659
F.3.1 Généralités 659
F.3.2 ArrayT 659

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

F.3.3 RecordT 660
Annexe G (normative) Structure de l'objet de données "Stockage de Données" 663
Annexe H (normative) Conformité du Maître et du Dispositif 664
H.1 Exigences de compatibilité électromagnétique (CEM) 664
H.1.1 Généralités 664
H.1.2 Conditions de fonctionnement 664
H.1.3 Critères de performances 664
H.1.4 Essais d'immunité exigés 665
H.1.5 Essais d'émission exigés 666
H.1.6 Configurations d'essai de Maître 666
H.1.7 Configurations d'essai de Dispositifs 668
H.2 Stratégies d'essais de conformité 669
H.2.1 Essai d'un Dispositif 669
H.2.2 Essai d'un Maître 670
Annexe I (informative) Probabilités d'erreurs résiduelles 671
I.1 Probabilité d'erreurs résiduelles du mécanisme d'intégrité des données SDCI 671
I.2 Conditions d'essais CEM dérivées 671
Annexe J (informative) Exemple d'une séquence de transmission d'ISDU 673
Annexe K (informative) Méthodes recommandées pour la détection de modifications de paramètres 675
K.1 Signature CRC 675
K.2 Compteur de versions 675
Bibliographie 676

Figure 1 – Exemple de service confirmé 363
Figure 2 – Ordre de stockage en mémoire et de transmission pour des types de données à base de MOTS 363
Figure 3 – Exemple d'état imbriqué 364
Figure 4 – Compatibilité de la SDCI avec l'IEC 61131-2 365
Figure 5 – Domaine de la technologie SDCI au sein de la hiérarchie de l'automatisation 366
Figure 6 – Modèle de Dispositif générique pour SDCI (vue du côté Maître) 367
Figure 7 – Relation entre nature des données et types de transmission 368
Figure 8 – Transfert d'objets au niveau de la couche d'application (AL) 369
Figure 9 – Structure logique du Maître et du Dispositif 370
Figure 10 – Système de connexion à trois conducteurs 371
Figure 11 – Topologie de l'interface SDCI 372
Figure 12 – Couche physique (Maître) 372
Figure 13 – Couche physique (Dispositif) 373
Figure 14 – Schéma de référence du circuit d'attaque de la ligne 375
Figure 15 – Schéma de référence du récepteur 376
Figure 16 – Schéma de référence pour le système de connexion à 3 conducteurs SDCI 376
Figure 17 – Définition des niveaux de tension 377
Figure 18 – Seuils de commutation 378
Figure 19 – Courant d'appel et charge (exemple) 380
Figure 20 – Temps de mise sous tension pour Alimentation 1 381

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 21 – Format d'une trame UART de l'interface SDCI 382
Figure 22 – Diagramme en œil pour la détection de 'H' et 'L' 383
Figure 23 – Diagramme en œil pour la détection correcte d'une trame UART 383
Figure 24 – Demande de réactivation 385
Figure 25 – Définition des classes d'Accès A et B 387
Figure 26 – Vue de face de l'agencement des broches 388
Figure 27 – Schéma de référence de la capacité effective de la ligne et de la résistance de boucle 390
Figure 28 – Structure et services de la couche de liaison de données (Maître) 391
Figure 29 – Structure et services de la couche de liaison de données (Dispositif) 392
Figure 30 – Diagrammes d'états de la couche de liaison de données 408
Figure 31 – Exemple d'une tentative d'établissement de communication 409
Figure 32 – Échec d'une tentative d'établissement de communication 409
Figure 33 – Stratégie de nouvelle tentative d'établissement de communication 410
Figure 34 – Procédure de repli 411
Figure 35 – Diagramme d'états du module de traitement de mode DL du Maître 412
Figure 36 – Sous-machine 1 d'établissement de la communication 413
Figure 37 – Diagramme d'états du module de traitement de mode DL du Dispositif 416
Figure 38 – Séquences de messages SDCI 419
Figure 39 – Vue d'ensemble des types de séquence M 420
Figure 40 – Diagramme d'états du module de traitement des messages du Maître 421
Figure 41 – Sous-machine "Response 3" du module de traitement des messages 422
Figure 42 – Sous-machine "Response 8" du module de traitement des messages 422
Figure 43 – Sous-machine "Response 15" du module de traitement des messages 422
Figure 44 – Diagramme d'états du module de traitement des messages du Dispositif 426
Figure 45 – Mode d'entrelacement pour la transmission segmentée des Données de Processus 428
Figure 46 – Diagramme d'états du module de traitement des Données de Processus du Maître 428
Figure 47 – Diagramme d'états du module de traitement des Données de Processus du Dispositif 430
Figure 48 – Diagramme d'états du module de traitement de Données à la Demande du Maître 431
Figure 49 – Diagramme d'états du module de traitement de Données à la Demande du Dispositif 433
Figure 50 – Structure de l'ISDU 434
Figure 51 – Diagramme d'états du module de traitement des ISDU du Maître 436
Figure 52 – Diagramme d'états du module de traitement des ISDU du Dispositif 437
Figure 53 – Diagramme d'états du module de traitement des commandes du Maître 439
Figure 54 – Diagramme d'états du module de traitement des commandes du Dispositif 441
Figure 55 – Diagramme d'états du module de traitement d'Événements du Maître 443
Figure 56 – Diagramme d'états du module de traitement d'Événements du Dispositif 444
Figure 57 – Structure et services de la couche d'application (Maître) 446
Figure 58 – Structure et services de la couche d'application (Dispositif) 446
Figure 59 – Diagramme d'états d'OD de la couche AL du Maître 456

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 60 – Diagramme d'états d'OD de la couche AL du Dispositif 458
Figure 61 – Diagramme de séquence pour la transmission de Données à la Demande 459
Figure 62 – Diagramme de séquence pour des Données à la Demande en cas d'erreurs 460
Figure 63 – Diagramme de séquence pour des Données à la Demande en cas d'expiration de délai 460
Figure 64 – Diagramme d'états d'Événements de la couche AL du Maître 461
Figure 65 – Diagramme d'états d'Événements de la couche AL du Dispositif 462
Figure 66 – Planification d'Événements uniques 463
Figure 67 – Diagramme de séquence pour des Données de Processus de sortie 464
Figure 68 – Diagramme de séquence pour des Données de Processus d'entrée 465
Figure 69 – Structure et services de la Gestion Système du Maître 466
Figure 70 – Organigramme séquentiel du cas d'utilisation "Configuration d'Accès x" 467
Figure 71 – Diagramme d'états principal de la Gestion Système du Maître 474
Figure 72 – Sous-machine de la SM du Maître CheckCompatibility_1 476
Figure 73 – Activité pour l'état "CheckVxy" 478
Figure 74 – Activité pour l'état "CheckCompV10" 478
Figure 75 – Activité pour l'état "CheckComp" 479
Figure 76 – Activité (écriture de paramètres) dans l'état "RestartDevice" 479
Figure 77 – Sous-machine de la SM du Maître checkSerNum_3 480
Figure 78 – Activité (vérification du SerialNumber) pour l'état CheckSerNum_31 481
Figure 79 – Structure et services de la Gestion Système (Dispositif) 482
Figure 80 – Organigramme séquentiel du cas d'utilisation "INACTIVE – SIO – SDCI – SIO" 483
Figure 81 – Diagramme d'états de la Gestion Système du Dispositif 490
Figure 82 – Organigramme séquentiel de démarrage ordinaire du Dispositif 493
Figure 83 – Organigramme séquentiel du démarrage d'un Dispositif en mode compatibilité 494
Figure 84 – Organigramme séquentiel du démarrage d'un Dispositif en cas d'échec de la compatibilité 495
Figure 85 – Structure et services d'un Dispositif 496
Figure 86 – Diagramme d'états du Gestionnaire de Paramètres (PM) 498
Figure 87 – Résultat positif et négatif d'une vérification de paramètre 501
Figure 88 – Téléchargement aval positif de Paramètre de bloc avec demande de Stockage des Données 503
Figure 89 – Téléchargement aval négatif de Paramètre de bloc 504
Figure 90 – Diagramme d'états du Stockage des Données (DS) 507
Figure 91 – Séquence du message de demande de Stockage des Données 508
Figure 92 – Synchronisation du cycle 512
Figure 93 – Flux d'Événements en cas d'erreurs successives 519
Figure 94 – Temporisation de l'indicateur à LED du Dispositif 520
Figure 95 – Relation générique entre technologie SDCI et technologie d'automatisation 521
Figure 96 – Structure, applications et services d'un Maître 522
Figure 97 – Modèle d'objet du Maître et des Accès 523
Figure 98 – Services SMI 524

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 99 – Coordination des applications d'un Maître 553
Figure 100 – Diagramme de séquences de démarrage par l'intermédiaire du Gestionnaire de Configuration 555
Figure 101 – Diagramme d'états du Gestionnaire de Configuration 556
Figure 102 – Activité pour l'état "CheckPortMode_0" 560
Figure 103 – Principal diagramme d'états du mécanisme de Stockage des Données 561
Figure 104 – Sous-machine "UpDownload_2" du mécanisme de Stockage des Données 562
Figure 105 – Sous-machine du Stockage des Données "Upload_7" 563
Figure 106 – Diagramme de séquence du téléchargement amont de Stockage des Données 563
Figure 107 – Sous-machine du Stockage des Données "Download_10" 564
Figure 108 – Diagramme de séquence du téléchargement aval de Stockage des Données 564
Figure 109 – Diagramme d'états de l'Échange de Données à la Demande 568
Figure 110 – Contrôle des flux DeviceEvent 569
Figure 111 – Contrôle des flux Événement d'Accès 570
Figure 112 – Propagation des informations de diagnostic SDCI au moyen d'Événements 571
Figure 113 – Principes du mapping d'entrée de Données de Processus 572
Figure 114 – Informations du qualificatif d'Accès (PQI) 572
Figure 115 – Principes du mapping de sortie de Données de Processus 573
Figure 116 – Propagation de l'état qualificatif de PD entre Maître et Dispositif 574
Figure 117 – Paramètre actif et de sauvegarde 576
Figure 118 – Mise en service hors site 577
Figure 119 – Modèle générique de Maître pour l'intégration système 582
Figure 120 – PDCT par l'intermédiaire de l'application passerelle 584
Figure 121 – Exemple 1 de présentation de l'affichage d'un PDCT 584
Figure 122 – Exemple 2 de présentation de l'affichage d'un PDCT 585
Figure A.1 – Contrôle de séquence M 586
Figure A.2 – Octet de Somme de contrôle/Type de séquence M 587
Figure A.3 – Octet de Somme de contrôle/État 588
Figure A.4 – Principe de calcul et de compression de la somme de contrôle 589
Figure A.5 – Séquence M de TYPE_0 590
Figure A.6 – Séquence M de TYPE_1_1 591
Figure A.7 – Séquence M de TYPE_1_2 591
Figure A.8 – Séquence M de TYPE_1_V 592
Figure A.9 – Séquence M de TYPE_2_1 592
Figure A.10 – Séquence M de TYPE_2_2 593
Figure A.11 – Séquence M de TYPE_2_3 593
Figure A.12 – Séquence M de TYPE_2_4 593
Figure A.13 – Séquence M de TYPE_2_5 594
Figure A.14 – Séquence M de TYPE_2_V 594
Figure A.15 – Temporisation de séquence M 598
Figure A.16 – Octet de "Service I" 600

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure A.17 – Vérification de l'intégrité des ISDU par l'intermédiaire de CHKPDU 603
Figure A.18 – Exemples de formats de demande d'ISDU 603
Figure A.19 – Exemples d'ISDU de réponse 604
Figure A.20 – Exemples d'ISDU de demande de lecture et d'écriture 605
Figure A.21 – Structure du StatusCode de type 1 605
Figure A.22 – Structure du StatusCode de type 2 606
Figure A.23 – Indication d'Événements activés 607
Figure A.24 – Structure du EventQualifier 607
Figure B.1 – Classification et mapping des Paramètres Directs 609
Figure B.2 – MinCycleTime 611
Figure B.3 – Capacité de séquence M 612
Figure B.4 – RevisionID 613
Figure B.5 – ProcessDataIn 613
Figure B.6 – Espace d'index pour des objets de données ISDU 615
Figure B.7 – Structure d'OffsetTime 626
Figure E.1 – Attribution des identifiants d'ArgBlock 639
Figure F.1 – Exemples de codage de petit UIntegerT 653
Figure F.2 – Exemples de codage de grand UIntegerT 653
Figure F.3 – Exemples de codage de IntegerT 655
Figure F.4 – Accès singulier de StringT 656
Figure F.5 – Exemple de codage d'OctetStringT 657
Figure F.6 – Définition de TimeT 657
Figure F.7 – Exemple d'une structure de données ArrayT 659
Figure F.8 – Deuxième exemple (2) d'une structure de RecordT 661
Figure F.9 – Deuxième exemple (3) d'une structure de RecordT 661
Figure F.10 – Demandes d'écriture pour l'Exemple 3 662
Figure H.1 – Montage d'essai de décharges électrostatiques (Maître) 667
Figure H.2 – Montage d'essai de champ électromagnétique aux fréquences radioélectriques (Maître) 667
Figure H.3 – Montage d'essai de transitoires rapides (Maître) 667
Figure H.4 – Montage d'essai de fréquences radioélectriques en mode commun (Maître) 668
Figure H.5 – Montage d'essai de décharges électrostatiques (Dispositif) 668
Figure H.6 – Montage d'essai de champ électromagnétique (Dispositif) 669
Figure H.7 – Montage d'essai de transitoires rapides (Dispositif) 669
Figure H.8 – Montage d'essai de fréquences radioélectriques en mode commun (Dispositif) 669
Figure I.1 – Probabilité d'erreurs résiduelles du mécanisme d'intégrité des données SDCI 671
Figure J.1 – Exemple de transmissions d'ISDU *(1 sur 2)* 673
Figure J.1 *(2 sur 2)* 674

Tableau 1 – Attributions des services du Maître et du Dispositif 373
Tableau 2 – PL_SetMode 374

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 3 – PL_WakeUp 374
Tableau 4 – PL_Transfer 374
Tableau 5 – Caractéristiques électriques d'un récepteur 377
Tableau 6 – Caractéristiques électriques d'un Accès du Maître 379
Tableau 7 – Caractéristiques électriques d'un Dispositif 380
Tableau 8 – Temps de mise sous tension 381
Tableau 9 – Caractéristiques dynamiques de la transmission 384
Tableau 10 – Caractéristiques de la demande de réactivation 385
Tableau 11 – Caractéristiques électriques d'une Classe d'Accès B du Maître 387
Tableau 12 – Connectique et affectation des broches du Maître 388
Tableau 13 – Connectique et affectation des broches du dispositif 389
Tableau 14 – Caractéristiques du câblage 389
Tableau 15 – Affectation des conducteurs du câble 390
Tableau 16 – Attributions des services au sein du Maître et du Dispositif 393
Tableau 17 – DL_ReadParam 393
Tableau 18 – DL_WriteParam 394
Tableau 19 – DL_Read 395
Tableau 20 – DL_Write 395
Tableau 21 – DL_ISDUTransport 396
Tableau 22 – DL_ISDUAbort 397
Tableau 23 – DL_PDOutputUpdate 397
Tableau 24 – DL_PDOutputTransport 398
Tableau 25 – DL_PDInputUpdate 398
Tableau 26 – DL_PDInputTransport 399
Tableau 27 – DL_PDCycle 399
Tableau 28 – DL_SetMode 400
Tableau 29 – DL_Mode 401
Tableau 30 – DL_Event 401
Tableau 31 – DL_EventConf 402
Tableau 32 – DL_EventTrigger 402
Tableau 33 – DL_Control 402
Tableau 34 – Services DL-A au sein du Maître et du Dispositif 403
Tableau 35 – OD 404
Tableau 36 – PD 405
Tableau 37 – EventFlag 406
Tableau 38 – PDInStatus 406
Tableau 39 – MHInfo 406
Tableau 40 – ODTrig 407
Tableau 41 – PDTrig 407
Tableau 42 – Caractéristiques des procédures et des nouvelles tentatives de réactivation 410
Tableau 43 – Caractéristiques temporelles du repli 411
Tableau 44 – Table de transitions d'états du module de traitement de mode DL du Maître 414

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 45 – Table de transitions d'états du module de traitement de mode DL du Dispositif 417
Tableau 46 – Table de transitions d'états du module de traitement des messages du Maître 423
Tableau 47 – Table de transitions d'états du module de traitement des messages du Dispositif 426
Tableau 48 – Table de transitions d'états du module de traitement des Données de Processus du Maître 429
Tableau 49 – Table de transitions d'états du module de traitement des Données de Processus du Dispositif 430
Tableau 50 – Table de transitions d'états du module de traitement des Données à la Demande du Maître 432
Tableau 51 – Table de transitions d'états du module de traitement des Données à la Demande du Dispositif 433
Tableau 52 – Définitions du FlowCTRL 435
Tableau 53 – Table de transitions d'états du module de traitement des ISDU du Maître 436
Tableau 54 – Table de transitions d'états du module de traitement des ISDU du Dispositif 438
Tableau 55 – Codes de contrôle 439
Tableau 56 – Table de transitions d'états du module de traitement des commandes du Maître 440
Tableau 57 – Table de transitions d'états du module de traitement des commandes du Dispositif 441
Tableau 58 – Mémoire d'Événements 442
Tableau 59 – Table de transitions d'états du module de traitement d'Événements du Maître 443
Tableau 60 – Table de transitions d'états du module de traitement d'Événements du Dispositif 445
Tableau 61 – Services AL au sein du Maître et du Dispositif 447
Tableau 62 – AL_Read 447
Tableau 63 – AL_Write 448
Tableau 64 – AL_Abort 449
Tableau 65 – AL_GetInput 450
Tableau 66 – AL_NewInput 451
Tableau 67 – AL_SetInput 451
Tableau 68 – AL_PDCycle 451
Tableau 69 – AL_GetOutput 452
Tableau 70 – AL_NewOutput 452
Tableau 71 – AL_SetOutput 453
Tableau 72 – AL_Event 454
Tableau 73 – AL_Control 455
Tableau 74 – États et transitions pour le diagramme d'états d'OD de la couche AL du Maître 456
Tableau 75 – États et transitions pour le diagramme d'états d'OD de la couche AL du Dispositif 458
Tableau 76 – États et transitions du diagramme d'états d'Événements de la couche AL du Maître 461

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 77 – États et transitions du diagramme d'états d'Événements de la couche AL du Dispositif 462
Tableau 78 – Services SM au sein du Maître 468
Tableau 79 – SM_SetPortConfig 468
Tableau 80 – Définition de InspectionLevel (IL) 469
Tableau 81 – Définitions des Modes Cibles 470
Tableau 82 – SM_GetPortConfig 470
Tableau 83 – SM_PortMode 471
Tableau 84 – SM_Operate 472
Tableau 85 – Table de transitions d'états de la Gestion Système du Maître 475
Tableau 86 – Table de transitions d'états de la sous-machine du Maître CheckCompatibility_1 477
Tableau 87 – Table de transitions d'états de la sous-machine du Maître checkSerNum_3 480
Tableau 88 – Services SM au sein du Dispositif 484
Tableau 89 – SM_SetDeviceCom 484
Tableau 90 – SM_GetDeviceCom 485
Tableau 91 – SM_SetDeviceIdent 486
Tableau 92 – SM_GetDeviceIdent 487
Tableau 93 – SM_SetDeviceMode 488
Tableau 94 – SM_DeviceMode 488
Tableau 95 – Table de transitions d'états de la Gestion Système du Dispositif 491
Tableau 96 – Table de transitions d'états du diagramme d'états du PM 499
Tableau 97 – Définitions des vérifications de paramètre 502
Tableau 98 – Étapes et règles pour la vérification de Paramètre de bloc 504
Tableau 99 – Réponses hiérarchisées de l'ISDU sur les paramètres de commande 505
Tableau 100 – Table de transitions d'états du diagramme d'états de Stockage des Données 507
Tableau 101 – Vue d'ensemble des options de réinitialisation et de leur impact sur les Dispositifs 513
Tableau 102 – Vue d'ensemble des constantes de protocole pour les Dispositifs 516
Tableau 103 – Classification d'incidents de diagnostics de Dispositifs 518
Tableau 104 – Temporisation pour des indicateurs à LED 520
Tableau 105 – Services SMI 524
Tableau 106 – SMI_MasterIdentification 527
Tableau 107 – SMI_PortConfiguration 528
Tableau 108 – SMI_ReadbackPortConfiguration 529
Tableau 109 – SMI_PortStatus 531
Tableau 110 – SMI_DSToParServ 532
Tableau 111 – SMI_ParServToDS 534
Tableau 112 – SMI_DeviceWrite 535
Tableau 113 – SMI_DeviceRead 536
Tableau 114 – SMI_ParamWriteBatch 538
Tableau 115 – SMI_ParamReadBatch 540
Tableau 116 – SMI_PortPowerOffOn 541

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 117 – SMI_DeviceEvent 542
Tableau 118 – SMI_PortEvent 543
Tableau 119 – SMI_PDIn 544
Tableau 120 – SMI_PDOut 546
Tableau 121 – SMI_PDInOut 547
Tableau 122 – SMI_PDInIQ 548
Tableau 123 – SMI_PDOutIQ 550
Tableau 124 – SMI_PDReadbackOutIQ 551
Tableau 125 – Variables internes et Événements contrôlant les applications d'un Maître 553
Tableau 126 – Table de transitions d'états du Gestionnaire de Configuration 556
Tableau 127 – États et transitions des diagrammes d'états de Stockage des Données 565
Tableau 128 – Table de transitions d'états du diagramme d'états de l'ODE 568
Tableau 129 – Niveaux de sauvegarde recommandés du Stockage de Données 578
Tableau 130 – Critères des paramètres de sauvegarde ("Sauvegarde/Restauration") 579
Tableau 131 – Critères des paramètres de sauvegarde ("Restauration") 580
Tableau A.1 – Valeurs du canal de communication 586
Tableau A.2 – Valeurs de R/W 587
Tableau A.3 – Valeurs de types de séquence M 587
Tableau A.4 – Types de données pour les données utilisateur 588
Tableau A.5 – Valeurs d'état des PD 588
Tableau A.6 – Valeurs du drapeau d'Événement 589
Tableau A.7 – Types de séquence M pour le mode STARTUP 595
Tableau A.8 – Types de séquence M pour le mode PREOPERATE 595
Tableau A.9 – Types de séquence M pour le mode OPERATE (protocole hérité) 595
Tableau A.10 – Types de séquence M pour le mode OPERATE 596
Tableau A.11 – Valeurs recommandées de MinCycleTimes 598
Tableau A.12 – Définition du "Service I" réduit (quartet) 600
Tableau A.13 – Syntaxe des ISDU 601
Tableau A.14 – Définition de la longueur réduite (quartet) et de la longueur étendue d'octet ExtLength 601
Tableau A.15 – Utilisation des formats d'Index 602
Tableau A.16 – Mapping des EventCodes (type 1) 606
Tableau A.17 – Valeurs de "INSTANCE" 607
Tableau A.18 – Valeurs de "SOURCE" 608
Tableau A.19 – Valeurs de "TYPE" 608
Tableau A.20 – Valeurs de "MODE" 608
Tableau B.1 – Pages 1 et 2 de Paramètres Directs 610
Tableau B.2 – Types de MasterCommands 611
Tableau B.3 – Valeurs possibles de MasterCycleTime et de MinCycleTime 612
Tableau B.4 – Valeurs de ISDU 612
Tableau B.5 – Valeurs de SIO 614
Tableau B.6 – Combinaisons admises de BYTE (octet) et Length (longueur) 614
Tableau B.7 – Règles de mise en œuvre des paramètres et commandes 616

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau B.8 – Attribution des Index d'objets de données (paramètres de Dispositif) 616
Tableau B.9 – Codage d'une SystemCommand (ISDU) 619
Tableau B.10 – Attributions de DataStorageIndex 620
Tableau B.11 – Structure de l'Index_List 621
Tableau B.12 – Possibilités de verrouillage du Dispositif 622
Tableau B.13 – Paramètre DeviceStatus 625
Tableau B.14 – DetailedDeviceStatus (Index 0x0025) 626
Tableau B.15 – Codage de la base de temps et valeurs d'OffsetTime 627
Tableau C.1 – ErrorTypes 628
Tableau C.2 – ErrorTypes dérivés 631
Tableau C.3 – ErrorTypes relatifs à la SMI 632
Tableau D.1 – EventCodes pour les Dispositifs 634
Tableau D.2 – EventCodes pour les Accès 637
Tableau E.1 – ArgBlock types et leurs ArgBlockID 640
Tableau E.2 – MasterIdent 641
Tableau E.3 – PortConfigList 642
Tableau E.4 – PortStatusList 644
Tableau E.5 – On-request_Data 645
Tableau E.6 – DS_Data 646
Tableau E.7 – DeviceParBatch 646
Tableau E.8 – IndexList 647
Tableau E.9 – PortPowerOffOn 647
Tableau E.10 – PDIn 648
Tableau E.11 – PDOut 648
Tableau E.12 – PDInOut 649
Tableau E.13 – PDInIQ 650
Tableau E.14 – PDOutIQ 650
Tableau E.15 – DeviceEvent 650
Tableau E.16 – PortEvent 651
Tableau E.17 – VoidBlock 651
Tableau E.18 – JobError 651
Tableau F.1 – BooleanT 652
Tableau F.2 – Codage BooleanT 652
Tableau F.3 – UIntegerT 653
Tableau F.4 – IntegerT 653
Tableau F.5 – Codage de IntegerT (8 octets) 654
Tableau F.6 – Codage de IntegerT (4 octets) 654
Tableau F.7 – Codage de IntegerT (2 octets) 654
Tableau F.8 – Codage de IntegerT (1 octet) 654
Tableau F.9 – Float32T 655
Tableau F.10 – Codage de Float32T 655
Tableau F.11 – StringT 656
Tableau F.12 – OctetStringT 657

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau F.13 – TimeT 657
Tableau F.14 – Codage de TimeT 658
Tableau F.15 – TimeSpanT 658
Tableau F.16 – Codage de TimeSpanT 658
Tableau F.17 – Règles de structuration d'ArrayT 659
Tableau F.18 – Exemple d'Accès à un type ArrayT 659
Tableau F.19 – Règles de structuration de RecordT 660
Tableau F.20 – Premier exemple 1 d'Accès à un RecordT 660
Tableau F.21 – Premier exemple 2 d'Accès à un RecordT 661
Tableau F.22 – Premier exemple 3 d'Accès à un RecordT 661
Tableau G.1 – Structure de l'objet de données DS enregistré 663
Tableau G.2 – Informations d'en-tête associées aux objets de données DS enregistrés 663
Tableau H.1 – Conditions d'essai CEM pour l'interface SDCI 665
Tableau H.2 – Niveaux d'essais CEM 665
Tableau K.1 – Polynômes générateurs de CRC corrects 675

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

_____

AUTOMATES PROGRAMMABLES –

Partie 9: Interface de communication numérique point à point pour petits capteurs et actionneurs (SDCI)

AVANT-PROPOS

1) La Commission Électrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. À cet effet, l'IEC – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de l'IEC"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'IEC, participent également aux travaux. L'IEC collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à l'IEC, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de l'IEC, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de l'IEC ou de toute autre Publication de l'IEC, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

L'IEC 61131-9 a été établie par le sous-comité 65B: Équipements de mesure et de contrôle-commande, du comité 65 de l'IEC: Mesure, commande et automation dans les processus industriels. Il s'agit d'une Norme internationale.

Cette seconde édition annule et remplace la première édition parue en 2013. Cette édition constitue une révision technique.

Cette édition inclut les modifications techniques majeures suivantes par rapport à l'édition précédente:

a) les exigences relatives à la Classe d'Accès B ont été redéfinies;
b) les bonnes pratiques relatives au Stockage des Données (sauvegarde des paramètres du Dispositif) et aux fonctions de réinitialisation d'un Dispositif ont été ajoutées;

c) une nouvelle Interface Normalisée du Maître (SMI) pour l'Accès au PLC, IT et un outil technique ont été ajoutés.

Le texte de la présente Norme internationale est issu des documents suivants:

Projet	Rapport de vote
65B/1218/FDIS	65B/1219/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à son approbation.

La langue utilisée pour le développement de la présente Norme internationale est l'anglais.

Le présent document a été rédigé selon les Directives ISO/IEC, Partie 2, il a été développé selon les Directives ISO/IEC, Partie 1 et les Directives ISO/IEC, Supplément IEC, disponibles sous www.iec.ch/members_experts/refdocs. Les principaux types de documents développés par l'IEC sont décrits plus en détail sous www.iec.ch/standardsdev/publications.

Une liste de toutes les parties de la série IEC 61131, publiées sous le titre général *Automates programmables*, peut être consultée sur le site web de l'IEC.

Le comité a décidé que le contenu du présent document ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous http://webstore.iec.ch dans les données relatives au document recherché. À cette date, le document sera

- reconduit,
- supprimé,
- remplacé par une édition révisée, ou
- amendé.

IMPORTANT – Le logo *"colour inside"* qui se trouve sur la page de couverture de ce document indique qu'il contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer ce document en utilisant une imprimante couleur.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

INTRODUCTION

0.1 Généralités

L'IEC 61131-9 fait partie d'une série de normes relatives aux automates programmables et aux périphériques associés, et il convient de la lire conjointement avec les autres parties de la série.

En cas de conflit entre la présente norme et d'autres normes IEC (à l'exception des normes fondamentales de sécurité), il convient de considérer les dispositions du présent document qui régissent le domaine des automates programmables et leurs périphériques associés.

L'utilisation accrue des microcontrôleurs intégrés dans des capteurs et actionneurs de faible coût a été l'occasion d'ajouter des données de diagnostic et de configuration permettant de prendre en charge les exigences croissantes des applications.

Le moteur premier de la technologie SDCI (IO-Link™ [1]) est la nécessité pour ces capteurs et actionneurs de faible coût d'échanger ces données de diagnostic et de configuration avec un contrôleur (PC ou Automate programmable) en utilisant une technologie de communication numérique peu onéreuse tout en préservant une rétrocompatibilité avec les signaux DI/DO courants.

Au sein des concepts de bus de terrain, la technologie SDCI définit une interface générique pour la connexion de capteurs et d'actionneurs à une unité Maître, qui peut être associée à des fonctionnalités de passerelle pour constituer un nœud déporté d'E/S de bus de terrain.

Tout Dispositif conforme SDCI peut être relié à un Accès disponible quelconque de l'interface du Maître. Les Dispositifs conformes SDCI réalisent en leur sein une conversion physique en numérique, puis en communiquent directement le résultat en format normalisé au moyen d'une "commutation codée" de la ligne de signalisation d'E/S 24 V, rendant ainsi inutile l'utilisation de divers modules DI, DO, AI, AO et d'une multitude de câbles.

La topologie physique est réalisée en point à point de chaque Dispositif vers le Maître au moyen de trois conducteurs sur des distances allant jusqu'à 20 m. L'interface SDCI physique est rétrocompatible avec la signalisation E/S 24 V classique définie dans l'IEC 61131-2. Elle permet des vitesses de transmission de 4,8 kbit/s, 38,4 kbit/s et 230,4 kbit/s.

Le Maître de l'interface SDCI détecte, identifie et gère les Dispositifs branchés aux Accès de l'interface.

Des outils permettent d'associer les Dispositifs à leurs Descriptifs de Dispositifs d'E/S électroniques (IODD - I/O Device Description) ainsi que leur configuration ultérieure pour qu'ils correspondent aux exigences des applications.

La technologie SDCI définit trois différents niveaux de fonctionnalités de diagnostic: pour une réaction immédiate par des besoins automatisés au cours de la phase de production, pour une réaction à moyen terme par intervention de l'opérateur ou pour la mise en service et la maintenance à long terme sur la base d'informations de diagnostic étendues.

La structure du présent document est décrite en 4.8.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

---

[1] IO-Link™ est une marque commerciale de "IO-Link Community". Cette information est fournie pour la commodité des utilisateurs du présent document et ne signifie nullement que l'IEC recommande l'emploi exclusif de la marque ou de l'un quelconque de ses produits. La conformité au présent document ne nécessite pas l'utilisation des logos déposés pour IO-Link™. L'utilisation des logos déposés pour IO-LinkTM nécessite l'autorisation du "IO-Link Community".

La conformité à l'IEC 61131-9 ne peut pas être revendiquée si les exigences de l'Annexe H ne sont pas satisfaites.

Les termes d'ordre général sont définis dans l'IEC 61131-1 ou dans la série IEC 60050 [1][2]. Les termes plus spécifiques sont définis dans le présent document.

0.2 Déclaration de propriété

La Commission Électrotechnique Internationale (IEC) attire l'attention sur le fait qu'il est déclaré que la conformité avec les dispositions du présent document peut impliquer l'utilisation d'un brevet. L'IEC ne prend pas position quant à la preuve, à la validité et à la portée de ces droits de propriété.

Le détenteur de ces droits de propriété a donné l'assurance à l'IEC qu'il consent à négocier des licences avec des demandeurs du monde entier à des termes et conditions raisonnables et non discriminatoires. À ce propos, la déclaration du détenteur des droits de propriété est enregistrée à l'IEC. Des informations peuvent être obtenues dans la base de données des droits de propriété, disponible à l'adresse suivante: http://patents.iec.ch.

L'attention est d'autre part attirée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété autres que ceux qui ont été enregistrés dans la base de données des droits de propriété. L'IEC ne saurait être tenue pour responsable de l'identification de tels droits de brevets et de ne pas avoir signalé leur existence.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

2 Les chiffres entre crochets renvoient à la Bibliographie.

AUTOMATES PROGRAMMABLES –

Partie 9: Interface de communication numérique point à point pour petits capteurs et actionneurs (SDCI)

1 Domaine d'application

La présente partie de l'IEC 61131 spécifie une technologie d'interface de communication numérique point à point pour petits capteurs et actionneurs – SDCI – (généralement connue sous l'appellation IO-Link™ [3]). Elle permet d'étendre les interfaces classiques d'entrées et de sorties numériques (telles que définies dans l'IEC 61131-2) en une liaison de communication point à point pour l'échange de données complexe dans les deux sens. Cette technologie permet également de transférer des paramètres aux Dispositifs et de fournir au système d'automatisation des informations de diagnostic et d'identification depuis les Dispositifs.

Cette technologie est principalement destinée à l'utilisation en automatisation d'usine avec de simples capteurs et actionneurs intégrant de petits microcontrôleurs économiques.

Le présent document spécifie les services et le protocole de communication SDCI (couche physique, couche de liaison de données et couche Application conformément au modèle de référence ISO/OSI) pour les Maîtres et Dispositifs SDCI.

Le présent document inclut également des exigences d'essai CEM.

Le présent document ne couvre pas les interfaces ou systèmes de communication comportant des connexions multipoints ou à liaisons multiples, ou encore l'intégration de SDCI dans des systèmes de niveau supérieur tels que les bus de terrain.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

2 Références normatives

Les documents suivants sont cités dans le texte de sorte qu'ils constituent, pour tout ou partie de leur contenu, des exigences du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 60664 (all parts), *Coordination de l'isolement des matériels dans les réseaux d'énergie électrique à basse tension*

IEC 60947-5-2, *Appareillage à basse tension – Partie 5-2: Appareils et éléments de commutation pour circuits de commande – Détecteurs de proximité*

IEC 61000-4-2, *Compatibilité électromagnétique (CEM) – Partie 4-2: Techniques d'essai et de mesure – Essai d'immunité aux décharges électrostatiques*

---

[3] IO-Link™ est une marque commerciale de "IO-Link Community". Cette information est fournie pour la commodité des utilisateurs du présent document et ne signifie nullement que l'IEC recommande l'emploi exclusif de la marque ou de l'un quelconque de ses produits. La conformité au présent document ne nécessite pas l'utilisation des logos déposés pour IO-Link™. L'utilisation des logos déposés pour IO-LinkTM nécessite l'autorisation du "IO-Link Community".

IEC 61000-4-3, *Compatibilité électromagnétique (CEM) – Partie 4-3: Techniques d'essai et de mesure – Essai d'immunité aux champs électromagnétiques rayonnés aux fréquences radioélectriques*

IEC 61000-4-4, *Compatibilité électromagnétique (CEM) – Partie 4-4: Techniques d'essai et de mesure – Essai d'immunité aux transitoires électriques rapides en salves*

IEC 61000-4-5, *Compatibilité électromagnétique (CEM) – Partie 4-5: Techniques d'essai et de mesure – Essai d'immunité aux ondes de choc*

IEC 61000-4-6, *Compatibilité électromagnétique (CEM) – Partie 4-6: Techniques d'essai et de mesure – Immunité aux perturbations conduites, induites par les champs radioélectriques*

IEC 61000-4-11, *Compatibilité électromagnétique (CEM) – Partie 4-11: Techniques d'essai et de mesure – Essais d'immunité aux creux de tension, coupures brèves et variations de tension pour les appareils à courant d'entrée inférieur ou égal à 16 A par phase*

IEC 61000-6-2, *Compatibilité électromagnétique (CEM) – Partie 6-2: Normes génériques – Norme d'immunité pour les environnements industriels*

IEC 61000-6-4, *Compatibilité électromagnétique (CEM) – Partie 6-4: Normes génériques – Norme sur l'émission pour les environnements industriels*

IEC 61076-2-101, *Connecteurs pour équipements électroniques – Exigences de produit – Partie 2-101: Connecteurs circulaires – Spécification particulière pour les connecteurs M12 à vis*

IEC 61010-2-201 *Exigences de sécurité pour appareils électriques de mesurage, de régulation et de laboratoire – Partie 2-201: Exigences particulières pour les équipements de commande*

IEC 61131-1, *Automates programmables – Partie 1: Informations générales*

IEC 61131-2, *Mesurage et contrôle des processus industriels – Automates programmables – Partie 2: Exigences et essais des équipements*

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3 Termes, définitions, symboles, termes abrégés et conventions

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions de l'IEC 61131-1 et l'IEC 61131-2 ainsi que les suivants s'appliquent.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse http://www.electropedia.org/
- ISO Online browsing platform: disponible à l'adresse http://www.iso.org/obp

3.1.1
adresse
partie d'une commande de séquence M utilisée pour référencer des données dans des catégories de données d'un canal de communication

3.1.2
couche d'application
AL
<SDCI> partie du protocole chargée de la transmission des objets de Données de Processus et des objets de Données à la Demande

Note 1 à l'article: Le terme abrégé "AL" est dérivé du terme anglais développé correspondant "*application layer*".

3.1.3
Paramètre de Bloc
Accès cohérent à des paramètres par l'intermédiaire de multiples Index ou Sous-index

3.1.4
somme de contrôle
<SDCI> partie complémentaire des mesures globales d'intégrité des données dans la couche de liaison de données en supplément au bit de parité de l'UART

3.1.5
CHKPDU
données de protection de l'intégrité dans un canal de communication ISDU générées par traitement XOR des octets d'une demande ou d'une réponse

3.1.6
commutation codée
communication SDCI, fondée sur les niveaux de signaux binaires normalisés de l'IEC 61131-2

3.1.7
COM1
mode de communication SDCI à une vitesse de transmission de 4,8 kbit/s

3.1.8
COM2
mode de communication SDCI à une vitesse de transmission de 38,4 kbit/s

3.1.9
COM3
mode de communication SDCI à une vitesse de transmission de 230,4 kbit/s

3.1.10
COMx
un des trois modes de communication SDCI possibles: COM1, COM2 ou COM3

3.1.11
canal de communication
connexion logique entre Maître et Dispositif

Note 1 à l'article: Quatre canaux de communication sont définis: un canal de processus, un canal de page et d'ISDU (pour les paramètres), et un canal de diagnostic.

3.1.12
erreur de communication
perturbation imprévue du protocole de transmission SDCI

3.1.13
durée de cycle
temps nécessaire à la transmission d'une séquence M entre un Maître et son Dispositif, y compris le temps de repos qui suit

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.14
Dispositif
homologue passif unique d'un Maître, tel qu'un capteur ou un actionneur

Note 1 à l'article: "Dispositif" avec un "D" majuscule désigne un équipement SDCI, tandis que "dispositif" avec un "d" minuscule est une désignation générique.

3.1.15
Paramètre Direct
paramètre à adressage direct (page) transféré de manière acyclique par l'intermédiaire du canal de communication de page sans acquittement

3.1.16
paramètre dynamique
partie d'un jeu de paramètres de Dispositif définie par des interfaces utilisateur embarquées, telles que des boutons d'apprentissage ou des panneaux de commande, et qui s'ajoutent aux paramètres statiques

3.1.17
Événement
instance d'un changement de conditions dans un Dispositif

Note 1 à l'article: "Événement" avec un "É" majuscule désigne un Événement SDCI, tandis que "événement" avec un "é" minuscule est utilisé de manière générique.

Note 2 à l'article: Un Événement est indiqué par le Drapeau (indicateur) d'Événement dans les informations cycliques d'état du Dispositif; le transfert acyclique des données d'Événement (en général des informations de diagnostic) est réalisé par l'intermédiaire du canal de communication de diagnostic.

3.1.18
repli
passage d'un Accès de la commutation codée au mode signal de commutation

3.1.19
niveau de vérification
degré de vérification de l'identité du Dispositif

3.1.20
entrelacement
échange de données cycliques segmentées pour des Données de Processus comportant plus de 2 octets au cours de cycles consécutifs

3.1.21
Unité de données de service indexée
ISDU
unité de données de service indexée utilisée pour la transmission acyclique avec acquittement de paramètres qui peuvent être segmentés en un certain nombre de séquences M

Note 1 à l'article: Le terme abrégé "ISDU" est dérivé du terme anglais développé correspondant "*indexed service data unit*".

3.1.22
Dispositif hérité
Dispositif conçu conformément à [8]

3.1.23
Maître hérité
Maître conçu conformément à [8]

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.24
séquence M
séquence de deux messages comprenant un message du Maître et le message du Dispositif résultant

3.1.25
commande de séquence M
premier octet dans un message du Maître indiquant l'opération de lecture/écriture, le type de canal de communication et l'adresse, par exemple décalage ou régulation de flux

3.1.26
erreur de séquence M
contenu de message imprévu ou erroné, ou absence de réponse

3.1.27
type de séquence M
format particulier de séquence M parmi un ensemble spécifié de formats de séquence M

3.1.28
Maître
homologue actif connecté par l'intermédiaire d'Accès à un jusqu'à n Dispositifs et qui fournit une interface avec la passerelle vers les systèmes de communication de niveau supérieur ou des automates programmables

Note 1 à l'article: "Maître" avec un "M" majuscule désigne un équipement SDCI, tandis que "maître" avec un "m" minuscule est une désignation générique.

3.1.29
message
<SDCI> séquence de trames UART transférées d'un Maître à son Dispositif ou réciproquement selon les règles du protocole SDCI

3.1.30
Données à la Demande
OD
données transmises de manière acyclique à la demande de l'application du Maître et constituées de paramètres ou de données d'Événement

Note 1 à l'article: Le terme abrégé "OD" est dérivé du terme anglais développé correspondant "*On-request Data*".

3.1.31
couche physique
première couche du modèle de référence ISO-OSI, qui fournit les moyens mécaniques, électriques, fonctionnels et procéduraux nécessaires à l'activation au maintien et à la désactivation des connexions physiques destinées à la transmission de bits entre entités de liaison de données

Note 1 à l'article: La couche physique fournit également des moyens de réactivation et procédures de repli.

[SOURCE: ISO/IEC 7498-1:1994 [5], 7.7.2, modifié – Texte extrait du 7.7.2, Note 1 à l'article ajoutée]

3.1.32
Accès
interface de support de communication du Maître vers un Dispositif

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.33
Données de Processus
PD
valeurs d'entrée ou de sortie, à partir de ou vers un procédé d'automatisation en continu ou par lot transférées de manière cyclique, hautement prioritaire et selon une planification configurée automatiquement entre un Maître et un Dispositif

Note 1 à l'article: Le terme abrégé "PD" est dérivé du terme anglais développé correspondant "*Process Data*".

3.1.34
cycle de Données de Processus
transfert complet de toutes les Données de Processus à partir de ou vers un Dispositif individuel qui peut inclure plusieurs cycles en cas de segmentation (entrelacement)

3.1.35
paramètre unique
Accès de paramètre indépendant par l'intermédiaire d'un Index ou Sous-index unique

3.1.36
SIO
mode de fonctionnement d'Accès conforme aux entrées et sorties numériques définies dans l'IEC 61131-2 qui est établi après une mise sous tension, un repli ou un échec de tentatives de communication

Note 1 à l'article: Le terme abrégé "SIO" est dérivé du terme anglais développé correspondant "*standard input output*".

3.1.37
paramètre statique
partie d'un jeu de paramètres de Dispositif à sauvegarder dans un Maître dans le cas d'un remplacement sans outillage technique

3.1.38
signal de commutation
signal binaire en provenance de ou vers un Dispositif, en mode SIO (par opposition à la communication SDCI à "commutation codée")

3.1.39
Gestion Système
SM
<SDCI> moyens de commande et de coordination des couches de communication internes et des exceptions dans le Maître et ses Accès, ainsi que dans chaque Dispositif

Note 1 à l'article: Le terme abrégé "SM" est dérivé du terme anglais développé correspondant "*system management*".

3.1.40
trame UART
<SDCI> séquence binaire commençant par un bit de départ, suivi de huit bits transportant un octet de données, suivi par un bit de parité paire et se terminant par un bit d'arrêt

3.1.41
réactivation
procédure permettant de faire passer un Dispositif du mode SIO en SDCI

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.1.42
demande de réactivation
WURQ
service de couche physique utilisé par le Maître pour lancer la réactivation d'un Dispositif, et le mettre en un état prêt à recevoir

Note 1 à l'article: Le terme abrégé "WURQ" est dérivé du terme anglais développé correspondant "*wake-up request*".

3.2 Symboles et termes abrégés

Δf_{DTRM}	écart admissible par rapport au débit de transfert de données (mesuré en %)
ΔVS	ondulation de l'alimentation (mesurée en V)
AL	application layer (couche d'application)
BEP	bit error probability (probabilité d'erreur sur les bits)
C/Q	connection for communication (C) or switching (Q) signal (SIO) (connexion pour signal (SIO) de communication (C) ou de commutation (Q))
CL_{eff}	capacité effective totale des câbles (mesurée en nF)
CQ	capacité d'entrée au niveau de la connexion C/Q (mesurée en nF)
DI	digital input (entrée numérique)
DL	data link layer (couche de liaison de données)
DO	digital output (sortie numérique)
f_{DTR}	data transfer rate (débit de transfert de données) (mesuré en bit/s)
H/L	high/low signal at receiver output (signal haut/bas à la sortie du récepteur)
E/S	entrée/sortie
ILL	input load current at input C/Q to $V0$ (courant de charge d'entrée au niveau de C/Q d'entrée vers $V0$) (mesuré en A)
IODD	IO Device Description (descriptif de dispositif d'E/S) (voir 10.9)
$IP24_M$	courant d'alimentation continu supplémentaire des Dispositifs
IQ	courant d'attaque à l'état de fonctionnement saturé ON (mesuré en A)
IQH	courant d'attaque du côté haut, à l'état de fonctionnement saturé ON (mesuré en A)
IQL	courant d'attaque du côté bas, à l'état de fonctionnement saturé ON (mesuré en A)
$IQPK$	courant d'attaque maximal, à l'état de fonctionnement non saturé ON (mesuré en A)
$IQPKH$	courant d'attaque maximal du côté haut, à l'état de fonctionnement non saturé ON (mesuré en A)
$IQPKL$	courant d'attaque maximal du côté bas, à l'état de fonctionnement non saturé ON (mesuré en A)
IQQ	courant passif au niveau de C/Q d'entrée vers $V0$ avec pilotes de sortie inactifs (mesuré en A)
IQ_{WU}	amplitude du courant de demande de réactivation du Maître (mesurée en A)
IS	courant d'alimentation au niveau de $V+$ (mesuré en A)
$ISIR$	capacité de fourniture d'impulsion de courant au niveau de $V+$ (mesurée en A)
LED	light emitting diode (diode électroluminescente)
L−	alimentation (−)
L+	alimentation (+)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

N24	alimentation 24 V supplémentaire (−)
NRZ	Non Retour à Zéro
n_{WU}	nombre de nouvelles tentatives de réactivation
On/Off	signal de commutation Actif/Non actif du circuit d'attaque
OD	On-request Data (Données à la Demande)
OVD	détection de surcharge du signal
P24	alimentation 24 V supplémentaire (+)
PD	Process Data (Données de Processus)
PDCT	port and Device configuration tool (outil de configuration des Accès et Dispositifs)
PL	physical layer (couche physique)
PLC	programmable logic controller (automate programmable)
PS	power supply (alimentation) (mesurée en V)
QIS_D	consommation de charge d'alimentation
r	temps nécessaire pour atteindre un niveau stable en référence au début du bit de départ (mesuré en T_{BIT})
RL_{eff}	résistance de boucle de câble (mesurée en Ω)
s	temps nécessaire pour quitter un niveau stable en référence au début du bit de départ (mesuré en T_{BIT})
SDCI	single-drop digital communication interface (interface de communication numérique point à point)
SIO	standard input output (digital switching mode) (entrée/sortie normalisée (mode de communication numérique)) [IEC 61131-2]
SM	system management (gestion système)
SMI	standardized Master interface (interface normalisée du Maître)
t_1	délai de transfert de trame UART sur le Maître (mesuré en T_{BIT})
t_2	délai de transfert de trame UART sur le Dispositif (mesuré en T_{BIT})
t_A	délai de réponse sur le Dispositif (mesuré en T_{BIT})
T_{BIT}	temps binaire (mesuré en s)
t_{CYC}	durée de cycle sur un niveau de séquence M (mesurée en s)
t_{DF}	temps de décroissance (mesuré en s)
T_{DMT}	temps de retard pendant l'établissement de la communication de l'Accès du Maître (mesuré en T_{BIT})
T_{DR}	temps de croissance (mesuré en s)
T_{DSIO}	temps de retard sur le dispositif pour le passage en mode SIO par suite d'une demande de réactivation (mesuré en s)
T_{DWU}	délai de nouvelle tentative de réactivation (mesurée en s)
$t_{M\text{-}sequence}$	durée de séquence M (mesurée en T_{BIT})
t_{idle}	temps au repos entre deux séquences M (mesuré en s)
t_H	temps de détection d'un niveau haut (mesuré en s)
t_L	temps de détection d'un niveau bas (mesuré en s)
t_{ND}	temps de suppression du bruit (mesuré en s)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

T_{RDL}	délai de l'état prêt à la réactivation après mise sous tension (mesuré en s)
T_{REN}	délai d'activation de la réception (mesuré en s)
T_{SD}	temps de détection du dispositif (mesuré en s)
T_{WU}	durée d'impulsion de demande de réactivation (mesurée en s)
UART	universal asynchronous receiver transmitter (émetteur-récepteur asynchrone universel)
UML	unified modelling language (langage de modélisation unifié) [série ISO/IEC 19505]
$V+$	tension en L+
$V0$	tension en L−
$VD+_L$	chute de tension sur la ligne entre les connexions L+ sur le Maître et le Dispositif (mesurée en V)
$VD0_L$	chute de tension sur la ligne entre les connexions L− sur le Maître et le Dispositif (mesurée en V)
VDQ_L	chute de tension sur la ligne entre les connexions C/Q sur le Maître et le Dispositif (mesurée en V)
$VHYS$	hystérésis de la tension de seuil du récepteur (mesurée en V)
VI	tension d'entrée au niveau de la connexion C/Q en référence à $V0$ (mesurée en V)
VIH	plage de tension d'entrée au niveau de la connexion C/Q pour un signal haut (mesurée en V)
VIL	plage de tension d'entrée au niveau de la connexion C/Q pour un signal bas (mesurée en V)
$VP24_M$	tension d'alimentation en courant continu supplémentaire des Dispositifs
VRQ	tension résiduelle sur le circuit d'attaque à l'état de fonctionnement saturé ON (mesurée en V)
$VRQH$	tension résiduelle sur le circuit d'attaque du côté haut, à l'état de fonctionnement ON (mesurée en V)
$VRQL$	tension résiduelle sur le circuit d'attaque du côté bas, à l'état de fonctionnement saturé ON (mesurée en V)
VTH	tension de seuil du récepteur en référence à $V0$ (mesurée en V)
$VTHH$	tension de seuil du récepteur pour une détection sûre d'un signal haut (mesurée en V)
$VTHL$	tension de seuil du récepteur pour une détection sûre d'un signal bas (mesurée en V)
WURQ	wake-up request pulse (impulsion de demande de réactivation)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.3 Conventions

3.3.1 Généralités

Le modèle de service, les primitives de service et les diagrammes présentés dans le présent document sont des descriptions entièrement abstraites. La mise en œuvre des services peut refléter des sujets individuels et peut différer en fonction de la situation.

3.3.2 Paramètres de service

Des primitives de service sont utilisées pour représenter des interactions entre fournisseur/consommateur de services (ISO/IEC 10731 [18]). Elles acheminent des paramètres qui définissent les informations disponibles dans l'interaction fournisseur/consommateur. Il

n'est pas nécessaire de déclarer explicitement chacun des paramètres d'une quelconque interface particulière.

La spécification des services dans le présent document utilise un format tabulaire pour décrire les paramètres constitutifs des primitives de service. Les paramètres applicables à chaque groupe de primitives de service sont présentés dans des tableaux. Chaque tableau est constitué de cinq colonnes, au plus:

1) nom du paramètre;
2) primitive de demande (.req);
3) primitive d'indication (.ind);
4) primitive de réponse (.rsp); et
5) primitive de confirmation (.cnf).

Un paramètre (ou élément constitutif de ce paramètre) est présenté dans chaque rangée de chaque tableau. Dans les colonnes des primitives de service concernées, un code permet de spécifier le type d'utilisation du paramètre sur la primitive spécifiée dans la colonne.

M Le paramètre est obligatoire pour la primitive.

U Le paramètre est une option utilisateur et peut ou ne peut pas être mis en œuvre en fonction de son utilisation dynamique par l'utilisateur du service. S'il n'est pas fourni, une valeur par défaut du paramètre est prise pour hypothèse.

C Le paramètre est conditionné par d'autres paramètres ou par l'environnement de l'utilisateur du service.

– Le paramètre n'est jamais présent.

S Le paramètre est un élément sélectionné.

Certaines entrées sont en outre qualifiées par un symbole entre parenthèses. Il peut s'agir:

a) d'une contrainte spécifique au paramètre; "(=)" indique que le paramètre est sémantiquement équivalent au paramètre dans la primitive de service qui se trouve immédiatement à sa gauche dans le tableau;

b) d'une indication selon laquelle une éventuelle note s'applique à l'entrée "(n)" et que la note "n" suivante contient des informations supplémentaires concernant le paramètre et son utilisation.

3.3.3 Procédures de services

Les procédures sont définies en matière:

– d'interactions entre entités d'application par échange d'unités de données de protocole; et
– d'interactions entre un fournisseur de services de couche de communication et un consommateur de services de couche de communication dans le même système par invocation de primitives de service.

Ces procédures s'appliquent à des instances de communication entre systèmes qui prennent en charge des services de communication comportant des contraintes de temps au sein des couches de communication.

3.3.4 Attributs de service

La nature des différents services (Maître et Dispositif) est caractérisée par des attributs. Tous les services sont définis du point de vue de la couche concernée vers la couche au-dessus.

I Initiateur d'un service (vers la couche au-dessus)

R Destinataire (répondeur) d'un service (à partir de la couche au-dessus)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

3.3.5 Figures

Les conventions suivantes sont utilisées dans les figures qui représentent la structure et les services de couches de protocole:

– une flèche avec seulement un nom de service représente à la fois une demande et la confirmation correspondante, la demande étant effectuée dans le sens de la flèche,

– une demande sans confirmation, ainsi que toutes les indications et réponses sont libellées en tant que telles (c'est-à-dire service.req, service.ind, service.rsp).

La Figure 1 donne un exemple de service confirmé.

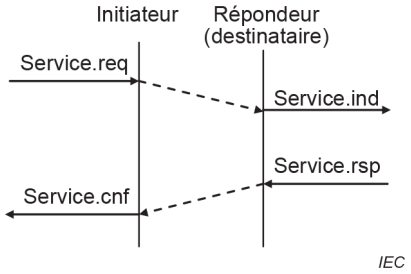


Figure 1 – Exemple de service confirmé

3.3.6 Tableaux

Dans le présent document, certains tableaux présentent des zones surlignées (en gris) afin d'accentuer le contenu de l'information (structure) et afin de signaler les contenus réservés ou spéciaux.

3.3.7 Ordre de transmission des octets

La Figure 2 représente la manière dont des types de données à base d'Accès sont transférés de la mémoire au support de transmission et réciproquement (c'est-à-dire l'octet de poids fort transmis en premier, voir 7.3.3.2 et 7.3.6.1).

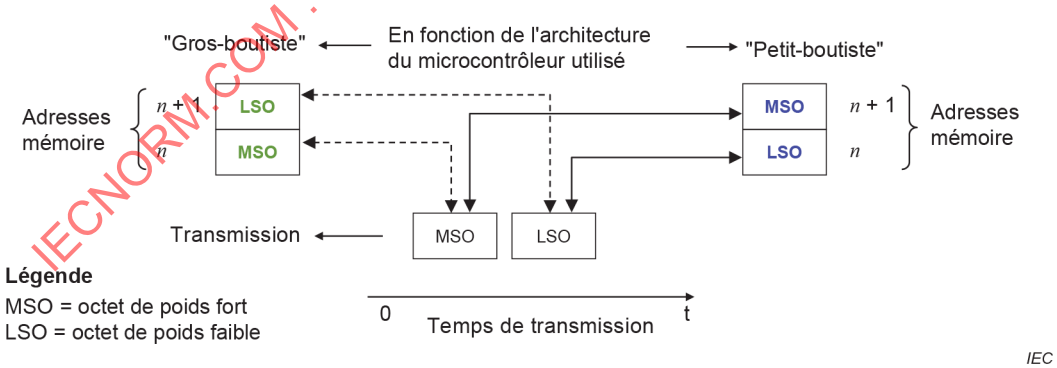


Figure 2 – Ordre de stockage en mémoire et de transmission pour des types de données à base de MOTS

3.3.8 Descriptifs comportementaux

Les descriptifs comportementaux utilisent les notations du langage UML 2 (série ISO/IEC 19505 [19]) (par exemple, des diagrammes d'états, de séquences, d'activités, des chronogrammes, des conditions de protection).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Les diagrammes d'états sont la principale source de mises en œuvre, alors que les diagrammes de séquences présentent certains cas d'utilisation. Les caractéristiques des diagrammes d'états sont les suivantes

– déclencheurs/événements provenant de demandes externes ("invocations") ou modifications internes (des délais d'expiration, par exemple),
– [guard(s)] en tant qu'expressions booléennes pour les sorties d'états,
– transitions numérotées décrivant les actions en plus des déclencheurs à l'intérieur des tables de transition d'état séparés.

La présentation de ces tableaux est conforme à l'IEC TR 62390 [14].

Dans le présent document, le concept "d'états imbriqués" avec des super-états et des sous-états est utilisé comme cela est indiqué dans l'exemple de la Figure 3.

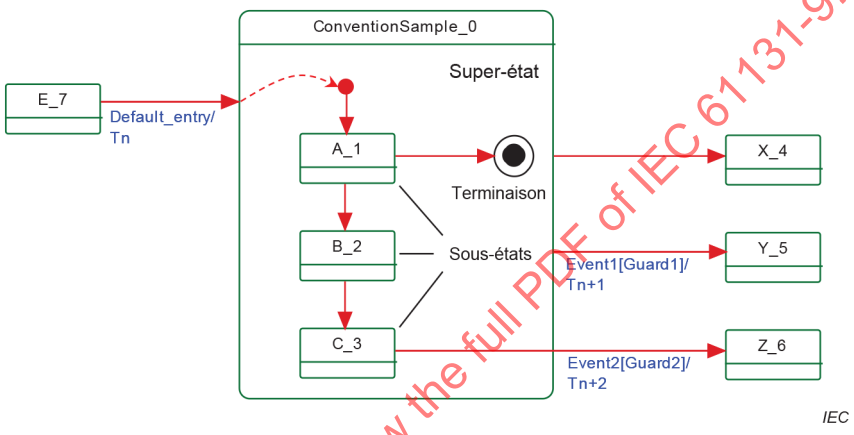


Figure 3 – Exemple d'état imbriqué

UML 2 permet de hiérarchiser les états avec des super-états et des sous-états. Le super-état le plus élevé représente l'ensemble de l'automate.

Ce concept permet de simplifier la modélisation, le contenu des super-états pouvant être déplacé vers un schéma séparé. Une icône en forme de lunettes représente en général ce contenu.

Par rapport à des automates "plats", un ensemble particulier de règles doit être respecté pour les "états imbriqués":

a) une transition vers le bord d'un super-état (Default_entry, par exemple) implique une transition vers le sous-état initial (A_1, par exemple);
b) la transition vers un état de terminaison à l'intérieur d'un super-état implique une transition sans événement et protection vers un état extérieur (X_4, par exemple). Le super-état devient inactif;
c) une transition entre l'un des sous-états (A_1, B_2 ou C_3, par exemple) vers un état à l'extérieur (Y_5) peut se produire chaque fois que Event1 se produit et que Guard1 est "true" (vrai). Cela est utile en cas d'erreurs communes à l'intérieur des sous-états. Le super-état devient inactif;
d) une transition entre un sous-état particulier (C_3, par exemple) vers un état à l'extérieur (Z_6) peut se produire chaque fois que Event2 se produit et que Guard2 est "true" (vrai). Le super-état devient inactif.

En raison des restrictions liées à l'outil de conception UML, les exceptions suivantes s'appliquent.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Pour les diagrammes d'états, un paramètre de service (en lettres majuscules) est relié au nom de service par un caractère de soulignement, par exemple dans DL_SetMode_INACTIVE.

Pour les diagrammes de séquences, la primitive de service est reliée par un caractère de soulignement au lieu d'un point, et le paramètre de service est ajouté entre parenthèses, par exemple dans DL_Event_ind (OPERATE).

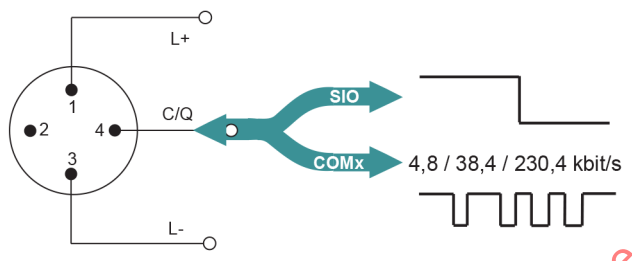
Les contraintes temporelles sont libellées "tm(time in ms)".

Les invocations de services reçues en asynchrone ne sont pas modélisées en détail dans les diagrammes d'états.

4 Vue d'ensemble de l'interface SDCI (IO-Link™)

4.1 Objectif de la technologie

La Figure 4 représente le concept de base de l'interface SDCI.



Broche	Signal	Définition	Norme
1	L+	24 V	IEC 61131-2
2	I/Q	Non connecté, DI ou DO	IEC 61131-2
3	L-	0 V	IEC 61131-2
4	Q	"Signal de commutation" (SIO)	IEC 61131-2
	C	"Commutation codée" (COM1, COM2, COM3)	IEC 61131-9

IEC

Figure 4 – Compatibilité de la SDCI avec l'IEC 61131-2

La technologie d'interface de communication numérique point à point pour petits capteurs et actionneurs – SDCI – (généralement connue sous l'appellation IO-Link™) définit une voie de migration, à partir des interfaces existantes d'entrées et de sorties numériques pour des Dispositifs à commutation 24 V telles que définies dans l'IEC 61131-2, permettant d'obtenir une liaison de communication point à point. Ainsi, les modules d'E/S numériques des périphériques de bus de terrain existants, par exemple, peuvent être remplacés par des modules Maître SDCI fournissant à la fois des interfaces DI/DO classiques et l'interface SDCI. La technologie de transmission analogique peut être remplacée par la technologie SDCI qui associe sa robustesse ainsi que ses fonctionnalités de paramétrage et de diagnostic à l'économie d'efforts de conversion numérique/analogique et analogique/numérique.

4.2 Positionnement dans la hiérarchie de l'automatisation

La Figure 5 représente le domaine de la technologie SDCI au sein de la hiérarchie de l'automatisation.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

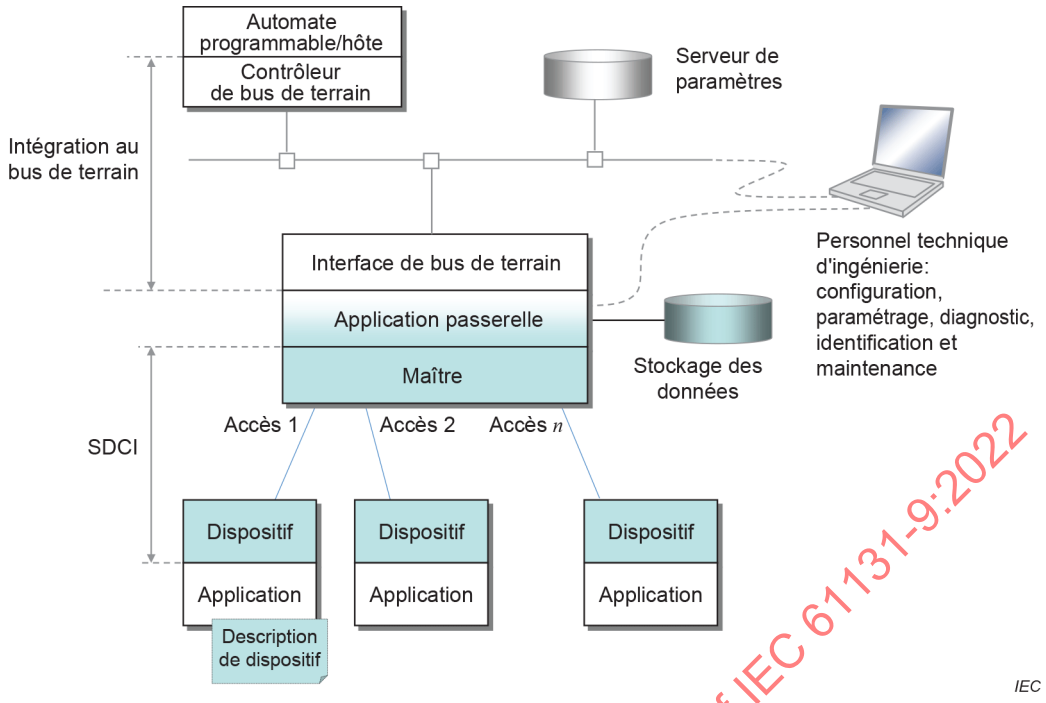


Figure 5 – Domaine de la technologie SDCI au sein de la hiérarchie de l'automatisation

La technologie SDCI définit une interface générique pour la connexion de capteurs et d'actionneurs à une unité Maître, qui peut être associée à des fonctionnalités de passerelle pour constituer un nœud déporté d'E/S de bus de terrain.

Le point de départ de la conception de l'interface SDCI est l'entrée numérique (DI) 24 V conventionnelle définie dans l'IEC 61131-2 et l'interface de sortie (DO) spécifiée dans le Tableau 6. L'interface SDCI offre ainsi la connectivité de capteurs 24 V conventionnels ("signaux de commutation") comme mode de fonctionnement par défaut. Une connectivité supplémentaire est également fournie pour des actionneurs si un Accès est configuré en "mode de communication point à point".

De nos jours, de nombreux capteurs et actionneurs sont déjà équipés de microcontrôleurs proposant une interface UART qui peut être étendue par ajout de quelques composants matériels et d'un logiciel protocolaire prenant en charge la communication SDCI. Ce second mode de fonctionnement utilise la "commutation codée" de ligne de signal d'E/S 24 V. Dès son activation, le mode SDCI prend en charge le paramétrage, l'échange des données cycliques, les rapports de diagnostic, les informations d'identification et de maintenance, ainsi que le stockage externe des paramètres pour la sauvegarde des réglages des Dispositifs et un rechargement rapide des dispositifs de remplacement. Les capteurs et actionneurs ayant une capacité SDCI sont appelés "Dispositifs" dans le présent document. Pour améliorer les performances au démarrage, ces Dispositifs comportent généralement une mémoire non volatile de stockage des paramètres.

NOTE La configuration et le paramétrage des Dispositifs sont pris en charge par un descriptif de dispositifs fondé sur le langage XML (voir [6]), non couvert par le présent document.

4.3 Câblage, connecteurs et alimentation

La connexion par défaut (Classe d'Accès A) comporte quatre broches (voir la Figure 4). Le câblage par défaut pour la Classe d'Accès A est conforme à l'IEC 60947-5-2 et n'utilise que trois conducteurs: 24 V, 0 V et une ligne de signal. Le quatrième conducteur peut être utilisé comme ligne de signal supplémentaire conformément à l'IEC 61131-2.

Des connexions à cinq broches (Classe d'Accès B) sont spécifiées pour des Dispositifs exigeant une puissance supplémentaire fournie à partir d'une alimentation 24 V indépendante.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

NOTE Un Dispositif de Classe d'Accès A utilisant le quatrième conducteur est incompatible avec un Maître de Classe d'Accès B.

La longueur maximale des câbles est de 20 m; il n'est pas exigé de blindage.

4.4 Fonctionnalités de communication de l'interface SDCI

Le modèle de Dispositif générique est représenté à la Figure 6 et expliqué dans le paragraphe suivant.

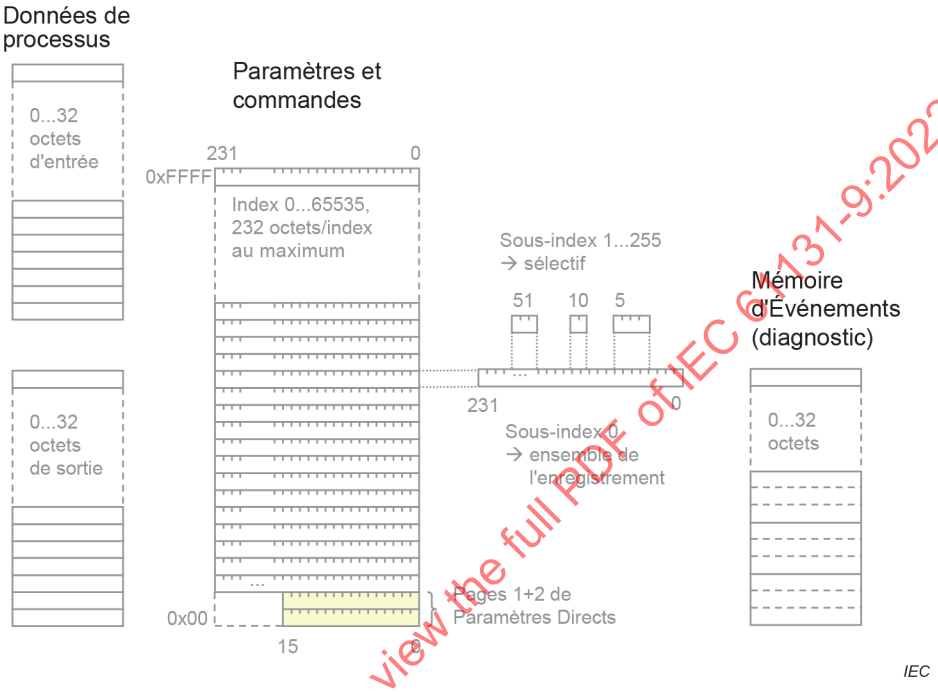


Figure 6 – Modèle de Dispositif générique pour SDCI (vue du côté Maître)

Un Dispositif peut recevoir des Données de Processus (de sortie) pour commander un processus d'automatisation par lot ou en continu ou envoyer des Données de Processus (d'entrée) représentant son état courant ou des valeurs de mesure. Le Dispositif fournit généralement des paramètres permettant à l'utilisateur de configurer ses fonctions et répondre ainsi à des besoins particuliers. Pour tenir compte de telles circonstances, il est défini un vaste espace de paramétrage auquel il est possible d'accéder par l'intermédiaire d'un Index (0 à 65 535; à structure prédéfinie) et d'un Sous-index (0 à 255).

Les deux premières entrées de l'index, 0 et 1, sont réservées aux pages 1 et 2 des Paramètres Directs, de 16 octets chacune, au plus. La page de paramètres 1 est principalement spécifique aux commandes du Maître telles que démarrage et repli du Dispositif, récupération d'informations de fonctionnement et d'identification spécifiques au Dispositif. La page de paramètres 2 prévoit un maximum de 16 octets de paramètres spécifiques au Dispositif.

Chacun des autres index (2 à 65 535) permet d'accéder à un enregistrement d'une taille maximale de 232 octets. Le sous-index 0 spécifie la transmission de l'enregistrement complet indiqué par l'Index et les autres sous-index spécifient le transfert d'éléments de données sélectionnés dans l'enregistrement.

Dans un enregistrement, les éléments de données individuels peuvent commencer à partir de n'importe quel décalage binaire, et leur longueur peut aller de 1 bit à 232 octets; cependant, le nombre total d'éléments de données dans l'enregistrement ne peut pas dépasser 255. L'organisation des éléments de données dans un enregistrement est spécifiée dans le Descriptif de Dispositif d'E/S (IODD).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Toute modification d'une condition de Dispositif qui exige un rapport ou une intervention est stockée dans une mémoire d'Événements avant transmission. Un drapeau d'Événement est ensuite établi dans l'échange des données cycliques pour indiquer l'existence d'un Événement.

La communication entre un Maître et un Dispositif est effectuée en point à point et se fonde sur le principe de l'envoi d'un message de demande par un Maître, puis l'envoi d'un message de réponse par un Dispositif (voir la Figure 38). Pris ensemble, ces messages sont appelés une Séquence M. Plusieurs types de Séquence M sont définis pour prendre en charge les exigences de l'utilisateur en matière de transmission de données (voir la Figure 39).

Comme cela est représenté à la Figure 7, plusieurs catégories de données sont transmises par des canaux de communication individuels de la couche de liaison de données.

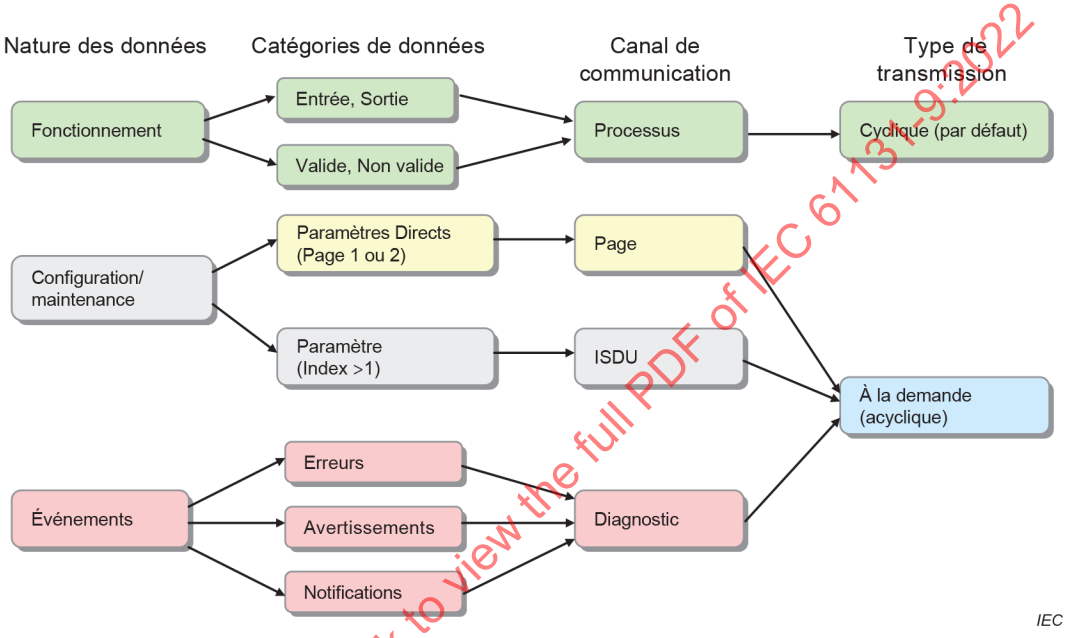


Figure 7 – Relation entre nature des données et types de transmission

– Les données opérationnelles telles que les entrées et sorties de Dispositif sont transmises de manière cyclique par l'intermédiaire d'un canal de processus. Les données opérationnelles peuvent également être associées à des qualificatifs tels que valide/non valide.
– Les paramètres de configuration et de maintenance sont transmis de manière acyclique. Un canal de page est prévu pour accéder directement aux pages de paramètres 1 et 2, et un canal ISDU est utilisé pour accéder à des paramètres et commandes supplémentaires.
– Les Événements de Dispositifs sont transmis de manière acyclique par l'intermédiaire d'un canal de diagnostic. Les rapports d'Événements de Dispositifs comportent trois niveaux de sévérité: erreur, avertissement et notification.

Le premier octet d'un message de Maître commande le sens de transfert des données (lecture/écriture) et le type de canal de communication.

La Figure 8 indique que chaque Accès d'un Maître comporte sa propre couche de liaison de données en interface avec une couche commune d'application du Maître. Dans la couche d'application, les services de la couche de liaison de données sont traduits en actions sur les objets de Données de Processus (entrée/sortie), les objets de Données à la Demande (lecture/écriture), et les Événements. Les applications du Maître comprennent le Gestionnaire de Configuration (CM - Configuration Manager), le mécanisme de Stockage des Données (DS - Data Storage), l'Unité de Diagnostic (DU - Diagnosis Unit), l'Échange de Données à la Demande (ODE - On-request Data Exchange) et l'Échange de Données de Processus (PDE - Process Data Exchange).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

La Gestion Système vérifie l'identification des Dispositifs connectés et règle les Accès et les Dispositifs pour qu'ils correspondent à la configuration choisie et aux propriétés de ces mêmes Dispositifs connectés. Elle gère également, au démarrage par exemple, les diagrammes d'états des couches d'application (AL) et de liaison de données (DL).

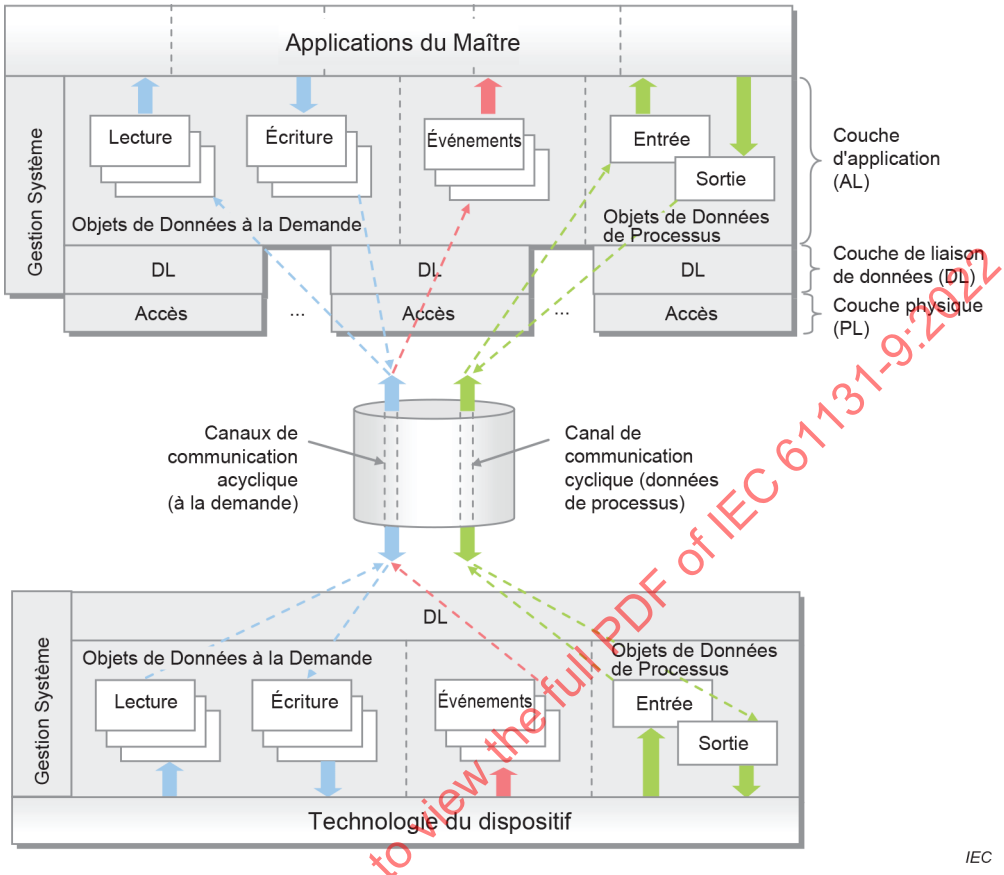


Figure 8 – Transfert d'objets au niveau de la couche d'application (AL)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

4.5 Rôle d'un Maître

Un Maître prend en charge 1 à n Accès et les couches de liaison de données correspondantes. Au démarrage, il modifie les Accès en fonction des modes sélectionnés par l'utilisateur: DEACTIVATED, IOL_MANUAL, IOL_AUTOSTART, DI_C/Q ou DO_C/Q. S'il y a demande de communication, le Maître utilise une impulsion de courant de réactivation spéciale pour lancer la communication avec le Dispositif. Ensuite, le Maître règle automatiquement la vitesse de transmission sur COM1, COM2 ou COM3 (voir le Tableau 9) et vérifie la "personnalité" du Dispositif connecté, c'est-à-dire son VendorID (identifiant de fournisseur), son DeviceID (identifiant de Dispositif), et ses propriétés de communication.

En cas de discordance entre les paramètres du Dispositif et le jeu de paramètres enregistré dans le Maître, les paramètres dans le Dispositif sont écrasés (voir 11.4) ou, en fonction de la configuration, les paramètres enregistrés dans le Maître sont mis à jour.

Le Maître est chargé d'assembler et de désassembler toutes les données échangées avec les Dispositifs (voir l'Article 11).

Le Maître fournit une zone Stockage des Données d'au moins 2 048 octets par Dispositif pour sauvegarde des données de Dispositifs (voir 11.4). Pour ses propres besoins opérationnels, le Maître peut combiner ces données de Dispositif avec toutes autres données pertinentes et les mettre à la disposition d'applications de niveau supérieur à des fins de sauvegarde du Maître ou de contrôle des recettes (voir 13.4.2).

4.6 Configuration SDCI

La prise en charge technique d'un Maître est généralement assurée par un Outil de Configuration des Accès et Dispositifs (PDCT). Le PDCT permet de configurer les propriétés des Accès et celles des Dispositifs (voir les paramètres représentés à la Figure 6). Cet outil associe un interpréteur de Descriptifs de Dispositifs d'E/S (IODD) et un configurateur (voir l'Article 13). L'IODD fournit l'ensemble des propriétés nécessaires à l'établissement de la communication ainsi que les paramètres et limites correspondantes nécessaires à l'établissement de la fonction souhaitée d'un capteur ou d'un actionneur. Le PDCT permet également de compiler les Données de Processus pour propagation sur le bus de terrain et inversement.

4.7 Mise en correspondance avec les bus de terrain

Les spécifications d'intégration des Maîtres dans des systèmes de niveau supérieur (un système de bus de terrain, par exemple), c'est-à-dire la définition des fonctions de passerelle d'échange de données avec des entités de niveau supérieur sur un bus de terrain, ne relèvent pas du domaine d'application du présent document. Toutefois, il est obligatoire de mettre toutes les fonctions du présent document à la disposition des utilisateurs dans le cadre d'une intégration particulière en fonction du niveau de capacité de la technologie du système de niveau supérieur, à l'exception des fonctions explicitement déclarées comme étant facultatives.

EXEMPLE Ces fonctions comprennent la mise en correspondance des Données de Processus échangées, le paramétrage contrôlé par programme ou la mise en œuvre d'un serveur de paramètres distant, ou encore la propagation des informations de diagnostic.

L'intégration d'un PDCT dans des outils techniques d'ingénierie d'un bus de terrain particulier ou d'un autre système de niveau supérieur ne relève pas du domaine d'application du présent document.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

4.8 Structure de la norme

La structure logique du Maître et du Dispositif est représentée à la Figure 9. L'Article 5 définit la Couche Physique (PL) de la SDCI, l'Article 6 décrit de manière détaillée le mode SIO. L'Article 7 définit les services et le protocole de la Couche de Liaison de Données (DL), la réactivation, les Séquences M et les modules de traitement de couche DL. L'Article 8 définit les services et le protocole de la Couche d'Application (AL) et l'Article 9 précise les responsabilités liées à la Gestion Système (SM).

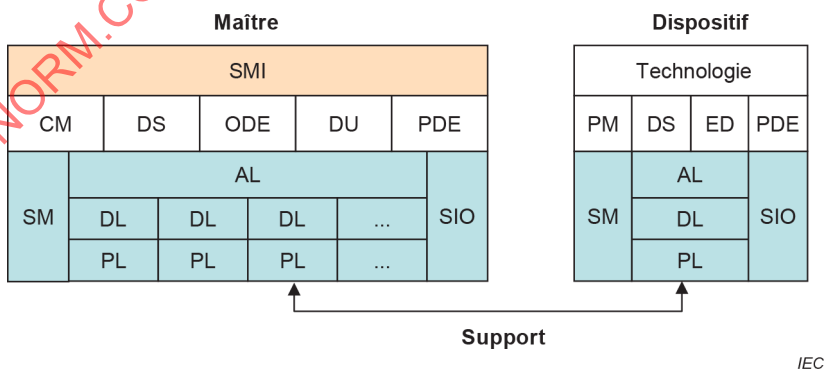


Figure 9 – Structure logique du Maître et du Dispositif

L'Article 10 spécifie les applications et fonctionnalités des Dispositifs, telles que l'Échange de Données de Processus (PDE), la Gestion des Paramètres (PM), le Stockage des Données (DS), et le Répartiteur d'Événements (ED). Les applications de Dispositif spécifiques à la technologie ne sont pas traitées dans le présent document. Elles peuvent être spécifiées dans des profils de familles de Dispositifs particulières.

L'Article 11 spécifie les applications et fonctionnalités du Maître, telles que l'Échange de Données de Processus (PDE), l'Échange de Données à la Demande (ODE), la Gestion de la Configuration (CM), le Stockage des Données (DS) et l'Unité de Diagnostic (DU). Une interface normalisée du Maître (SMI) assure le comportement uniforme au moyen des services spécifiés et permet d'utiliser un PDCT (outil Maître) pour différentes marques Maître.

L'Article 12 donne un aperçu des bonnes pratiques holistiques sur le comportement de stockage des données du Maître et du Dispositif.

L'Article 13 présente les aspects relatifs à l'intégration du SDCI dans différents royaumes d'automatisation et IT.

Plusieurs annexes normatives et informatives sont incluses. L'Annexe A définit les types de Séquences M disponibles. L'Annexe B décrit les paramètres de la page de Paramètres Directs et les paramètres fixes des Dispositifs. L'Annexe C récapitule les types d'erreurs en transmission acyclique et l'Annexe D énumère les EventCodes (codes d'Événements) (informations de diagnostic des Dispositifs). L'Annexe E spécifie le codage des blocs d'arguments pour les services SMI. L'Annexe F spécifie les types de données de base et composites disponibles. L'Annexe G définit la structure des objets Stockage des données. L'Annexe H traite de la conformité et des exigences d'essai de compatibilité électromagnétique, et l'Annexe I fournit des graphiques de probabilités d'erreur résiduelle, démontrant le niveau d'intégrité des données de SDCI. L'Annexe J informative donne un exemple de séquence de transmission de données acycliques. L'Annexe K informative explique deux méthodes recommandées de détection des modifications de paramètres dans le contexte du Stockage des Données.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

5 Couche Physique (PL)

5.1 Généralités

5.1.1 Principes de base

Le système de connexion à 3 conducteurs de l'interface SDCI est fondé sur les spécifications de l'IEC 60947-5-2. Les trois fils sont utilisés comme suit: (L+) pour l'alimentation 24 V, (L−) pour la masse, et (C/Q) pour le signal de commutation (Q) ou la communication SDCI (C), comme cela est représenté à la Figure 10.

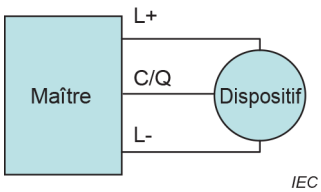


Figure 10 – Système de connexion à trois conducteurs

NOTE 1 Les capteurs binaires conformes à l'IEC 60947-5-2 sont compatibles avec le système de connexion à 3 conducteurs SDCI (y compris du point de vue de la consommation d'énergie).

La prise en charge du système de connexion à 3 conducteurs SDCI est obligatoire pour le Maître. Les Accès présentant cette caractéristique sont appelés Accès de classe A.

La Classe d'Accès A utilise un connecteur à quatre broches. Le quatrième conducteur peut être utilisé comme ligne de signal supplémentaire conformément à l'IEC 61131-2. Sa prise en charge est facultative tant pour les Maîtres que pour les Dispositifs.

Des connexions à cinq conducteurs (Classe d'Accès B) sont spécifiées pour des Dispositifs nécessitant une puissance supplémentaire fournie à partir d'une alimentation 24 V indépendante (voir 5.5.1).

NOTE 2 Un dispositif de Classe d'Accès A utilisant le quatrième conducteur est incompatible avec un Maître de Classe d'Accès B.

5.1.2 Topologie

La topologie du système SDCI utilise des liaisons point à point entre un Maître et ses Dispositifs, comme cela est représenté à la Figure 11. Le Maître peut avoir plusieurs Accès de connexion des Dispositifs. Un seul Dispositif doit être connecté à chaque Accès.

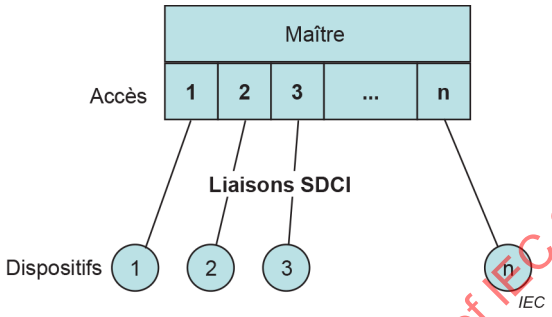


Figure 11 – Topologie de l'interface SDCI

5.2 Services de Couche Physique

5.2.1 Vue d'ensemble

La Figure 12 donne une vue d'ensemble de la couche physique du Maître et de ses primitives de service.

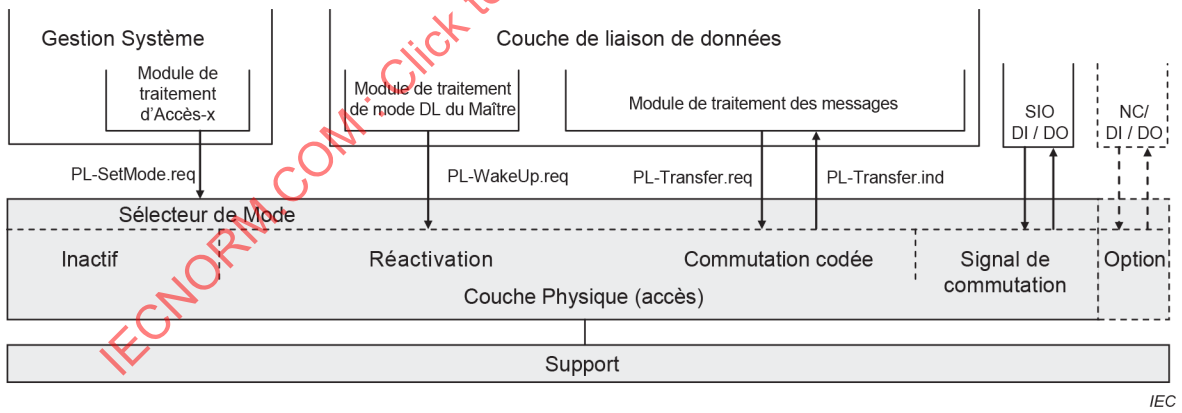


Figure 12 – Couche physique (Maître)

La couche physique définit le fonctionnement de la ligne C/Q à la Figure 4, ainsi que le circuit d'attaque de la ligne (émetteur) et le récepteur correspondants d'un Accès particulier. Le Maître exploite cette ligne selon trois principaux modes (voir Figure 12): Inactif, "Signal de commutation" (DI/DO) ou "Commutation codée" (COMx). Le service PL-SetMode.req est chargé du passage dans l'un de ces modes.

Si l'Accès est en mode inactif, la ligne C/Q doit avoir une haute impédance (flottante). En mode SIO, l'Accès peut être utilisé comme une interface d'entrée ou de sortie normalisée conformément aux définitions de l'IEC 61131-2 ou comme présenté dans le Tableau 6 respectivement. Les couches de communication de l'interface SDCI sont contournées comme cela est représenté à la Figure 12; les signaux sont directement traités au sein de l'application

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

du Maître. En mode SDCI, le service PL_WakeUp.req génère une configuration de signal spéciale (impulsion de courant) qui peut être détectée par un Dispositif conforme SDCI connecté à cet Accès (voir 5.3.3.3).

La Figure 13 donne une vue d'ensemble de la couche physique du Dispositif et de ses primitives de service.

La couche physique d'un Dispositif conforme à la Figure 13 respecte le même principe, sauf qu'il n'y a pas d'état inactif. Par défaut, à la mise sous tension ou lors du rebranchement du câble, le Dispositif doit fonctionner en mode SIO, comme une entrée numérique (du point de vue du Maître). Le Dispositif doit toujours pouvoir détecter une impulsion de courant de réactivation (demande de réactivation) excepté lors d'un état inactif permanent. Le service PL_WakeUp.ind rend compte d'une détection réussie de la demande de réactivation (généralement une interruption du microcontrôleur), qui est la condition exigée pour que le Dispositif passe en mode SDCI.

Une Commande du Maître MasterCommand spéciale (repli) envoyée par l'intermédiaire de l'interface SDCI ramène le Dispositif en mode SIO.

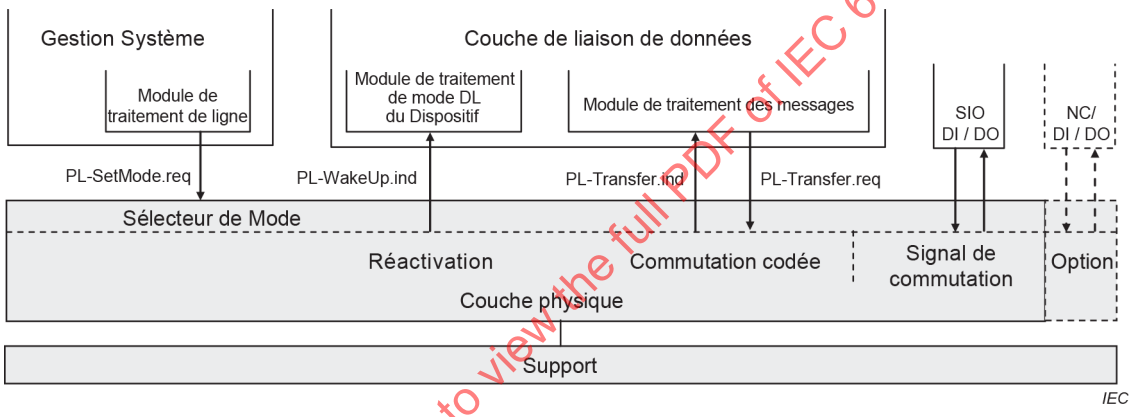


Figure 13 – Couche physique (Dispositif)

Par la suite, les services spécifiés sont ceux fournis par la PL à la Gestion Système et à la Couche de Liaison de Données (voir la Figure 85 et la Figure 96 pour une vue d'ensemble exhaustive de tous les services). Le Tableau 1 énumère les attributions du Maître et du Dispositif quant à leur rôle d'initiateur ou de destinataire des services individuels de couche physique (PL).

Tableau 1 – Attributions des services du Maître et du Dispositif

Désignation du service	Maître	Dispositif
PL-SetMode	R	R
PL-WakeUp	R	I
PL-Transfer	I / R	R / I
Légende (voir 3.3.4) I Initiateur du service R Destinataire (Répondeur) du service		

5.2.2 Services de PL

5.2.2.1 PL_SetMode

Le service PL-SetMode est utilisé pour établir les caractéristiques électriques et les configurations de la Couche Physique. Les paramètres des primitives de service sont énumérés dans le Tableau 2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 2 – PL_SetMode

Nom du paramètre	.req
Argument TargetMode	M M

Argument
Pour la demande de service, les paramètres spécifiques au service sont transmis dans l'argument.

TargetMode
Ce paramètre indique le mode de fonctionnement demandé.

Valeurs admises:
INACTIVE (impédance haute de la ligne C/Q),
DI (ligne C/Q en mode d'entrée numérique),
DO (ligne C/Q en mode de sortie numérique),
COM1 (ligne C/Q en mode COM1),
COM2 (ligne C/Q en mode COM2),
COM3 (ligne C/Q en mode COM3).

5.2.2.2 PL_WakeUp

Le service PL-WakeUp lance ou indique une séquence spécifique qui prépare la Couche Physique à envoyer et à recevoir des demandes de communication (voir 5.3.3.3). Ce service non confirmé n'a pas de paramètres. Son aboutissement avec succès ne peut être vérifié que par un Maître lors d'une tentative de communication avec le Dispositif. Les primitives de service sont énumérées dans le Tableau 3.

Tableau 3 – PL_WakeUp

Nom du paramètre	.req	.ind
<aucun>		

5.2.2.3 PL_Transfer

Le service PL-Transfer est utilisé pour l'échange de données SDCI entre la Couche de Liaison de Données et la Couche Physique. Les paramètres des primitives de service sont énumérés dans le Tableau 4.

Tableau 4 – PL_Transfer

Nom du paramètre	.req	ind.
Argument Données	M	M
Résultat (+)		S
Résultat (−) État		S M

Argument
Pour la demande de service, les paramètres spécifiques au service sont transmis dans l'argument.

Données

Ce paramètre contient la valeur des données transférées sur l'interface SDCI.

Valeurs admises: 0…255.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

État

Ce paramètre contient des informations supplémentaires concernant l'état du transfert.

Valeurs admises:
PARITY_ERROR (l'UART a détecté une erreur de parité),
FRAMING_ERROR (détection de bit d'arrêt UART non valide),
OVERRUN (collision d'octets dans l'UART).

5.3 Émetteur/Récepteur

5.3.1 Méthode de description

La couche physique est spécifiée au moyen d'exigences électriques et temporelles. Les exigences électriques définissent séparément des niveaux de signaux et de courants pour le Maître et le Dispositif sous la forme de schémas de référence. Les exigences temporelles définissent le processus de transmission des signaux (spécifiquement pour le récepteur) et une fonction spéciale de détection des signaux.

5.3.2 Exigences électriques

5.3.2.1 Généralités

Le circuit d'attaque de la ligne est défini par un schéma de référence correspondant à la Figure 14. Du côté du Maître, un émetteur comprend une combinaison de deux circuits d'attaque et d'un collecteur de courant. Du côté du Dispositif, sous sa forme la plus simple, l'émetteur se présente sous la forme d'un circuit d'attaque à commutation p. Il peut y avoir en option un circuit d'attaque supplémentaire à commutation n ou sans commutation (ceci permet également d'opter pour un fonctionnement en configuration de sortie symétrique).

À l'état opérationnel ON (Actif), les variables descriptives sont la tension résiduelle VRQ, le courant d'attaque normalisé IQ et le courant de crête $IQPK$. La source est commandée par le signal On/Off (Actif/Non actif). Un événement de courant de surcharge est indiqué à la sortie "Surcharge" (OVD). Cette fonctionnalité peut être utilisée pour la détection de l'impulsion de courant (de réactivation).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

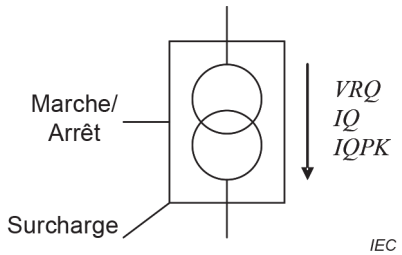


Figure 14 – Schéma de référence du circuit d'attaque de la ligne

Le récepteur est défini par un schéma de référence comme cela est représenté à la Figure 15. Il assure la fonction d'un comparateur et il est spécifié par ses seuils de commutation VTH et une hystérésis $VHYS$ entre les seuils de commutation. La sortie indique le niveau logique (Haut ou Bas (H/L)) à l'entrée du récepteur.

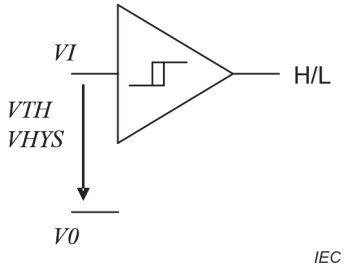
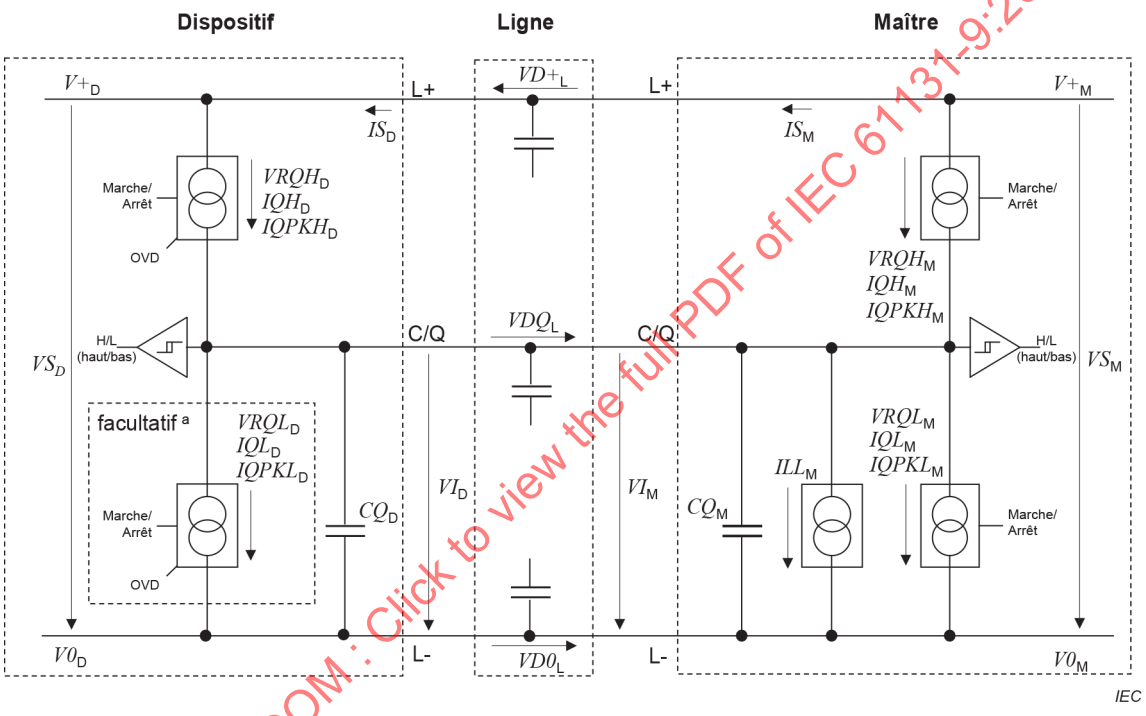


Figure 15 – Schéma de référence du récepteur

La Figure 16 représente le schéma de référence de l'interconnexion du Maître et du Dispositif pour le système de connexion à 3 conducteurs SDCI.



[a] Facultatif: circuit d'attaque côté bas (en configuration symétrique uniquement)

Figure 16 – Schéma de référence pour le système de connexion à 3 conducteurs SDCI

Les représentations graphiques et tableaux de paramètres suivants font référence aux définitions des niveaux de tension donnés à la Figure 17. Les index des paramètres font référence au Maître (M), au Dispositif (D) ou à la ligne (L). Les chutes de tension sur $VD{+}_{L}$, VDQ_{L} et $VD0_{L}$ de la ligne sont implicitement spécifiées en 5.5 par le biais des paramètres du câblage.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

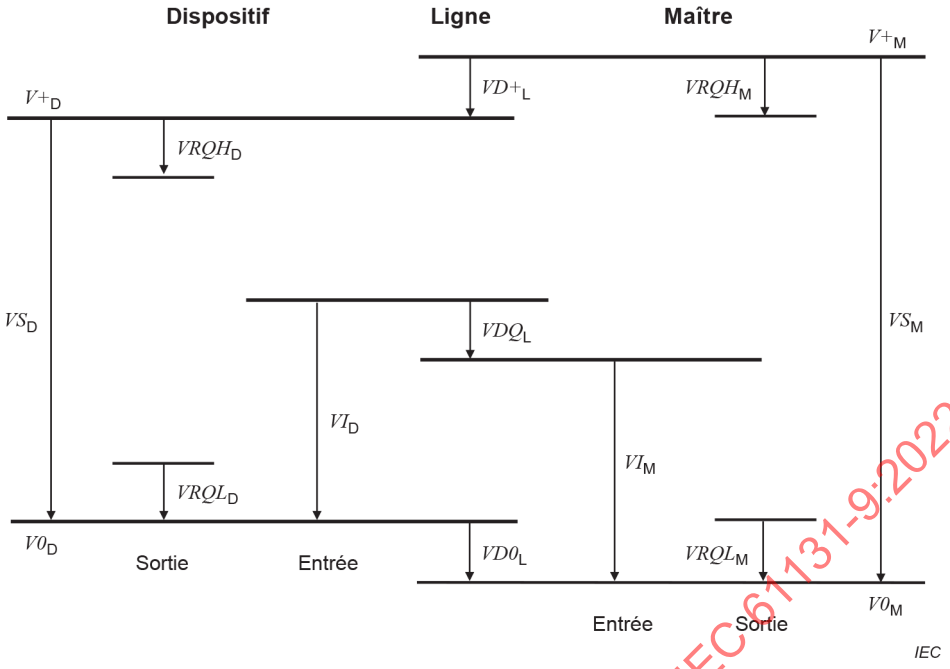


Figure 17 – Définition des niveaux de tension

5.3.2.2 Récepteur

La définition des plages de tension et des seuils de commutation est la même pour le Maître et le Dispositif. Les définitions du Tableau 5 s'appliquent (voir également 5.4.1).

Tableau 5 – Caractéristiques électriques d'un récepteur

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
$VTHH_{D,M}$	Seuil d'entrée 'H'	10,5	n/a	13	V	Voir la NOTE 1
$VTHL_{D,M}$	Seuil d'entrée 'L'	8	n/a	11,5	V	Voir la NOTE 1
$VHYS_{D,M}$	Hystérésis entre seuils d'entrée 'H' et 'L'	0	n/a	n/a	V	Ne doit pas être négative Voir la NOTE 2
VIL_D	Plage de tension admissible 'L'	$V0_D$ 1−,0	n/a	n/a	V	En référence à la tension d'alimentation négative pertinente Voir la NOTE 3
VIL_M	Plage de tension admissible 'L'	$V0_M$	n/a	n/a	V	
VIH_D	Plage de tension admissible 'H'	n/a	n/a	$V+_D$ + 1,0	V	En référence à la tension d'alimentation positive pertinente Voir la NOTE 3
VIH_M	Plage de tension admissible "H"	n/a	n/a	$V+_M$	V	

NOTE 1 Les seuils sont compatibles avec la définition des entrées numériques de type 1 données dans la IEC 61131-2.

NOTE 2 Tension d'hystérésis $VHYS = VTHH - VTHL$.

NOTE 3 En raison du 5.4.1, les signaux du récepteur Maître VI_M sont toujours dans les limites des plages d'alimentations admises.

La Figure 18 représente les seuils de commutation pour la détection des signaux Bas et Haut.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

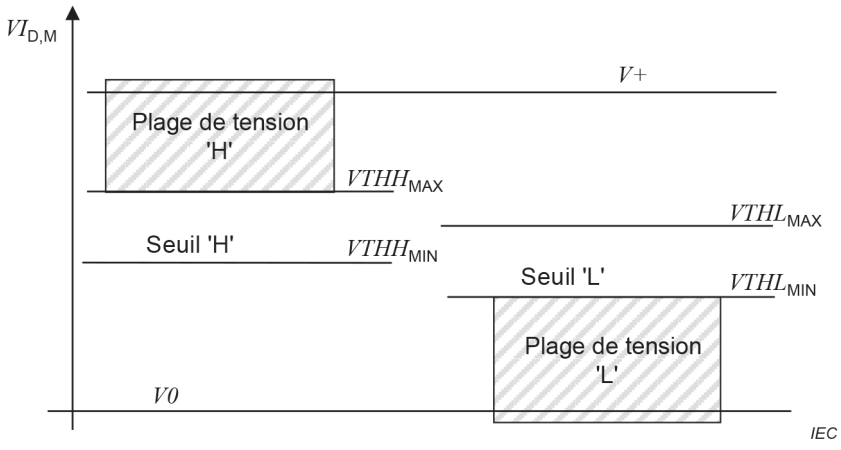


Figure 18 – Seuils de commutation

5.3.2.3 Accès du Maître

Les définitions du Tableau 6 sont valides pour les caractéristiques électriques d'un Accès du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 6 – Caractéristiques électriques d'un Accès du Maître

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
VS_M	Tension d'alimentation des Dispositifs	20	24	30	V	Voir la Figure 17
IS_M	Courant d'alimentation des Dispositifs	200	n/a	n/a	mA	Voir 5.4.1
$ISIR_M$	Capacité d'impulsion de courant pour des Dispositifs	400	n/a	n/a	mA	Voir la Figure 19
ILL_M	Courant de charge ou de décharge pour					
	0 V < VI_M < 5 V	0	n/a	15	mA	Voir la NOTE 1
	5 V < VI_M < 15 V	5/2	n/a	15	mA	
	15 V< VI_M < 30 V	5	n/a	15	mA	
$VRQH_M$	Tension résiduelle 'H'	n/a	n/a	3	V	Chute de tension relativement à $V+_M$, au courant d'attaque maximal IQH_M
$VRQL_M$	Tension résiduelle 'L'	n/a	n/a	3	V	Chute de tension relativement à $V0_M$, au courant d'attaque maximal IQL_M
IQH_M	Courant d'attaque continu 'H'	100	n/a	n/a	mA	
$IQPKH_M$	Courant crête de sortie 'H'	500	n/a	n/a	mA	Valeur absolue Voir la NOTE 2
IQL_M	Courant d'attaque continu 'L'	100	n/a	n/a	mA	
$IQPKL_M$	Courant crête de sortie 'L'	500	n/a	n/a	mA	Valeur absolue Voir la NOTE 2
CQ_M	Capacité d'entrée	n/a	n/a	1,0	nF	f= 0 MHz à 4 MHz

NOTE 1 Un courant minimal de 2 mA pour le mode DI est compatible avec la définition des entrées numériques de type 1 données dans l'IEC 61131-2. En mode de communication, pour la plage 5 V < VI_M < 15 V, le courant minimal est de 5 mA au lieu de 2 mA afin d'obtenir des taux de dérive suffisamment faibles pour des Dispositifs strictement de commutation p.

NOTE 2 Courant de demande de réactivation (5.3.3.3).

Le Maître doit fournir une charge de 400 mA × 50 ms = 20 mAs dans les premières 50 ms qui suivent la mise sous tension sans arrêt de surcharge. Après 50 ms, la limite de courant propre au Maître ou au système s'applique.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

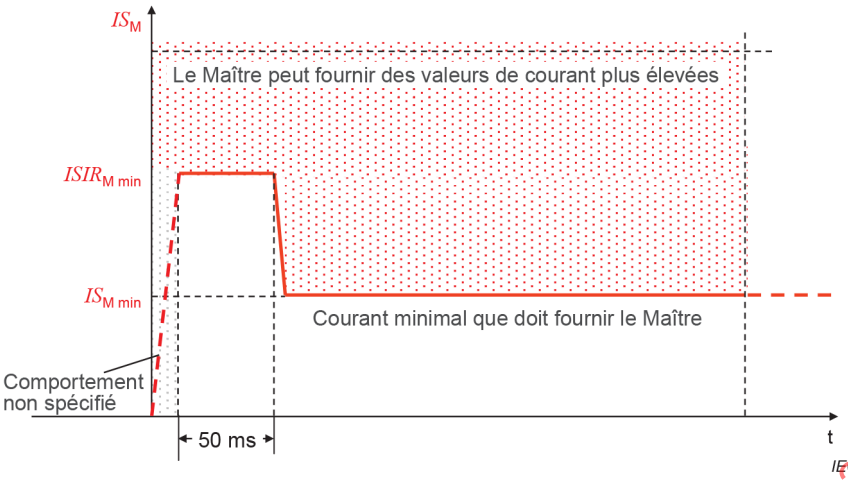


Figure 19 – Courant d'appel et charge (exemple)

5.3.2.4 Dispositif

Les définitions du Tableau 7 sont valides pour les caractéristiques électriques d'un Dispositif.

Tableau 7 – Caractéristiques électriques d'un Dispositif

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
VS_D	Tension d'alimentation	18	24	30	V	Voir la Figure 17
QIS_D	Consommation de charge à la mise sous tension	n/a	n/a	70	mAs	Voir l'équation (1) et le Tableau 8
ΔVS_D	Ondulation	n/a	n/a	1,3	V	Les limites de valeur absolue crête à crête ne doivent pas être dépassées. $f_{ondulation}$= courant continu à 100 kHz
$VRQH_D$	Tension résiduelle 'H'	n/a	n/a	3	V	Chute de tension relativement à $V+_D$ (IEC 60947-5-2)
$VRQL_D$	Tension résiduelle 'L'	n/a	n/a	3	V	Chute de tension relativement à $V0_D$
IQH_D	Courant d'attaque continu Sortie de commutation P (état "Actif")	50	n/a	minimal ($IQPKL_M$)	mA	Valeur minimale due au repli vers l'entrée numérique conformément à l'IEC 61131-2, type 2
IQL_D	Courant d'attaque continu Sortie de commutation N (état "Actif")	0	n/a	minimal ($IQPKH_M$)	mA	Uniquement pour des étages de sortie en configuration symétrique
IQQ_D	Courant passif à $V0_D$ (état "Non actif")	0	n/a	15	mA	Courant de polarisation ou résiduel avec étages du circuit d'attaque des sorties désactivés
CQ_D	Capacité d'entrée	0	n/a	1,0	nF	Capacité effective entre C/Q et L+ ou L− du Dispositif à l'état de réception

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le Dispositif doit être en mesure d'atteindre un état de fonctionnement stable (prêt pour la réactivation) consommant la charge maximale selon l'équation (1).

$$QIS_D = ISIR_M \times 50\text{ms} + (T_{RDL} - 50\text{ms}) \times IS_M \quad (1)$$

La Figure 20 représente comment le comportement de mise sous tension d'un Dispositif est défini par le temps de montée de l'Alimentation 1 et par le temps interne du Dispositif pour se tenir prêt pour l'opération de réactivation.

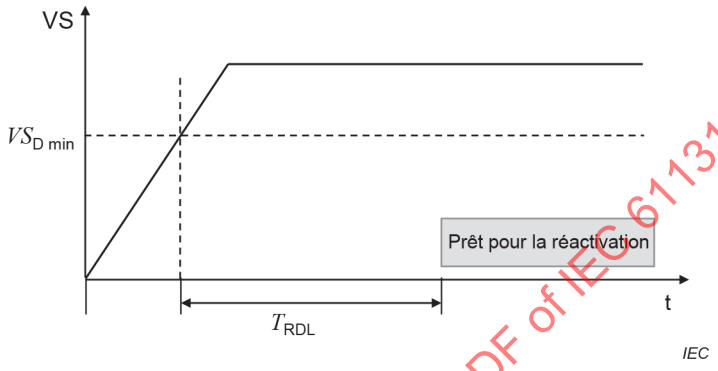


Figure 20 – Temps de mise sous tension pour Alimentation 1

Lors de la mise sous tension, il est obligatoire qu'un Dispositif atteigne l'état prêt à la réactivation dans les limites de temps spécifiées dans le Tableau 8.

Tableau 8 – Temps de mise sous tension

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
T_{RDL}	Prêt à la réactivation après la mise sous tension	n/a	n/a	300	ms	Temps de montée du Dispositif jusqu'à ce qu'il soit prêt pour la détection du signal de réactivation (voir la NOTE)
NOTE Équivaut au délai avant disponibilité de l'IEC 60947-5-2.						

La valeur de 1 nF pour la capacité d'entrée CQ_D est applicable pour une vitesse de transmission de 230,4 kbit/s. Elle peut être élargie à un niveau maximal de 10 nF dans le cas d'une conception d'étage en configuration symétrique lors d'un fonctionnement à des vitesses de transmission inférieures, sous réserve de satisfaire à toutes les exigences concernant les paramètres dynamiques spécifiées au 5.3.3.2.

5.3.3 Exigences temporelles

5.3.3.1 Méthode de transmission

La modulation "Non Retour à Zéro" (NRZ) est utilisée pour le codage bit par bit. Une valeur logique "1" correspond à une différence de potentiel de 0 V entre C/Q et L−. Une valeur logique "0" correspond à une différence de potentiel de +24 V entre C/Q et L−.

Le niveau de circuit ouvert sur la ligne C/Q est 0 V en référence à L−. Un bit de départ a une valeur logique "0", c'est-à-dire +24 V en référence à L−.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Une trame UART est utilisée pour le codage "octet de données" par "octet de données". Le format de la trame UART de l'interface SDCI est une chaîne binaire structurée comme cela est représenté à la Figure 21.

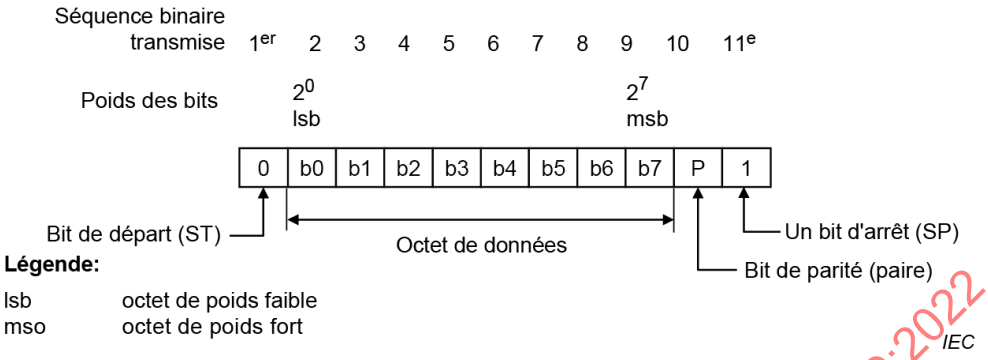


Figure 21 – Format d'une trame UART de l'interface SDCI

La définition du format de la trame UART repose sur l'ISO 1177 [20] et l'ISO/IEC 2022 [16].

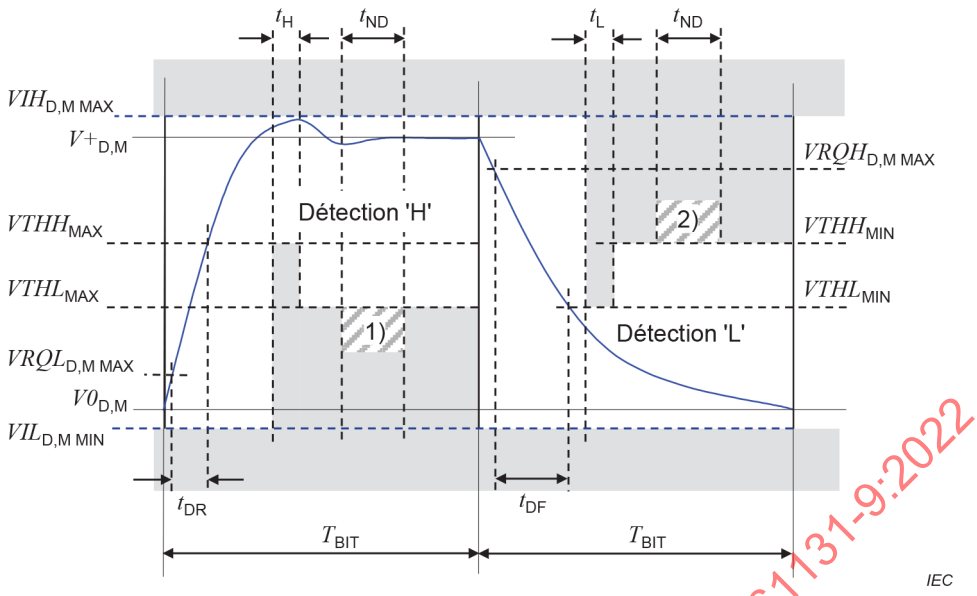
5.3.3.2 Caractéristiques de transmission

Les caractéristiques temporelles de transmission sont présentées sous la forme d'un diagramme en œil avec les plages de signaux admissibles (voir la Figure 22). Ces plages sont applicables au récepteur à la fois dans le Maître et dans le Dispositif.

Quelles que soient les conditions aux limites, l'émetteur doit générer sur la connexion C/Q du récepteur une caractéristique de tension qui s'inscrit dans la plage admissible du diagramme en œil.

Le récepteur doit détecter des bits comme une forme de signal valide dans la plage admissible du diagramme en œil sur la connexion C/Q. Les formes de signal dans les zones de "non-détection" (au-dessous de $VTHL_{MAX}$ ou au-dessus de $VTHH_{MIN}$ et au sein de t_{ND}) ne doivent pas donner lieu à des bits non valides.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022



Légende

1) = aucune détection 'L'

2) = aucune détection 'H'

Figure 22 – Diagramme en œil pour la détection de 'H' et 'L'

Pour qu'une trame UART soit correctement détectée, une caractéristique de signal, telle qu'elle est représentée à la Figure 23, est exigée du côté récepteur. Le délai de propagation du signal entre le signal C/Q et l'entrée UART doit être pris en considération. Le temps T_{BIT} indique toujours le débit binaire du récepteur.

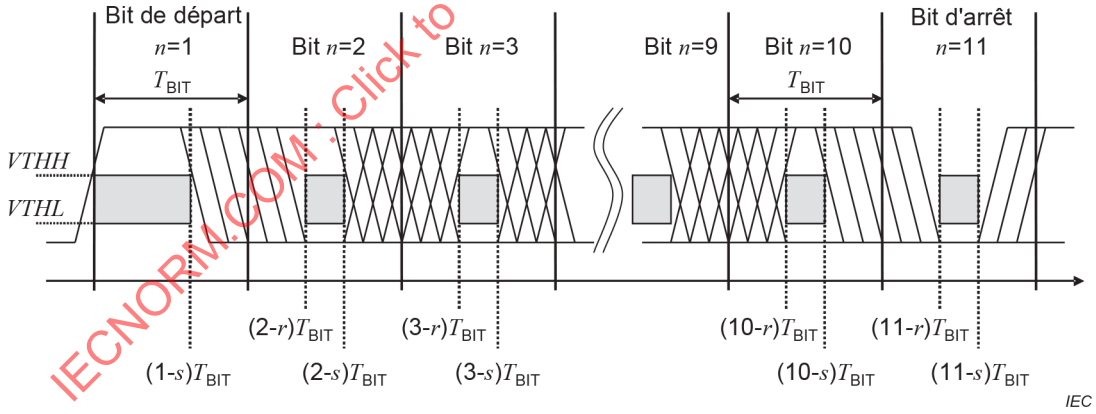


Figure 23 – Diagramme en œil pour la détection correcte d'une trame UART

Pour chaque bit n dans la séquence binaire (n =1…11) d'une trame UAR, le temps $(n\text{-}r)T_{BIT}$ (voir le Tableau 9 pour les valeurs de r) désigne le délai au bout duquel un niveau correct doit être atteint dans les plages 'H' ou 'L' comme cela est représenté dans le diagramme en œil de la Figure 22. Le temps $(n\text{-}s)$ T_{BIT} (voir le Tableau 9 pour les valeurs de s) décrit la durée qui doit s'écouler entre les changements de niveaux. On doit toujours faire référence au diagramme en œil de la Figure 22, lorsque les caractéristiques de signaux dans un temps binaire donné sont concernées.

Cette représentation permet une pondération variable des paramètres d'influence du récepteur, tels que: "exactitude de la vitesse de transmission", "distorsion de largeur binaire" et "taux de dérive" du récepteur.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le Tableau 9 présente les caractéristiques dynamiques de la transmission.

Tableau 9 – Caractéristiques dynamiques de la transmission

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
f_{DTR}	Vitesse de transmission	n/a	4,8 38,4 230,4	n/a	kbit/s	COM1 COM2 COM3
T_{BIT}	Temps binaire à 4,8 kbit/s à 38,4 kbit/s à 230,4 kbit/s		 208,33 26,04 4,34		 µs µs µs	
Δf_{DTRM}	Exactitude de la vitesse de transmission du Maître à 4,8 kbit/s à 38,4 kbit/s à 230,4 kbit/s	 −0,1 −0,1 −0,1	 n/a n/a n/a	 +0,1 +0,1 +0,1	 % % %	Tolérance sur la vitesse de transmission du Maître $\Delta T_{BIT}/T_{BIT}$
r	Début de la période de détection dans un bit donné en référence au front montant du bit de départ	0,65	n/a	n/a	–	Calculé dans chaque cas à partir de la fin d'un bit à un taux d'échantillonnage UART de 8
s	Fin de la période de détection dans un bit donné en référence au front montant du bit de départ	n/a	n/a	0,22	–	Calculé dans chaque cas à partir de la fin d'un bit à un taux d'échantillonnage UART de 8
T_{DR}	Temps de croissance à 4,8 kbit/s à 38,4 kbit/s à 230,4 kbit/s	0 0 0 0	n/a n/a n/a n/a	0,20 41,7 5,2 869	T_{BIT} µs µs ns	En référence à l'unité de temps binaire. Les valeurs minimales peuvent être critiques pour satisfaire aux exigences du H.1.5.
t_{DF}	Temps de décroissance à 4,8 kbit/s à 38,4 kbit/s à 230,4 kbit/s	0 0 0 0	n/a n/a n/a n/a	0,20 41,7 5,2 869	T_{BIT} µs µs ns	En référence à l'unité de temps binaire. Les valeurs minimales peuvent être critiques pour satisfaire aux exigences du H.1.5.
t_{ND}	Temps de suppression du bruit	n/a	n/a	1/16	T_{BIT}	Durée admissible d'un signal de réception au-dessus/au-dessous du seuil de détection sans qu'il n'y ait de détection
t_H	Temps de détection niveau Haut	1/16	n/a	n/a	T_{BIT}	Durée d'un signal de réception au-dessus du seuil de détection pour un niveau 'H'
t_L	Temps de détection niveau Bas	1/16	n/a	n/a	T_{BIT}	Durée d'un signal de réception au-dessous du seuil de détection pour un niveau 'H'

Les paramètres "r" et s' s'appliquent du côté récepteur Maître ou Dispositif, respectivement. Cette définition permet de spécifier de manière plus souple l'exactitude de l'oscillateur, la distorsion binaire et le taux de dérive du côté Dispositif. La distorsion de largeur-binaire globale sur le dernier bit de la trame UART doit fournir un niveau correct dans la plage représentée à la Figure 23.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

5.3.3.3 Impulsion de courant de réactivation

La fonction de réactivation est utilisée pour demander à un Dispositif de passer en mode COMx.

Une invocation de service (PL_WakeUp.req) à partir de la couche DL lance le processus de réactivation (voir 5.2.2.2).

La demande de réactivation (WURQ) commence par une impulsion de courant induite par le Maître (Accès) pendant une durée T_{WU}. La demande de réactivation comporte les phases suivantes (voir la Figure 24):

a) Injection d'un courant IQ_{WU} par le Maître en fonction du niveau de la connexion C/Q. Pour un signal d'entrée équivalent au niveau logique "1", il s'agit d'une source de courant; pour un signal d'entrée équivalent au niveau logique "0", il s'agit d'un collecteur de courant.

b) Temporisation du Dispositif jusqu'à ce qu'il soit prêt à recevoir.

L'impulsion de demande de réactivation peut être détectée par le Dispositif du fait d'une modification de la tension sur la ligne C/Q ou par évaluation du courant de l'élément du circuit d'attaque correspondant dans le temps T_{WU}. Des exemples de Dispositifs à faible puissance de sortie sont représentés à la Figure 24.

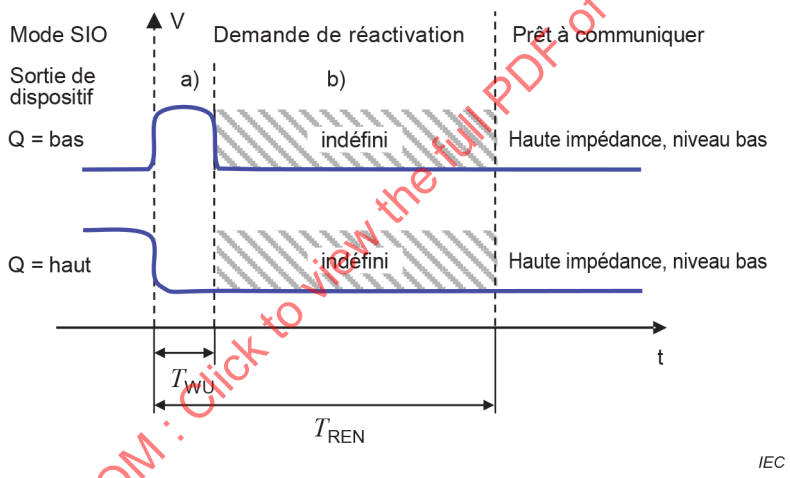


Figure 24 – Demande de réactivation

Le Tableau 10 définit les propriétés du courant et des temporisations associées à la demande de réactivation. Pour les valeurs de $IQPKL_M$ et $IQPKH_M$, voir le Tableau 6.

Tableau 10 – Caractéristiques de la demande de réactivation

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
IQ_{WU}	Amplitude de l'impulsion de courant de réactivation du Maître	$IQPKL_M$ ou $IQPKH_M$	n/a	n/a	mA	Impulsion de courant suivie par l'état de commutation du Dispositif
T_{WU}	Durée de l'impulsion de courant de réactivation du Maître	75	n/a	85	µs	Propriété du Maître
T_{REN}	Délai d'activation de la réception	n/a	n/a	500	µs	Propriété du Dispositif

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

5.4 Alimentation

5.4.1 Options d'alimentation

Le système de connexion SDCI comporte des lignes d'alimentation dédiées, outre la ligne de signalisation. La partie communication d'un Dispositif doit toujours être alimentée par le Maître en utilisant les lignes d'alimentation définies dans le système de connexion à 3 conducteurs (Alimentation 1).

Les fabricants ou les fournisseurs doivent souligner cette exigence dans le manuel d'utilisation du Maître. Toutes les mesures supplémentaires visant à augmenter la robustesse relèvent de la responsabilité du concepteur ou du fabricant du Maître.

Le courant d'alimentation minimal disponible à partir d'un Accès du Maître est spécifié dans le Tableau 6.

La section application du Dispositif peut être alimentée de trois manières différentes:

– par l'intermédiaire des lignes d'alimentation du système de connexion à 3 conducteurs SDCI (Accès de classe A), en utilisant Alimentation 1;
– à partir de lignes d'alimentation supplémentaires du système de connexion à 5 conducteurs SDCI (Accès de classe B), en utilisant une source d'alimentation supplémentaire au niveau du Maître (Power2) qui doit être non réactive, ce qui signifie qu'il n'a aucun impact sur les tensions et courants de Power1 et sur les communications SDCI;
– à partir d'une alimentation locale au niveau du Dispositif (spécifique à la conception) qui doit être non réactive à Power1, garantissant ainsi la bonne communication même en cas de défaillance de l'alimentation locale.

Il est recommandé que les Dispositifs ne consomment pas plus du courant minimal qu'un Maître doit prendre en charge (voir le Tableau 6). Cela permet de faciliter la manipulation des systèmes Maître/Dispositif sans enquêtes, vérification et calculs. Chaque fois qu'un Dispositif exige plus que le courant minimal, les capacités de l'Accès du Maître concerné et de son câblage doivent être vérifiées.

5.4.2 Classe d'Accès B

La Figure 25 représente deux classes d'Accès A et B. Les classes d'Accès B doivent être marqués pour les distinguer des classes d'Accès A compte tenu du risque lié aux incompatibilités sur la broche 2 et la broche 5.

Power 2 sur la Classe d'Accès B doit satisfaire aux exigences suivantes:

– isolation électrique de Power 2 à partir de Power 1;
– degré d'isolation selon la série IEC 60664 (ligne de fuite et distances d'isolement);
– sécurité électrique (SAVE) selon l'IEC 61010-2-201:2017;
– courant direct avec P24 (+) et N24 (-);
– les essais CEM doivent être réalisés avec l'ondulation maximale et la commutation de charge;
– le Dispositif doit continuer à communiquer correctement même en cas de défaillance de Power 2.

Le concepteur d'un Dispositif doit vérifier que Power 1 et Power 2 sont toujours isolés d'un point de vue électrique, même dans des déploiements ou des applications particuliers sur le site du client. Toute violation de cette règle sur un Accès peut avoir un impact sur tous les autres Accès.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

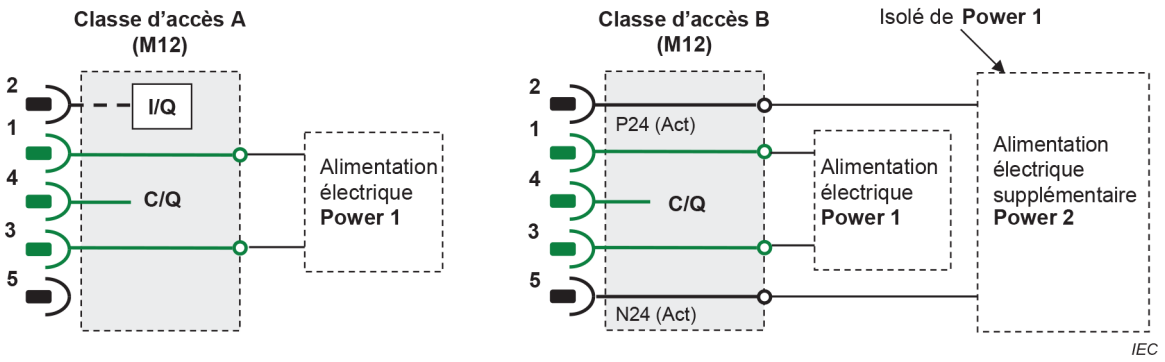


Figure 25 – Définition des classes d'Accès A et B

Le Tableau 11 présente les caractéristiques électriques d'une Classe d'Accès B du Maître (M12).

Tableau 11 – Caractéristiques électriques d'une Classe d'Accès B du Maître

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
$VP24_M$	Tension d'alimentation en courant continu supplémentaire des Dispositifs	20[a]	24	30	V	
$IP24_M$	Courant d'alimentation continu supplémentaire des Dispositifs	1,6[b]	n/a	3,5[c]	A	

[a] Une tension minimale doit être garantie pour les essais au courant d'alimentation recommandé maximal. Côté Dispositif, 18 V doivent être disponibles dans ce cas.

[b] Courant minimal afin de garantir un degré élevé d'interopérabilité.

[c] Courant maximal recommandé pour un calibre de fils de 0,34 mm^2 et un connecteur M12 normalisé est de 3,5 A. Le courant maximal dépend du type de connecteur, du calibre de fils, de la température maximale et du facteur de simultanéité des Accès (consulter le manuel d'utilisation d'un Maître).

En règle générale, les exigences des Dispositifs doivent être vérifiées pour savoir si elles satisfont aux capacités du Maître. Si le facteur de simultanéité des Accès du Maître existe, il doit être indiqué dans le manuel d'utilisation et être respecté par l'utilisateur du Maître.

5.4.3 Exigences de mise sous tension

Les exigences de mise sous tension sont spécifiées en 5.3.2.3 et 5.3.2.4.

5.5 Support

5.5.1 Connecteurs

La connectique et l'affectation des broches du Maître et du Dispositif se fondent sur les spécifications de l'IEC 60947-5-2, avec les extensions définies dans les alinéas ci-dessous.

La Classe d'Accès A utilise les connecteurs M5, M8 et M12, avec quatre broches au plus.

La Classe d'Accès B utilise uniquement les connecteurs M12 avec cinq broches.

Les connecteurs M12 doivent être mécaniquement codés A conformément à l'IEC 61076-2-101.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

En raison de l'héritage ou de la compatibilité, un câblage direct ou différents types de connecteurs peuvent également être utilisés, à condition qu'ils ne portent pas préjudice aux caractéristiques électriques et qu'ils emploient la désignation de signaux spécifiée dans le présent document.

Les connecteurs femelles sont attribués au Maître. Le Tableau 12 donne la connectique et l'affectation des broches et la Figure 26 représente l'agencement et le codage mécanique des connexions M12, M8 et M5.

Tableau 12 – Connectique et affectation des broches du Maître

Broche	Signal	Désignation	Remarque
1	L+	Alimentation (+)	Voir le Tableau 6
2	I/Q	NC/DI(OSSDe)/DO (Classe d'Accès A)	Option 1: NC (non connecté) Option 2: DI Option 3: DI, puis DO configurées Option 4: OSSDe (voir [10])
	P24	P24 (Classe d'Accès B)	Alimentation supplémentaire pour les Dispositifs d'alimentation (Classe d'Accès B)
3	L–	Alimentation (-)	Voir le Tableau 6
4	C/Q	SIO(OSSDe)/SDCI	Mode E/S normalisé (DI/DO) ou SDCI (voir les caractéristiques électriques de DO dans le Tableau 6). Voir [10] pour les définitions OSSDe.
5	NC	NC (Classe d'Accès A)	Ne doit pas être connecté du côté Maître (Classe d'Accès A).
	N24	N24 (Classe d'Accès B)	Potentiel de référence à l'alimentation supplémentaire (Classe d'Accès B)
NOTE Le connecteur M12 a toujours 5 broches du côté Maître (femelle).			

La Figure 26 représente l'agencement des deux classes d'Accès, A et B. Afin d'éviter les éventuels risques d'incompatibilité, les Accès de classe B doivent être marqués de manière à faire la distinction avec les Accès de classe A.

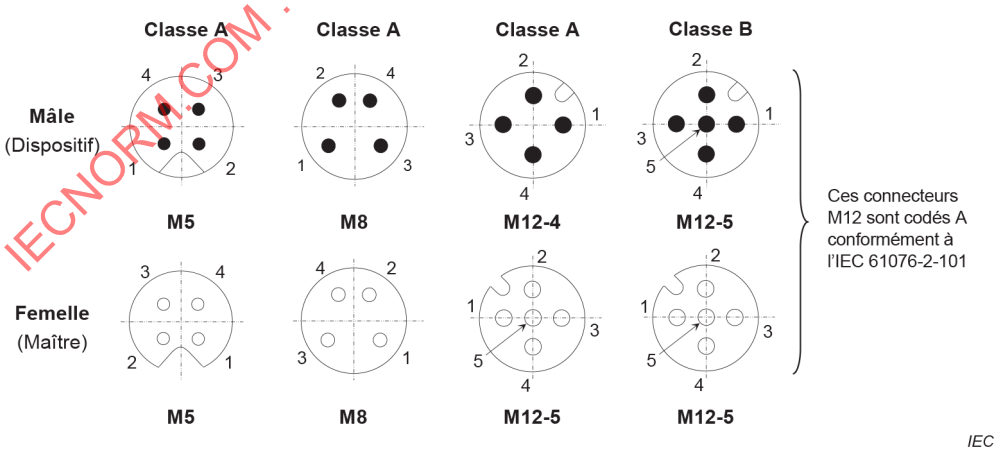


Figure 26 – Vue de face de l'agencement des broches

Les connecteurs mâles sont attribués au Dispositif. Le Tableau 13 répertorie la connectique et l'affectation des broches.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 13 – Connectique et affectation des broches du dispositif

Broche	Signal	Désignation	Remarque
1	L+	Alimentation (+)	Voir le Tableau 7
2	I/Q [a]	NC/DI(OSSDe)/DO/ AI/AO (Classe d'Accès A)	Option 1: NC (non connecté) Option 2: DI (vue du Maître) Option 3: DO (vue du Maître) Option 4: Signal analogique (I / U) [d] Option 5: OSSDe (voir [10])
	P24 [b]	P24 (Classe d'Accès B)	Alimentation supplémentaire des Dispositifs alimentés (Classe d'Accès B)
3	L−	Alimentation (-)	Voir le Tableau 7
4	C/Q [c]	SIO(OSSDe)/SDCI	Mode E/S normalisé (DI/DO) ou SDCI (voir les caractéristiques électriques de DO dans le Tableau 6). Voir [10] pour les définitions OSSDe.
5	NC	NC (Classe d'Accès A)	Option 1: Ne doit pas être connecté du côté Maître (Classe d'Accès A).
	N24	N24 (Classe d'Accès B)	Option 2: Référence à l'alimentation supplémentaire (Classe d'Accès B)

[a] Les signaux du Dispositif ne doivent pas avoir d'impact sur l'entrée I/Q d'un Maître. Les Dispositifs doivent résister au courant continu permanent (voir le Tableau 6) côté Maître.

[b] Les Dispositifs qui s'appuient sur une Classe d'Accès A doivent utiliser une connexion à 3 fils dans ce cas, afin d'éviter de contourner l'isolation galvanique.

[c] Un Maître doit toujours être en mesure d'établir et de maintenir une communication SDCI sans interférence.

[d] La valeur de U est en général comprise entre 0 V et 10 V, 1-5 V et pour I entre 0 et 20 mA, 4 et 20 mA.

5.5.2 Câblage

Pour la communication SDCI, le support de transmission est un câble à plusieurs conducteurs (trois ou plus). Les définitions des alinéas suivants couvrent implicitement les définitions de la tension statique du Tableau 5 et de la Figure 17. Pour assurer la fiabilité fonctionnelle, les propriétés du câblage doivent être conformes au Tableau 14.

Tableau 14 – Caractéristiques du câblage

Propriété	Minimale	Type	Maximale	Unité
Longueur L	0	n/a	20	m
Résistance de boucle globale RL_{eff} [a]	n/a	n/a	6,0 (pour un courant de 200 mA) 1,2 (pour un courant de 1 000 mA)	Ω
Capacité effective de la ligne CL_{eff}	n/a	n/a	3,0	nF (<1 MHz)

[a] La résistance de boucle globale doit être assignée de sorte que les tensions d'alimentation minimales du Dispositif soient garanties au courant d'alimentation maximal (voir le Tableau 7).

La résistance de boucle RL_{eff} et la capacité effective de la ligne CL_{eff} peuvent être mesurées comme cela est représenté à la Figure 27.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

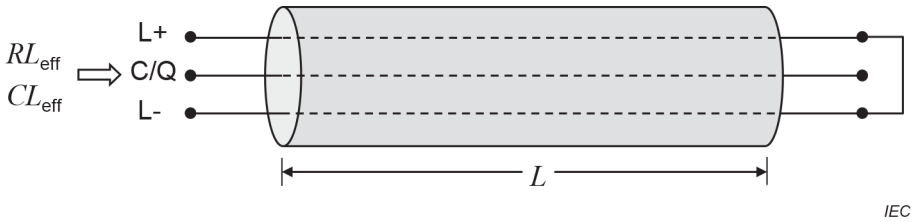


Figure 27 – Schéma de référence de la capacité effective de la ligne et de la résistance de boucle

Le Tableau 15 présente les conducteurs du câble et les codes de couleurs qui leur sont attribués.

Tableau 15 – Affectation des conducteurs du câble

Signal	Désignation	Couleur	Remarque
L–	Alimentation (-)	Bleu[a]	Système de connexion SDCI à 3 conducteurs
C/Q	Signal de communication	Noir[a]	Système de connexion SDCI à 3 conducteurs
L+	Alimentation (+)	Marron[a]	Système de connexion SDCI à 3 conducteurs
I/Q	DI ou DO	Blanc[a]	Facultatif
P24	Alimentation supplémentaire (+)	N'importe quelle autre	Facultatif
N24	Alimentation supplémentaire (-)	N'importe quelle autre	Facultatif
[a] Correspondant à l'IEC 60947-5-2.			

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

6 Entrées et sorties normalisées (SIO)

La Figure 85 et la Figure 96 représentent la façon dont le mode SIO permet à un Dispositif de contourner les couches de communication SDCI et de mapper le signal DI ou DO directement dans le message d'échange des données du bus de terrain ou système de niveau supérieur. Le changement entre le mode SDCI et le mode SIO est défini par la configuration de l'utilisateur ou implicitement par les services des applications du Maître. La Gestion Système prend en charge l'initialisation correspondante ou la désactivation des couches de communication SDCI et de la couche physique (mode commutation). Les caractéristiques des interfaces pour les signaux DI et DO sont issues de l'IEC 61131-2, type 1.

7 Couche de liaison de données (DL)

7.1 Généralités

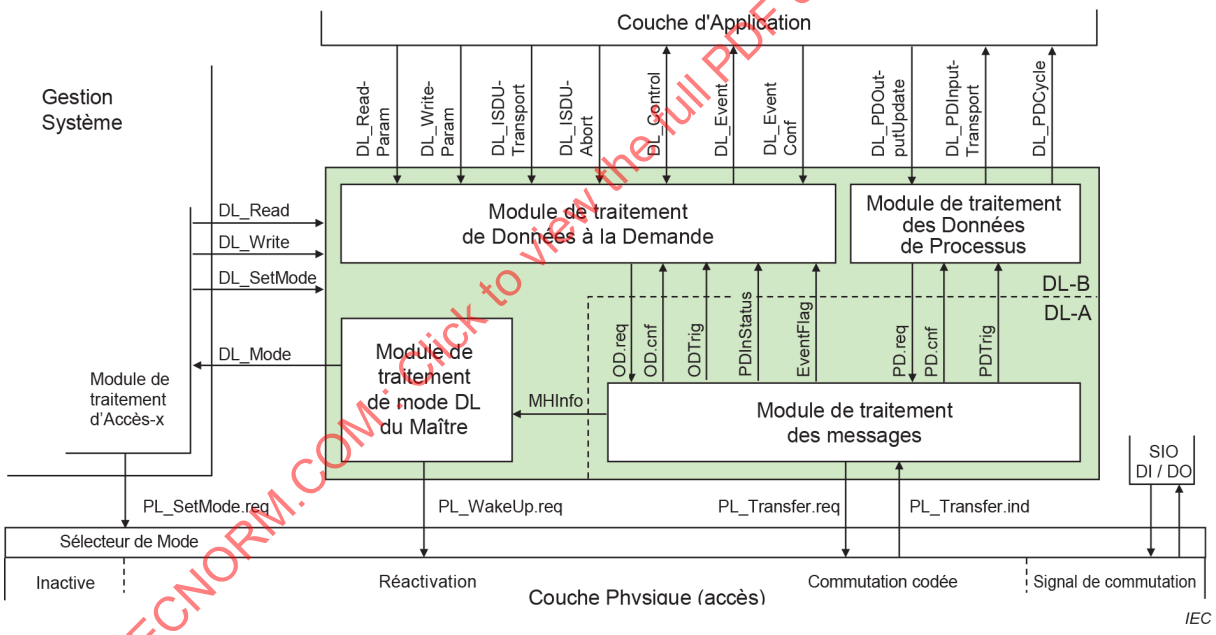
Les couches de liaison de données de l'interface SDCI sont chargées de la transmission des messages entre un Maître et un Dispositif, par l'intermédiaire de la liaison physique. La couche de liaison de données utilise plusieurs types de séquences M ("séquence de message") pour différentes catégories de données.

Un ensemble de services DL est utilisé par la couche d'application (AL) pour l'échange de Données de Processus (PD) et de Données à la Demande (OD). Un autre ensemble de services DL est utilisé par la Gestion Système (SM) pour la récupération des paramètres d'identification

de Dispositif et le réglage des diagrammes d'états dans la couche DL. La couche DL utilise des services PL pour commander la couche physique (PL) et échanger des trames UART. La couche DL est chargée de la détection d'erreurs dans les messages (qu'elles soient internes ou renvoyées par la PL) et de prendre les mesures correctives appropriées (par exemple, une nouvelle tentative).

Les couches de liaison de données sont structurées en fonction de la nature des catégories de données en modules de traitement de Données de Processus et modules de traitement de Données à la Demande, qui à leur tour utilisent un module de traitement de messages pour traiter les transmissions de messages demandées. Les modes spéciaux des Accès du Maître, tels que réactivation, COMx ainsi que SIO (désactivation de la communication), nécessitent un module de traitement de mode DL spécifique au sein de la DL du Maître. La modulation du signal spécial de réactivation nécessite une détection de signaux du côté du Dispositif et de ce fait, elle utilise un module de traitement de mode DL au sein de la DL du Dispositif. Chaque module de traitement comporte son propre diagramme d'états.

La couche de liaison de données est divisée en section A de DL avec ses propres services internes, et section B de DL avec des services externes. La couche DL utilise des invocations supplémentaires de procédures réglementaires internes entre les modules de traitement définies dans la section "éléments internes" des tables de transitions d'états correspondants. La Figure 28 donne une vue d'ensemble de la structure et des services de la couche de liaison de données du Maître.



NOTE Cette figure utilise les conventions données en 3.3.5.

Figure 28 – Structure et services de la couche de liaison de données (Maître)

La Figure 29 donne une vue d'ensemble de la structure et des services de la couche de liaison de données du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

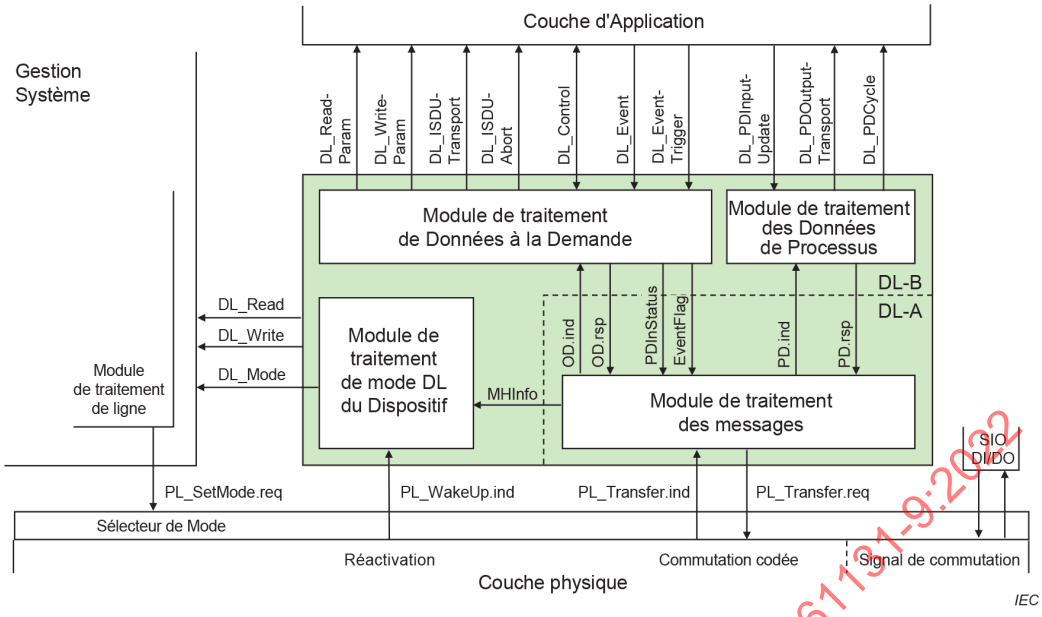


Figure 29 – Structure et services de la couche de liaison de données (Dispositif)

7.2 Services de la couche liaison de données

7.2.1 Services de la section B de couche DL

7.2.1.1 Vue d'ensemble des services au sein du Maître et du Dispositif

Le présent paragraphe 7.2.1 définit les services de la couche de liaison de données à fournir à la couche d'application et la Gestion Système par l'intermédiaire de ses interfaces externes. Le Tableau 16 énumère les attributions du Maître et du Dispositif quant à leur rôle d'initiateur ou de destinataire des services individuels de liaison de données (DL). Des champs vides indiquent la non-disponibilité de ce service au niveau du Maître ou du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 16 – Attributions des services au sein du Maître et du Dispositif

Désignation du service	Maître	Dispositif
DL_ReadParam	R	I
DL_WriteParam	R	I
DL_ISDUTransport	R	I
DL_ISDUAbort	R	I
DL_PDOutputUpdate	R	
DL_PDOutputTransport		I
DL_PDInputUpdate		R
DL_PDInputTransport	I	
DL_PDCycle	I	I
DL_SetMode	R	
DL_Mode	I	
DL_Event	I	R
DL_EventConf	R	
DL_EventTrigger		R
DL_Control	I / R	R / I
DL_Read	R	I
DL_Write	R	I
Légende (voir 3.3.4) I Initiateur du service R Destinataire (Répondeur) du service		

En ce qui concerne les conventions et la manière de lire les descriptifs de services fournis aux 7.2, 8.2, 9.2.2 et 9.3.2, voir 3.3.

7.2.1.2 DL_ReadParam

Le service DL_ReadParam est utilisé par la couche AL pour lire une valeur de paramètre en provenance du Dispositif, au moyen du canal de communication de page. Les paramètres des primitives de service sont énumérés dans le Tableau 17.

Tableau 17 – DL_ReadParam

Nom du paramètre	.req	.cnf	.ind	.rsp
Argument Adresse	M M		M M	
Résultat (+) Valeur		S M		S M
Résultat (−) ErrorInfo		S M		

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Adresse

Ce paramètre contient l'adresse du paramètre de Dispositif demandé, c'est-à-dire les adresses de paramètres de Dispositif dans le canal de communication de page (voir le Tableau B.1).

Valeurs admises: 0 à 31.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Valeur

Ce paramètre contient la valeur du paramètre de Dispositif à lire.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

7.2.1.3 DL_WriteParam

Le service DL_WriteParam est utilisé par la couche AL pour écrire une valeur de paramètre vers le Dispositif au moyen du canal de communication de page. Les paramètres des primitives de service sont énumérés dans le Tableau 18.

Tableau 18 – DL_WriteParam

Nom du paramètre	.req	.cnf	.ind
Argument	M		M
Adresse	M		M
Valeur	M		M
Résultat (+)		S	
Résultat (−)		S	
ErrorInfo		M	

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Adresse

Ce paramètre contient l'adresse du paramètre de Dispositif demandé, c'est-à-dire les adresses de paramètres de Dispositif dans le canal de communication de page.

Valeurs admises: 16 à 31, selon les droits d'Accès aux paramètres de Dispositif.

Valeur

Ce paramètre contient la valeur du paramètre de Dispositif à écrire.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

7.2.1.4 DL_Read

Le service DL_Read est utilisé par la Gestion Système pour lire une valeur de paramètre de Dispositif au moyen du canal de communication de page. Les paramètres des primitives de service sont énumérés dans le Tableau 19.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 19 – DL_Read

Nom du paramètre	.req	.cnf	.ind	.rsp
Argument	M		M	
Adresse	M		M	
Résultat (+)		S		S
Valeur		M		M
Résultat (−)		S		
ErrorInfo		M		

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Adresse

Ce paramètre contient l'adresse du paramètre de Dispositif demandé, c'est-à-dire les adresses de paramètres de Dispositif dans le canal de communication de page (voir le Tableau B.1).

Valeurs admises: 0 à 15, selon les droits d'Accès aux paramètres de Dispositif.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Valeur

Ce paramètre contient les valeurs du paramètre de Dispositif à lire.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

7.2.1.5 DL_Write

Le service DL_Write est utilisé par la Gestion Système pour écrire, vers le Dispositif, une valeur de paramètre de Dispositif, au moyen du canal de communication de page. Les paramètres des primitives de service sont énumérés dans le Tableau 20.

Tableau 20 – DL_Write

Nom du paramètre	.req	.cnf	.ind
Argument	M		M
Adresse	M		M
Valeur	M		M
Résultat (+)		S	
Résultat (−)		S	
ErrorInfo		M	

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Adresse

Ce paramètre contient l'adresse du paramètre de Dispositif demandé, c'est-à-dire les adresses de paramètres de Dispositif dans le canal de communication de page.

Valeurs admises: 0 à 15, selon les droits d'Accès aux paramètres.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Valeur

Ce paramètre contient la valeur du paramètre de Dispositif à écrire.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

7.2.1.6 DL_ISDUTransport

Le service DL_ISDUTransport est utilisé pour transporter une ISDU. Ce service est utilisé par le Maître pour envoyer une demande de service de la couche d'application du Maître vers le Dispositif. Il est utilisé par le Dispositif pour envoyer une réponse de service au Maître depuis la couche d'application du Dispositif. Les paramètres des primitives de service sont énumérés dans le Tableau 21.

Tableau 21 – DL_ISDUTransport

Nom du paramètre	.req	.ind	.cnf	.rsp
Argument	M	M		
ValueList	M	M		
Résultat (+)			S	S
Données			C	C
Qualificatif			M	M
Résultat (−)			S	S
ISDUTransportErrorInfo			M	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

ValueList

Ce paramètre contient les paramètres de fonctionnement pertinents.

Type de paramètre: Enregistrement.

Index

Valeurs admises: 2 à 65 535 (Voir B.2.1 pour les contraintes).

Sous-index

Valeurs admises: 0 à 255.

Données

Type de paramètre: Chaîne d'octets.

Sens

Valeurs admises:
READ (opération de lecture),
WRITE (opération d'écriture).

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Données

Type de paramètre: Chaîne d'octets.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Qualificatif

Valeurs admises: une réponse de service I du Dispositif conformément au Tableau A.12.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ISDUTransportErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:

NO_COMM	(pas de communication disponible),
STATE_CONFLICT	(service non disponible en l'état courant),
ISDU_TIMEOUT	(Expiration de la période d'acquittement des ISDU, voir le Tableau 102),
ISDU_NOT_SUPPORTED	(ISDU non mise en œuvre),
VALUE_OUT_OF_RANGE	(La valeur du paramètre de service ne respecte pas les plages définies).

7.2.1.7 DL_ISDUAbort

Le service DL_ISDUAbort abandonne la transmission ISDU en cours. Ce service n'a pas de paramètres. Les primitives de service sont énumérées dans le Tableau 22.

Tableau 22 – DL_ISDUAbort

Nom du paramètre	.req	.cnf
<aucun>		

Le service renvoie une confirmation après abandon de la transmission des ISDU.

7.2.1.8 DL_PDOutputUpdate

La couche d'application du Maître utilise le service DL_PDOutputUpdate pour mettre à jour les données de sortie (Données de Processus du Maître vers le Dispositif) sur la couche de liaison de données. Les paramètres des primitives de service sont énumérés dans le Tableau 23.

Tableau 23 – DL_PDOutputUpdate

Nom du paramètre	.req	.cnf
Argument	M	
OutputData	M	
Résultat (+)		S
TransportStatus		M
Résultat (−)		S
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

OutputData

Ce paramètre contient les Données de Processus fournies par la couche d'application.

Type de paramètre: Chaîne d'octets.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

TransportStatus

Ce paramètre indique si la couche de liaison de données est dans un état qui permet le transfert des données au(x) partenaire(s) de communication.

Valeurs admises:
OUI (transmission de données admise),
NON (transmission de données non admise).

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

7.2.1.9 DL_PDOutputTransport

La couche de liaison de données sur le Dispositif utilise les services DL_PDOutputTransport pour transférer le contenu des Données de Processus de sortie vers la couche d'application (du Maître vers le Dispositif). Les paramètres des primitives de service sont énumérés dans le Tableau 24.

Tableau 24 – DL_PDOutputTransport

Nom du paramètre	.ind
Argument	M
OutputData	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

OutputData

Ce paramètre contient les Données de Processus à transmettre à la couche d'application.

Type de paramètre: Chaîne d'octets.

7.2.1.10 DL_PDInputUpdate

La couche d'application du Dispositif utilise les services DL_PDInputUpdate pour mettre à jour les données de sortie (Données de Processus du Dispositif vers le Maître) sur la couche de liaison de données. Les paramètres des primitives de service sont énumérés dans le Tableau 25.

Tableau 25 – DL_PDInputUpdate

Nom du paramètre	.req	.cnf
Argument	M	
InputData	M	
Résultat (+)		S
TransportStatus		M
Résultat (−)		S
ErrorInfo		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

InputData

Ce paramètre contient les Données de Processus fournies par la couche d'application.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

TransportStatus

Ce paramètre indique si la couche de liaison de données est dans un état qui permet le transfert des données au(x) partenaire(s) de communication.

Valeurs admises:
OUI (transmission de données admise),
NON (transmission de données non admise).

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

7.2.1.11 DL_PDInputTransport

La couche de liaison de données sur le Maître utilise les services DL_PDInputTransport pour transférer le contenu des données d'entrée (Données de Processus du Dispositif vers le Maître) vers la couche d'application. Les paramètres des primitives de service sont énumérés dans le Tableau 26.

Tableau 26 – DL_PDInputTransport

Nom du paramètre	.ind
Argument	M
InputData	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

InputData

Ce paramètre contient les Données de Processus à transmettre à la couche d'application.

Type de paramètre: Chaîne d'octets.

7.2.1.12 DL_PDCycle

La couche de liaison de données utilise le service DL_PDCycle pour indiquer à la couche d'application la fin d'un cycle de Données de Processus. Ce service n'a pas de paramètres. Les primitives de service sont énumérées dans le Tableau 27.

Tableau 27 – DL_PDCycle

Nom du paramètre	.ind
<none>	

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.1.13 DL_SetMode

Le service DL_SetMode est utilisé par la Gestion Système pour configurer les diagrammes d'états de la couche de liaison de données et envoyer les valeurs de caractéristiques de fonctionnement exigées à la couche de liaison de données. Les paramètres des primitives de service sont énumérés dans le Tableau 28.

Tableau 28 – DL_SetMode

Nom du paramètre	.req	.cnf
Argument Mode ValueList	M M U	
Résultat (+)		S
Résultat (−) ErrorInfo		S M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Mode

Ce paramètre indique le mode demandé de la DL du Maître sur un Accès individuel.

Valeurs admises:
INACTIVE (Le module de traitement doit passer à l'état INACTIVE),
STARTUP (Le module de traitement doit passer à l'état STARTUP),
PREOPERATE (Le module de traitement doit passer à l'état PREOPERATE),
OPERATE (Le module de traitement doit passer à l'état OPERATE).

ValueList

Ce paramètre contient les paramètres de fonctionnement pertinents.

Structure des données: enregistrement.

M-sequenceTime: (à propager au module de traitement des messages)

M-sequenceType: (à propager au module de traitement des messages)

Valeurs admises:
TYPE_0,
TYPE_1_1, TYPE_1_2, TYPE_1_V,
TYPE_2_1, TYPE_2_2, TYPE_2_3, TYPE_2_4, TYPE_2_5, TYPE_2_V
(Le TYPE_1_1 force le mode d'entrelacement de la transmission de Données de Processus et de Données à la Demande, voir 7.3.4.2).

PDInputLength: (à propager au module de traitement des messages)

PDOutputLength: (à propager au module de traitement des messages)

OnReqDataLengthPerMessage: (à propager au module de traitement des messages)

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (service non disponible en l'état courant),
PARAMETER_CONFLICT (cohérence du jeu de paramètres non respectée).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.1.14 DL_Mode

La couche DL utilise le service DL_Mode pour informer la Gestion Système qu'un certain état opérationnel a été atteint. Les paramètres des primitives de service sont énumérés dans le Tableau 29.

Tableau 29 – DL_Mode

Nom du paramètre	.ind
Argument	M
RealMode	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

RealMode

Ce paramètre indique l'état du module de traitement DL-mode.

Valeurs admises:

INACTIVE	(Le module de traitement est passé à l'état INACTIVE),
COM1	(Mode COM1 établi),
COM2	(Mode COM2 établi),
COM3	(Mode COM3 établi),
COMLOST	(Perte de communication),
ESTABCOM	(Le module de traitement est passé à l'état EstablishCom),
STARTUP	(Le module de traitement est passé à l'état STARTUP),
PREOPERATE	(Le module de traitement est passé à l'état PREOPERATE),
OPERATE	(Le module de traitement est passé à l'état OPERATE).

7.2.1.15 DL_Event

Le service DL_Event indique un état en instance ou des informations d'erreur. La cause d'un Événement se trouve dans un Dispositif et l'application du Dispositif déclenche le transfert de l'Événement. Les paramètres des primitives de service sont énumérés dans le Tableau 30.

Tableau 30 – DL_Event

Nom du paramètre	.req	.ind
Argument	M	M
Instance	M	M
Type	M	M
Mode	M	M
EventCode	M	M
EventsLeft		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Instance

Ce paramètre indique l'origine de l'Événement.

Valeurs admises: Application (voir le Tableau A.17).

Type

Ce paramètre indique la catégorie de l'Événement.

Valeurs admises: ERROR, WARNING, NOTIFICATION (voir le Tableau A.19).

Mode

Ce paramètre indique le mode de l'Événement.

Valeurs admises: SINGLESHOT, APPEARS, DISAPPEARS (voir le Tableau A.20).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

EventCode

Ce paramètre contient un code identifiant un certain Événement (voir le Tableau D.1)..

Type de paramètre: Entier 16-Bit non signé.

EventsLeft

Ce paramètre indique le nombre d'Événements non traités.

7.2.1.16 DL_EventConf

Le service DL_EventConf confirme les Événements transmis au moyen du module de traitement d'Événements. Ce service n'a pas de paramètres. Les primitives de service sont énumérées dans le Tableau 31.

Tableau 31 – DL_EventConf

Nom du paramètre	.req	.cnf
<aucun>		

7.2.1.17 DL_EventTrigger

La demande DL_EventTrigger démarre la signalisation d'Événements (voir drapeau d'Événement à la Figure A.3) et fige la mémoire d'Événements dans la couche DL. La confirmation est renvoyée après traitement des Événements activés. Les demandes DL_EventTrigger supplémentaires sont ignorées jusqu'à ce que la précédente ait été confirmée (voir 7.3.8, 8.3.3 et Figure 66). Ce service n'a pas de paramètres. Les primitives de service sont énumérées dans le Tableau 32.

Tableau 32 – DL_EventTrigger

Nom du paramètre	.req	.cnf
<aucun>		

7.2.1.18 DL_Control

Le Maître utilise le service DL_Control pour acheminer des informations de commande, au moyen du mécanisme MasterCommand, à l'application du Dispositif correspondant et pour obtenir des informations de commande au moyen du mécanisme indicateur (drapeau) d'état des PD (voir A.1.5) et le service PDInStatus (voir 7.2.2.5). Les paramètres des primitives de service sont énumérés dans le Tableau 33.

Tableau 33 – DL_Control

Nom du paramètre	.req	.ind
Argument ControlCode	M M	M M(=)

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

ControlCode

Ce paramètre indique l'état du qualificatif des Données de Processus (PD).

Valeurs admises:
VALID (Données de Processus d'entrée valides; voir 7.2.2.5, 8.2.2.12),
INVALID (Données de Processus d'entrée non valides),

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

PDOUTVALID (Données de Processus de sortie valides; voir 7.3.7.1),
PDOUTINVALID (Données de Processus de sortie non valides ou manquantes).

7.2.2 Services de la section A de couche DL

7.2.2.1 Vue d'ensemble

Comme cela est défini en 7.1, la couche de liaison de données est divisée en une section supérieure B (DL-B) et une section inférieure A (DL-A). La couche DL-A comprend le module de traitement des messages, comme cela est représenté à la Figure 28 et à la Figure 29.

Le module de traitement des messages du Maître code les commandes et données en messages et les envoie au Dispositif connecté par l'intermédiaire de la couche physique. Il reçoit des messages du Dispositif par l'intermédiaire de la couche physique et achemine leur contenu aux modules de traitement correspondants sous la forme d'une confirmation. Lorsque le "drapeau d'Événement" est établi dans un message de Dispositif (voir A.1.5), le module de traitement des messages du Maître invoque un service EventFlag pour solliciter le module de traitement d'Événements.

Le module de traitement des messages du Maître doit utiliser une stratégie de renouvellement de tentative en cas de message altéré, c'est-à-dire s'il reçoit d'un Dispositif une somme de contrôle incorrecte ou en l'absence de toute somme de contrôle. Dans ces cas-là, le Maître doit répéter le message deux fois (voir le Tableau 102). En cas d'échec des nouvelles tentatives, une confirmation négative doit être fournie et le Maître doit relancer la communication au moyen du module de traitement d'Accès-x, en commençant par une réactivation.

Au bout d'une phase de démarrage, le module de traitement des messages lance un cycle opérationnel, le type de séquence M et la durée de cycle étant fournis par le service DL_SetMode.

Le Tableau 34 énumère les attributions du Maître et du Dispositif quant à leurs rôles d'initiateur (I) ou de destinataire (R) dans le contexte de l'exécution de leurs services DL-A individuels.

Tableau 34 – Services DL-A au sein du Maître et du Dispositif

Désignation du service	Maître	Dispositif
OD	R	I
PD	R	I
EventFlag	I	R
PDInStatus	I	R
MHInfo	I	I
ODTrig	I	
PDTrig	I	

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.2.2 OD

Le service OD est utilisé pour configurer les Données à la Demande du prochain message à envoyer. La confirmation du service contient les données en provenance du récepteur. Les paramètres des primitives de service sont énumérés dans le Tableau 35.

Tableau 35 – OD

Nom du paramètre	.req	.ind	.rsp	.cnf
Argument	M	M		
RWDirection	M	M		
ComChannel	M	M		
AddressCtrl	M	M		
Longueur	M	M		
Données	C	C		
Résultat (+)			S	S
Données			C	C(=)
Longueur			M	M
Résultat (−)			S	S
ErrorInfo			M	M(=)

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

RWDirection

Ce paramètre indique le sens, lecture ou écriture.

Valeurs admises:
READ (opération de lecture),
WRITE (opération d'écriture).

ComChannel

Ce paramètre indique le canal de communication sélectionné pour la transmission.

Valeurs admises: DIAGNOSIS, PAGE, ISDU (voir le Tableau A.1).

AddressCtrl

Ce paramètre contient l'adresse ou la valeur du contrôle de flux (voir A.1.2).

Valeurs admises: 0 à 31.

Longueur

Ce paramètre contient la longueur des données à transmettre.

Valeurs admises: 0 à 32.

Données

Ce paramètre contient les données à transmettre.

Type de Données: Chaîne d'octets.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Données

Ce paramètre contient la valeur des données à lire.

Longueur

Ce paramètre contient la longueur du paquet de données reçu.

Valeurs admises: 0 à 32.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.2.3 PD

Le service PD est utilisé pour configurer les Données de Processus à envoyer par l'intermédiaire du canal de communication du processus. La confirmation du service contient les données en provenance du récepteur. Les paramètres des primitives de service sont énumérés dans le Tableau 36.

Tableau 36 – PD

Nom du paramètre	.req	.ind	.rsp	.cnf
Argument	M	M		
PDInAddress	C	C(=)		
PDInLength	C	C(=)		
PDOut	C	C(=)		
PDOutAddress	C	C(=)		
PDOutLength	C	C(=)		
Résultat (+)			S	S
PDIn			C	C(=)
Résultat (−)			S	S
ErrorInfo			M	M(=)

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

PDInAddress

Ce paramètre contient l'adresse des Données de Processus d'entrée demandées (voir 7.3.4.2).

PDInLength

Ce paramètre contient la longueur des Données de Processus d'entrée demandées.

Valeurs admises: 0 à 32.

PDOut

Ce paramètre contient les Données de Processus à transférer du Maître vers le Dispositif.

Type de Données: Chaîne d'octets.

PDOutAddress

Ce paramètre contient l'adresse des Données de Processus de sortie transmises (voir 7.3.4.2).

PDOutLength

Ce paramètre contient la longueur des Données de Processus de sortie transmises.

Valeurs admises: 0 à 323.

Résultat(+)

Ce paramètre de sélection indique que le service a été exécuté avec succès.

PDIn

Ce paramètre contient les Données de Processus à transférer du Dispositif vers le Maître.

Type de Données: Chaîne d'octets.

Résultat (−)

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_COMM (pas de communication disponible),
STATE_CONFLICT (service non disponible en l'état courant).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.2.2.4 EventFlag

Le service EventFlag établit ou notifie l'état du "drapeau d'Événement" (voir A.1.5) lors d'une communication cyclique. Les paramètres des primitives de service sont énumérés dans le Tableau 37.

Tableau 37 – EventFlag

Nom du paramètre	.ind	.req
Argument Flag	M	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Flag

Ce paramètre contient la valeur du "drapeau d'Événement".

Valeurs admises:
TRUE ("Drapeau d'Événement"= 1),
FALSE ("Drapeau d'Événement" = 0).

7.2.2.5 PDInStatus

Le service PDInStatus établit et notifie le qualificatif de validité des Données de Processus d'entrée. Les paramètres des primitives de service sont énumérés dans le Tableau 38.

Tableau 38 – PDInStatus

Nom du paramètre	.req	.ind
Argument État	M	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

État

Ce paramètre contient l'indication de validité des Données de Processus d'entrée transmises.

Valeurs admises:
VALID (Données de Processus d'entrée valides sur la base du drapeau indicateur d'état des PD (voir A.1.5); voir 7.2.1.18).
INVALID (Données de Processus d'entrée non valides).

7.2.2.6 MHInfo

Le service MHInfo notifie un fonctionnement exceptionnel dans le module de traitement des messages. Les paramètres du service sont énumérés dans le Tableau 39.

Tableau 39 – MHInfo

Nom du paramètre	.ind
Argument MHInfo	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

MHInfo

Ce paramètre contient l'indication d'une situation exceptionnelle du module de traitement des messages.

Valeurs admises:
COMLOST (perte de communication),
ILLEGAL_MESSAGETYPE (détection d'un type de séquence M imprévu)
CHECKSUM_MISMATCH (détection d'une erreur de somme de contrôle)

7.2.2.7 ODTrig

Le service ODTrig est disponible uniquement au niveau du Maître. Le service déclenche le module de traitement de Données à la Demande ainsi que les modules de traitement des ISDU, des Commandes ou d'Événements actuellement chargés de fournir les Données à la Demande (au moyen du service OD) pour le prochain message de Maître. Les paramètres du service sont énumérés dans le Tableau 40.

Tableau 40 – ODTrig

Nom du paramètre	.ind
Argument DataLength	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

DataLength

Ce paramètre contient l'espace disponible, par message, pour les Données à la Demande (OD).

7.2.2.8 PDTrig

Le service PDTrig est disponible uniquement au niveau du Maître. Le service déclenche le module de traitement des Données de Processus qui fournit les Données de Processus (PD) pour le prochain message de Maître.

Les paramètres du service sont énumérés dans le Tableau 41.

Tableau 41 – PDTrig

Nom du paramètre	.ind
Argument DataLength	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

DataLength

Ce paramètre contient l'espace disponible, par message, pour les Données de Processus (PD).

7.3 Protocole de la couche liaison de données

7.3.1 Vue d'ensemble

La Figure 28 et la Figure 29 représentent la structure de la couche de liaison de données et de ses constituants; un module de traitement de mode DL, un module de traitement des messages, un module de traitement des Données de Processus, et un module de traitement de Données à la Demande permettant d'assurer les services spécifiés. Dans les paragraphes 7.3.2 à 7.3.8, le comportement (dynamique) de ces modules de traitement est défini par des diagrammes d'états UML et des tables de transitions.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le module de traitement de Données à la Demande prend en charge trois types indépendants de données: ISDU, commande et Événement. De ce fait, trois diagrammes d'états supplémentaires collaborent avec le diagramme d'états du module de traitement de Données à la Demande, comme cela est représenté à la Figure 30.

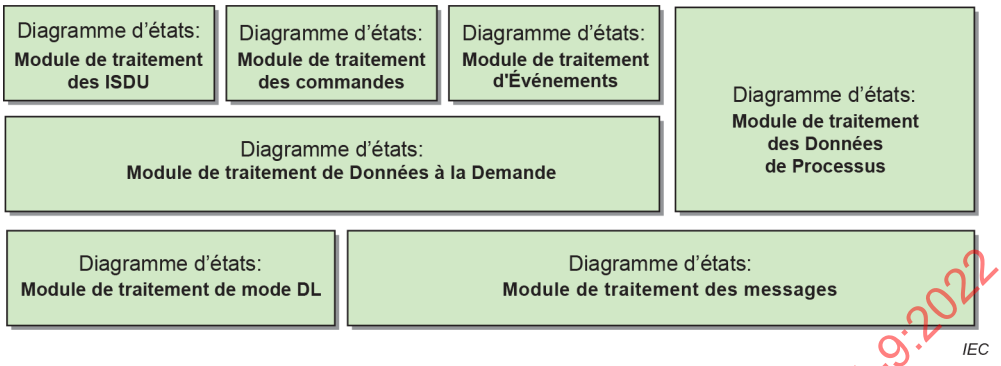


Figure 30 – Diagrammes d'états de la couche de liaison de données

Des diagrammes de séquence ou d'activité complémentaires présentent certains cas d'usage spécifiques. Voir l'IEC TR 62390 [14] et l'ISO/IEC 19505 (toutes les parties) [19].

Les éléments couverts par chaque module de traitement, tels que les messages, les procédures de réactivation, le mode d'entrelacement, les ISDU (Unités de données de service indexées) et les Événements sont définis dans le contexte du module de traitement correspondant.

7.3.2 Module de traitement de mode DL

7.3.2.1 Généralités

Le module de traitement de mode DL du Maître, représenté à la Figure 28 est chargé de la communication SDCI au moyen des services de la Couche Physique (PL) et des invocations réglementaires internes pour contrôler et surveiller le module de traitement des messages ainsi que les états des autres modules de traitement.

Le module de traitement de mode DL du Dispositif représenté à la Figure 29 est chargé de la détection d'une demande de réactivation et de l'établissement de la communication. Il reçoit des MasterCommands pour se synchroniser avec les états du module de traitement de mode DL du Maître STARTUP, PREOPERATE et OPERATE et gère l'activation et la désactivation des modules de traitement en fonction des besoins.

7.3.2.2 Procédures de réactivation et règles de conformité du Dispositif

La Gestion Système déclenche les actions suivantes sur la couche de liaison de données, à l'aide du service DL_SetMode (mode demandé = STARTUP).

Le module de traitement de mode DL du Maître tente d'établir la communication au moyen d'une demande de réactivation (PL_WakeUp.req) suivie par un message d'essai avec une séquence M de TYPE_0 (lecture de "MinCycleTime") conformément à la séquence représentée à la Figure 31.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

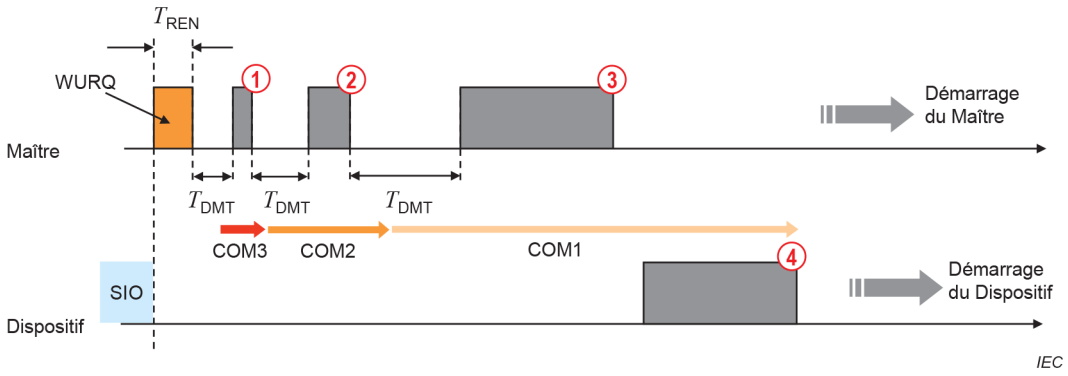


Figure 31 – Exemple d'une tentative d'établissement de communication

Après la demande de réactivation (WURQ), spécifiée en 5.3.3.3, le module de traitement de mode DL demande au module de traitement des messages d'envoyer le premier message d'essai au bout d'une période T_{REN} (voir le Tableau 10) et T_{DMT} (voir le Tableau 42). Les vitesses de transmission spécifiées COM1, COM2 et COM3 sont utilisées par ordre décroissant jusqu'à ce qu'une réponse soit obtenue, comme cela est représenté dans l'exemple de la Figure 31:

Étape ①: Message de Maître à une vitesse de transmission de COM3 (voir le Tableau 9).

Étape ②: Message de Maître à une vitesse de transmission de COM2 (voir le Tableau 9).

Étape ③: Message de Maître à une vitesse de transmission de COM1 (voir le Tableau 9).

Étape ④: Message de réponse du Dispositif à une vitesse de transmission de COM1.

Avant de lancer un (nouveau) message, le module de traitement de mode DL doit attendre pendant une période d'au moins T_{DMT}. T_{DMT} est spécifiée dans le Tableau 42.

En ce qui concerne la prise en charge des vitesses de transmission, la règle de conformité suivante s'applique aux Dispositifs:

- un Dispositif doit prendre en charge uniquement l'une des vitesses de transmission de COM1, COM2 ou COM3.

En cas d'échec d'une tentative d'établissement de communication, le module de traitement de mode DL du Maître ne doit lancer une nouvelle tentative (procédure de réactivation) qu'après un délai T_{DWU}, comme cela est représenté à la Figure 32 et comme cela est spécifié dans le Tableau 42.

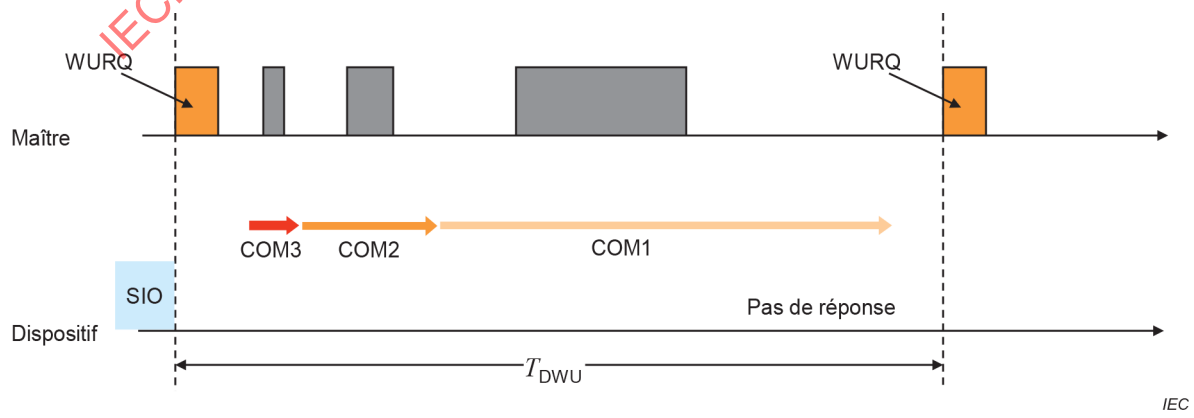


Figure 32 – Échec d'une tentative d'établissement de communication

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le Maître doit effectuer jusqu'à n_{WU}+1 demandes de réactivation successives, comme cela est représenté à la Figure 33. Si cette séquence initiale de nouvelle tentative de réactivation échoue, le Dispositif doit réinitialiser sa ligne C/Q en mode SIO après une période T_{DSIO} (T_{DSIO} est redéclenchée dans le Dispositif après chaque détection de WURQ). Le Maître ne doit déclencher une nouvelle séquence de tentative de réactivation qu'au bout d'un délai T_{SD}.

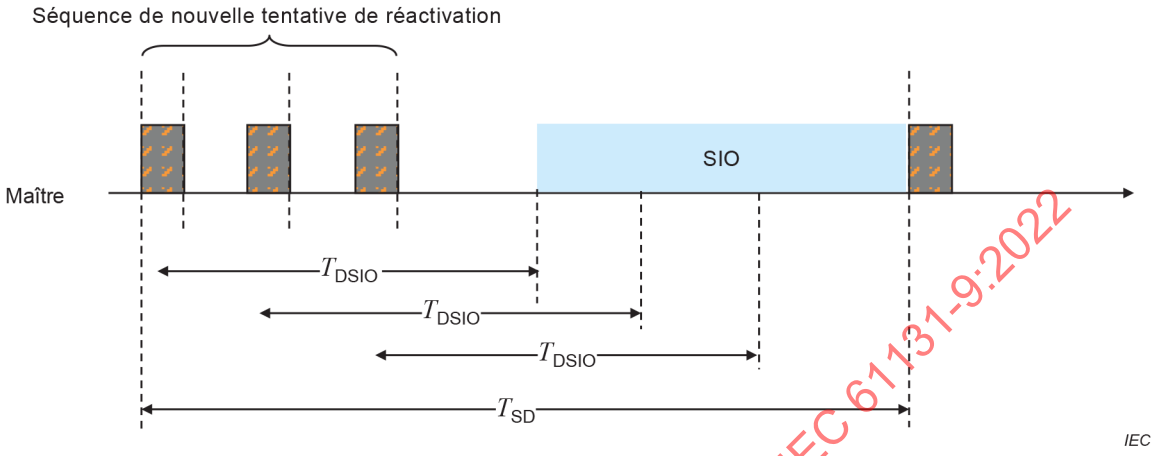


Figure 33 – Stratégie de nouvelle tentative d'établissement de communication

La couche DL du Maître doit demander à la PL de passer en mode SIO après échec d'une séquence de nouvelle tentative de réactivation.

Les valeurs de temporisation de la procédure et des nouvelles tentatives de réactivation sont spécifiées dans le Tableau 10 ainsi que dans le Tableau 42. Elles sont définies du point de vue du Maître.

Tableau 42 – Caractéristiques des procédures et des nouvelles tentatives de réactivation

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
T_{DMT}	Délai du message du Maître	27	n/a	37	T_{BIT}	Temps binaire de la vitesse de transmission de données ultérieure
T_{DSIO}	Délai E/S normalisé	60	n/a	300	ms	Au bout de $T_{DSIO,}$ le dispositif se replie en mode SIO (s'il est pris en charge)
T_{DWU}	Délai d'une nouvelle tentative de réactivation	30	n/a	50	ms	Au bout de $T_{DWU,}$ le Maître répète la demande de réactivation
n_{WU}	Nombre de nouvelles tentatives de réactivation	2	2	2		Nombre de nouvelles tentatives de demande de réactivation
T_{SD}	Temps de détection du Dispositif	0,5	n/a	1	s	Temps entre 2 séquences de demande de réactivation. (Voir la NOTE)
NOTE Caractéristique du Maître.						

La couche de liaison de données du Maître doit arrêter la procédure d'établissement de la communication lorsqu'elle trouve un Dispositif communiquant et doit rendre compte du mode COMx détecté à la Gestion Système au moyen d'une indication DL_Mode. Si la procédure échoue, l'erreur correspondante est notifiée en utilisant le même service.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.2.3 Procédure de repli

La Gestion Système induit les actions suivantes sur la couche de liaison de données, à l'aide du service DL_SetMode (mode = INACTIVE):

– une MasterCommand "Repli" (voir le Tableau B.2) force le Dispositif à passer en mode SIO;
– le Dispositif doit effectuer la transition vers le mode SIO après trois durées MasterCycleTimes et/ou dans les limites maximales de T_{FBD} après la commande MasterCommand "Repli". Ceci tient compte des éventuelles tentatives en cas d'échec de la MasterCommand, indiquée par une réponse négative du Dispositif;
– le Maître doit attendre au moins les limites maximales de T_{FBD} avant de lancer la procédure de démarrage suivante.

La Figure 34 représente la procédure de repli ainsi que les contraintes liées aux nouvelles tentatives et à la temporisation.

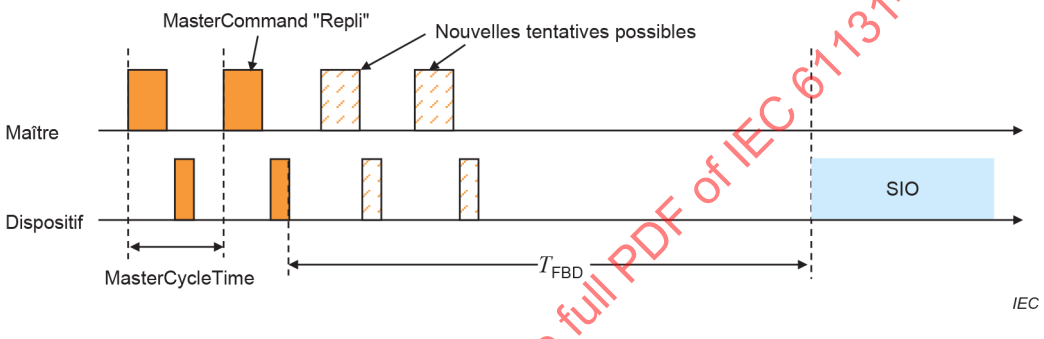


Figure 34 – Procédure de repli

Le Tableau 43 définit les caractéristiques temporelles du repli. Une description plus détaillée est fournie en A.2.6.

Tableau 43 – Caractéristiques temporelles du repli

Propriété	Désignation	Minimale	Type	Maximale	Unité	Remarque
T_{FBD}	Délai de repli	3 MasterCycle-Times (OPERATE) ou 3 $T_{initcyc}$ (PREOPERATE)	n/a	500	ms	Au bout d'un temps T_{FBD} le Dispositif doit passer en mode SIO (voir la Figure 34)

7.3.2.4 Diagramme d'états du module de traitement de mode DL du Maître

La Figure 35 représente le diagramme d'états du module de traitement de mode DL du Maître.

NOTE Les conventions utilisées dans les types de diagrammes UML sont définies en 3.3.8.

Après réception du service DL_SetMode_STARTUP de la Gestion Système, le module de traitement de mode DL doit d'abord générer une impulsion de courant de réactivation au moyen du service PL_WakeUp puis établir la communication. Cette procédure est spécifiée dans la sous-machine 1 à la Figure 36.

L'état "Startup_2" a pour but de vérifier l'identité d'un Dispositif en utilisant les données de la page de Paramètres Directs (voir la Figure 6). Lorsqu'il est dans l'état "PreOperate_3", le Maître attribue des paramètres au Dispositif en utilisant des ISDU. L'échange cyclique de Données de Processus est réalisé au cours de l'état "Operate". Dans cet état, des Données à la Demande

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

supplémentaires, telles que des ISDU, des commandes et des Événements peuvent être transmises en utilisant des types de séquences M appropriés (voir la Figure 39).

Dans les états PreOperate_3 et Operate_4, différents ensembles de modules de traitement sont activés dans le Maître.

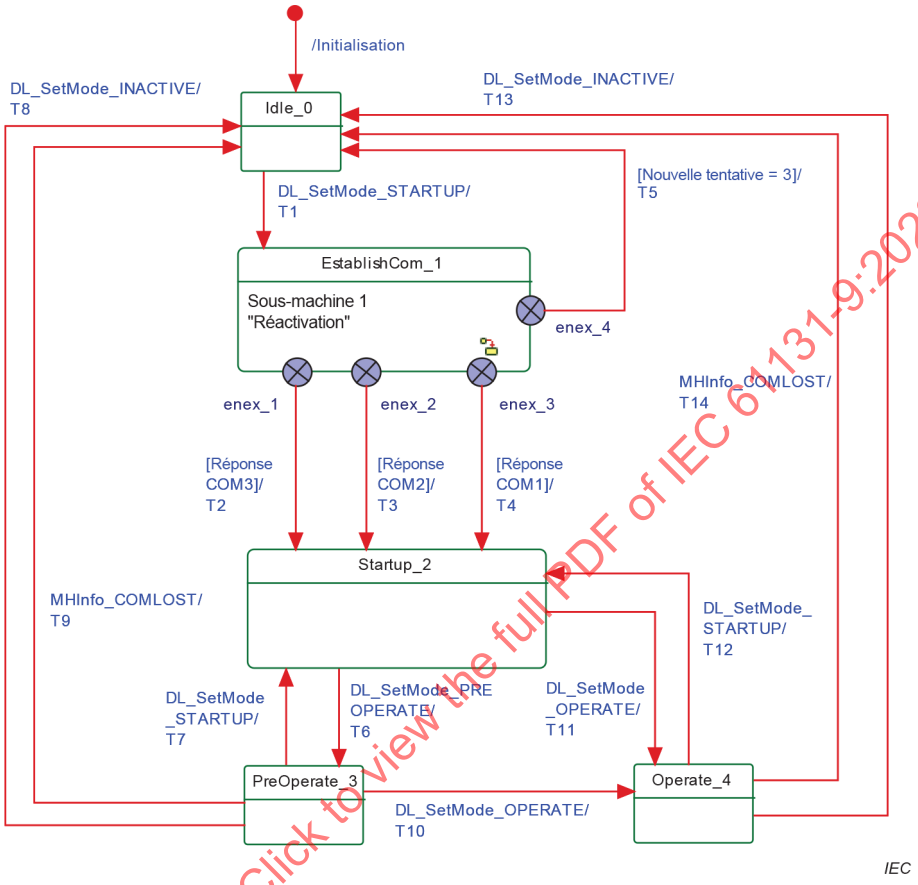


Figure 35 – Diagramme d'états du module de traitement de mode DL du Maître

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

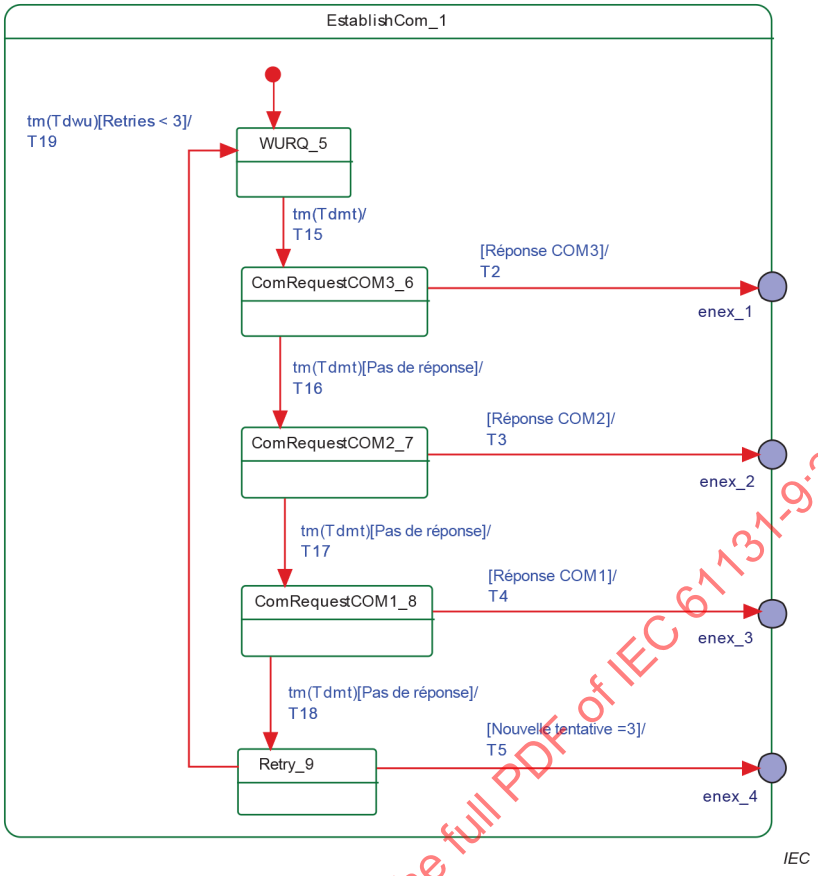


Figure 36 – Sous-machine 1 d'établissement de la communication

Le Tableau 44 présente la table de transitions d'états du module de traitement de mode DL du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 44 – Table de transitions d'états du module de traitement de mode DL du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Idle_0	Attente d'une demande de réactivation émise par la Gestion Système (SM): DL_SetMode (STARTUP)
EstablishComm_1	Exécution de la procédure de réactivation (sous-machine 1)
Startup_2	La Gestion Système utilise l'état STARTUP pour l'identification du Dispositif, la vérification et la configuration de la communication (voir la Figure 71)
Preoperate_3	Échange de Données à la Demande (paramètre, commandes, Événements) sans Données de Processus
Operate_4	Échange de Données de Processus et de Données à la Demande (paramètre, commandes, Événements)
SM: WURQ_5	Génération de l'impulsion de courant de réactivation: Invocation du service PL-Wake-Up (voir la Figure 12 et 5.3.3.3) et attente de T_{DMT} (voir le Tableau 42).
SM: ComRequestCOM3_6	Tentative d'envoi du message d'essai à une vitesse de transmission de COM3 au moyen du module de traitement des messages: Invocation de MH_Conf_COMx (voir la Figure 40) et attente de T_{DMT} (voir le Tableau 42).
SM: ComRequestCOM2_7	Tentative d'envoi du message d'essai à une vitesse de transmission de COM2 au moyen du module de traitement des messages: Invocation de MH_Conf_COMx (voir la Figure 40) et attente de T_{DMT} (voir le Tableau 42).
SM: ComRequestCOM1_8	Tentative d'envoi du message d'essai à une vitesse de transmission de COM1 au moyen du module de traitement des messages: Invocation de MH_Conf_COMx (voir la Figure 40) et attente de T_{DMT} (voir le Tableau 42).
SM: Retry_9	Vérification du nombre de nouvelles tentatives

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Set Retry = 0.
T2	1	2	Vitesse de transmission COM3 réussie. Module de traitement des messages activé et configuré en COM3 (voir la Figure 40, Transition T2). Activation du module de traitement des commandes (invocation de CH_Conf_ACTIVE à la Figure 53). Renvoi de DL_Mode.ind (STARTUP) et de DL_Mode.ind (COM3) à la SM.
T3	1	2	Vitesse de transmission COM2 réussie. Module de traitement des messages activé et configuré en COM2 (voir la Figure 40, Transition T2). Activation du module de traitement des commandes (invocation de CH_Conf_ACTIVE à la Figure 53). Renvoi de DL_Mode.ind (STARTUP) et de DL_Mode.ind (COM2) à la SM.
T4	1	2	Vitesse de transmission COM1 réussie. Module de traitement des messages activé et configuré en COM1 (voir la Figure 40, Transition T2). Activation du module de traitement des commandes (invocation de CH_Conf_ACTIVE à la Figure 53). Renvoi de DL_Mode.ind (STARTUP) et de DL_Mode.ind (COM1) à la SM.
T5	1	0	Renvoi de DL_Mode.ind (INACTIF) à la SM.
T6	2	3	La SM a demandé l'état PREOPERATE. Activation du module de traitement de Données à la Demande (invocation de OH_Conf_ACTIVE à la Figure 48), du module de traitement des ISDU (invocation de IH_Conf_ACTIVE à la Figure 51), et du module de traitement d'Événements (invocation de EH_Conf_ACTIVE à la Figure 55). Passage de l'état du module de traitement des messages à PREOPERATE (invocation de MH_Conf_PREOPERATE à la Figure 40). Renvoi de DL_Mode.ind (PREOPERATE) à la SM.
T7	3	2	La SM a demandé l'état STARTUP. Passage de l'état du module de traitement des messages à STARTUP (invocation de MH_Conf_STARTUP à la Figure 40). Désactivation du module de traitement des Données à la Demande (invocation de OH_Conf_INACTIVE à la Figure 48), ISDU (invocationl IH_Conf_IN-ACTIVE de la Figure 51) et Gestionnaire d'événements (invocation EH_Conf_INACTIVE de la Figure 55). Retour de DL_Mode.ind (STARTUP) au SM.
T8	3	0	La SM a demandé le mode SIO. Désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Renvoi de DL_Mode.ind (INACTIF) à la SM. Voir 7.3.2.3.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T9	3	0	Le module de traitement des messages notifie la perte de communication au moyen du service de DL-A MHInfo (COMLOST). Désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Renvoi de DL_Mode.ind (COMLOST) à la SM.
T10	3	4	La SM a demandé l'état OPERATE. Activation du module de traitement des Données de Processus (invocation de PD_Conf_SINGLE si le type de séquence M = TYPE_2_x, ou PD_Conf_INTERLEAVE si le type de séquence M = TYPE_1_1 à la Figure 46). Passage de l'état du module de traitement des messages à OPERATE (invocation de MH_Conf_OPERATE à la Figure 40). Renvoi de DL_Mode.ind (OPERATE) à la SM.
T11	2	4	La SM a demandé l'état OPERATE. Activation du module de traitement des Données de Processus (invocation de PD_Conf_SINGLE ou PD_Conf_INTERLEAVE à la Figure 46, selon la configuration de l'Accès du Maître). Activation des Données à la Demande (invocation de OH_Conf_ACTIVE à la Figure 48), ISDU (invocation de IH_Conf_ACTIVE à la Figure 51), et du module de traitement d'Événements (invocation de EH_Conf_ACTIVE à la Figure 55). Passage de l'état du module de traitement des messages à OPERATE (invocation de MH_Conf_OPERATE à la Figure 40). Renvoi de DL_Mode.ind (OPERATE) à la SM.
T12	4	2	La SM a demandé l'état STARTUP. Passage de l'état du module de traitement des messages à STARTUP (invocation de MH_Conf_STARTUP à la Figure 40). Désactivation des Données de Processus (invocation de PD_Conf_INACTIVE à la Figure 46), des Données à la Demande (invocation de OH_Conf_INACTIVE à la Figure 48), des ISDU (invocation de IH_Conf_INACTIVE à la Figure 51), et du module de traitement d'Événements (invocation de EH_Conf_INACTIVE à la Figure 55). Renvoi de DL_Mode.ind (STARTUP) à la SM.
T13	4	0	La SM a demandé l'état SIO. Désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Renvoi de DL_Mode.ind (INACTIF) à la SM. Voir 7.3.2.3.
T14	4	0	Le module de traitement des messages notifie la perte de communication au moyen du service de DL-A MHInfo (COMLOST). Désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Renvoi de DL_Mode.ind (COMLOST) à la SM.
T15	5	6	Établissement de la vitesse de transmission au mode COM3.
T16	6	7	Établissement de la vitesse de transmission au mode COM2.
T17	7	8	Établissement de la vitesse de transmission au mode COM1.
T18	8	9	Incrémentation du nombre de Nouvelles tentatives
T19	9	5	-

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
MH_Conf_COMx	Invocation	Cette invocation force le module de traitement des messages à envoyer un message à la vitesse de transmission COMx demandée et avec une séquence M de TYPE_0 (voir le Tableau 46).
MH_Conf_STARTUP	Invocation	Cette invocation force le module de traitement des messages à passer à l'état STARTUP (voir la Figure 40)
MH_Conf_PREOPERATE	Invocation	Cette invocation force le module de traitement des messages à passer à l'état PREOPERATE (voir la Figure 40)
MH_Conf_OPERATE	Invocation	Cette invocation force le module de traitement des messages à passer à l'état OPERATE (voir la Figure 40)
xx_Conf_ACTIVE	Invocation	Ces invocations activent le module de traitement correspondant, où xx remplace MH (module de traitement des messages), OH (module de traitement de Données à la Demande), IH (module de traitement des ISDU), CH (module de traitement des commandes) et/ou EH (module de traitement d'Événements)
xx_Conf_INACTIVE	Invocation	Ces invocations désactivent le module de traitement respectif, où xx remplace MH (module de traitement des messages), OH (module de traitement de Données à la Demande), IH (module de traitement des ISDU), CH (module de traitement des commandes) et/ou EH (module de traitement d'Événements)
Retry (Nouvelle tentative)	Variable	Nombre de nouvelles tentatives pour établir la communication

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.2.5 Diagramme d'états du module de traitement de mode DL du Dispositif

La Figure 37 représente le diagramme d'états du module de traitement de mode DL du Dispositif.

Dans les états PreOperate_3 et Operate_4, différents ensembles de modules de traitement sont activés dans le Dispositif.

Le Maître utilise des MasterCommands (voir le Tableau 44) pour modifier les états du Dispositif en SIO, STARTUP, PREOPERATE et OPERATE.

Lorsque le module de traitement des messages détecte des types de séquence M illicites (imprévues), il force le module de traitement de mode DL à passer à l'état STARTUP et à indiquer cet état à sa gestion système (voir 9.3.3.2) aux fins de synchronisation du Maître et du Dispositif.

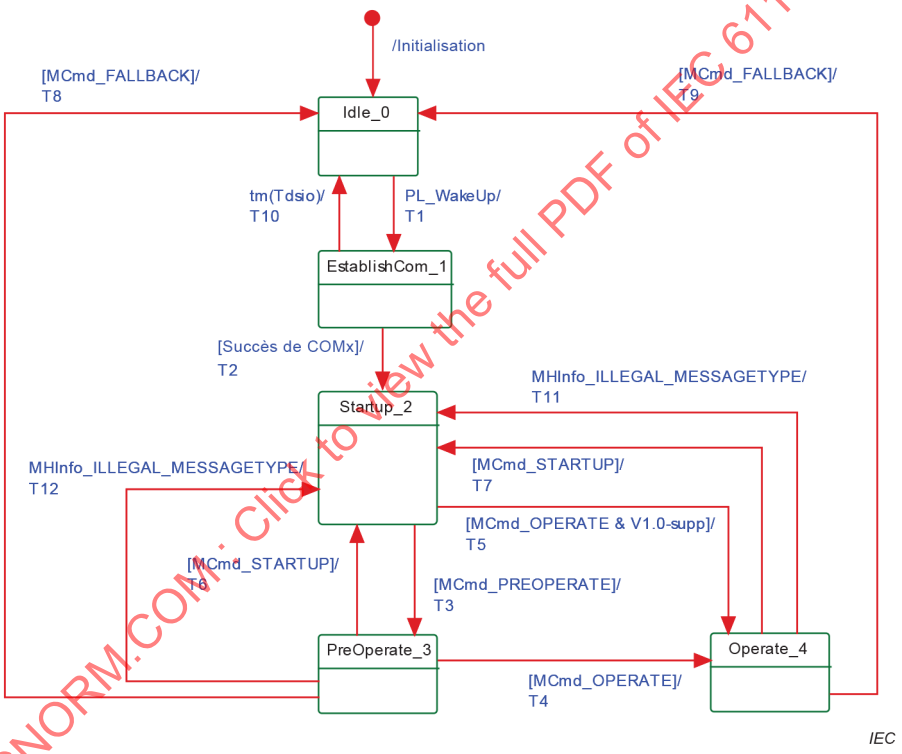


Figure 37 – Diagramme d'états du module de traitement de mode DL du Dispositif

Le Tableau 45 présente la table de transitions d'états du module de traitement de mode DL du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 45 – Table de transitions d'états du module de traitement de mode DL du Dispositif

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Idle_0	Attente de détection d'une impulsion de courant de réactivation (PL_WakeUp.ind).
EstablishComm_1	Module de traitement des messages activé et attente de messages d'essai COMx (voir le Tableau 44)
Startup_2	Vérification de compatibilité (voir 9.2.3.3). Les Dispositifs qui ne prennent pas en charge un Maître selon [8] restent à l'état STARTUP, prenant donc en charge d'autres identifications, mais aucun échange de données de processus dans ce cas.
Preoperate_3	Échange de Données à la Demande (paramètre, commandes, Événements) sans Données de Processus
Operate_4	Échange de Données de Processus (PD) et de Données à la Demande (paramètre, commandes, Événements)

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Impulsion de courant de réactivation détectée. Activation du module de traitement des messages (invocation de MH_Conf_ACTIVE à la Figure 44). Indication de l'état, au moyen du service DL_Mode.ind (ESTABCOM), à la SM.
T2	1	2	Une des trois vitesses de transmission (COM3, COM2 ou COM1) est établie. Activation du module de traitement de Données à la Demande (invocation de OH_Conf_ACTIVE à la Figure 49) et du module de traitement de commandes (invocation de CH_Conf_ACTIVE à la Figure 54). Indication de l'état, au moyen du service DL_Mode.ind (COM1, COM2 ou COM3), à la SM.
T3	2	3	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_PREOPERATE). Activation du module de traitement des ISDU (invocation de IH_Conf_ACTIVE à la Figure 52) et du module de traitement d'Événements (invocation de EH_Conf_ACTIVE à la Figure 56). Indication de l'état, au moyen du service DL_Mode.ind (PREOPERATE), à la SM.
T4	3	4	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_OPERATE). Activation du module de traitement des Données de Processus (invocation de PD_Conf_ACTIVE à la Figure 47). Indication de l'état, au moyen du service DL_Mode.ind (OPERATE), à la SM.
T5	2	4	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_OPERATE). Activation du module de traitement des Données de Processus (invocation de PD_Conf_ACTIVE à la Figure 47), du module de traitement des ISDU (invocation de IH_Conf_ACTIVE à la Figure 52) et du module de traitement d'Événements (invocation de EH_Conf_ACTIVE à la Figure 56). Indication de l'état, au moyen du service DL_Mode.ind (OPERATE), à la SM.
T6	3	2	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_STARTUP). Désactivation du module de traitement des ISDU (invocation de IH_Conf_INACTIVE à la Figure 52) et du module de traitement d'Événements (invocation de EH_Conf_INACTIVE à la Figure 56). Indication de l'état, au moyen du service DL_Mode.ind (STARTUP), à la SM.
T7	4	2	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_STARTUP). Désactivation du module de traitement des Données de Processus (invocation de PD_Conf_INACTIVE à la Figure 47), du module de traitement des ISDU (invocation de IH_Conf_INACTIVE à la Figure 52) et du module de traitement d'Événements (invocation de EH_Conf_INACTIVE à la Figure 56). Indication de l'état, au moyen du service DL_Mode.ind (STARTUP), à la SM.
T8	3	0	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_FALLBACK). Attente de l'expiration de T_{FBD} puis désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Indication de l'état, au moyen du service DL_Mode.ind (INACTIVE) à la SM (voir la Figure 81 et le Tableau 95).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T9	4	0	Le module de traitement de commandes de dispositif a reçu du Maître la commande MasterCommand (MCmd_FALLBACK). Attente de l'expiration de T_{FBD} puis désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Indication de l'état, au moyen du service DL_Mode.ind (INACTIVE) à la SM (voir la Figure 81 et le Tableau 95).
T10	1	0	Après des échecs de procédures de réactivation (voir la Figure 32), le Dispositif établit le mode SIO configuré, à l'expiration d'un délai T_{DSIO} (voir la Figure 33). Désactivation de tous les modules de traitement (Invocation de xx_Conf_INACTIVE). Indication de l'état, au moyen du service DL_Mode.ind (INACTIVE), à la SM.
T11	4	2	Le module de traitement des messages a détecté un type de séquence M illicite. Désactivation du module de traitement des Données de Processus (invocation de PD_Conf_INACTIVE à la Figure 47), du module de traitement des ISDU (invocation de IH_Conf_INACTIVE à la Figure 52) et du module de traitement d'Événements (invocation de EH_Conf_INACTIVE à la Figure 56). Indication de l'état, au moyen du service DL_Mode.ind (STARTUP), à la SM (voir la Figure 81 et le Tableau 95).
T12	3	2	Le module de traitement des messages a détecté un type de séquence M illicite. Désactivation du module de traitement des ISDU (invocation de IH_Conf_INACTIVE à la Figure 52) et du module de traitement d'Événements (invocation de EH_Conf_INACTIVE à la Figure 56). Indication de l'état, au moyen du service DL_Mode.ind (STARTUP) à la SM (voir la Figure 81 et le Tableau 95).

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
T_{FBD}	Temps	Voir le Tableau 43
T_{DSIO}	Temps	Voir la Figure 33
MCmd_XXXXXXX	Invocation	Toute MasterCommand reçue par le module de traitement des commandes du Dispositif (voir le Tableau 44 et la Figure 54, état "CommandHandler_2")
V1.0-supp	Flag	Le Dispositif prend en charge le mode V1.0

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.3 Module de traitement des messages

7.3.3.1 Généralités

Le rôle du module de traitement des messages est spécifié en 7.1 et 7.2.2.1. Le paragraphe 7.3.3 définit la structure et les types de séquences M ainsi que le comportement (dynamique) du module de traitement des messages.

7.3.3.2 Séquences M

Un Maître et son Dispositif échangent des données au moyen d'une séquence de messages (séquence M). Comme cela est représenté à la Figure 38, une séquence M comprend un message qui provient du Maître, suivi d'un message qui provient du Dispositif. Chaque message est constitué de trames UART.

Tous les types de données à octets multiples doivent être transmis sous forme de séquence "gros-boutiste", c'est-à-dire que l'octet de poids fort (MSO - most significant octet) doit être envoyé en premier, suivi par des octets de poids faible en ordre décroissant, l'octet de poids faible (LSO - least significant octet) étant envoyé en dernier, comme cela est représenté à la Figure 2.

Le message du Maître commence par l'octet MC ("M-sequence Control" – "Contrôle de séquence M") suivi par l'octet CKT ("CHECK/TYPE" – Vérification/Type) et, de manière facultative, par des octets PD ("Process Data" - "Données de processus") et/ou des octets OD ("On-request Data" - "Données à la Demande"). Le message de Dispositif commence

optionnellement par des octets PD "Données de Processus" et/ou des octets OD "Données à la Demande", suivis par l'octet CKS ("CHECK/STAT").

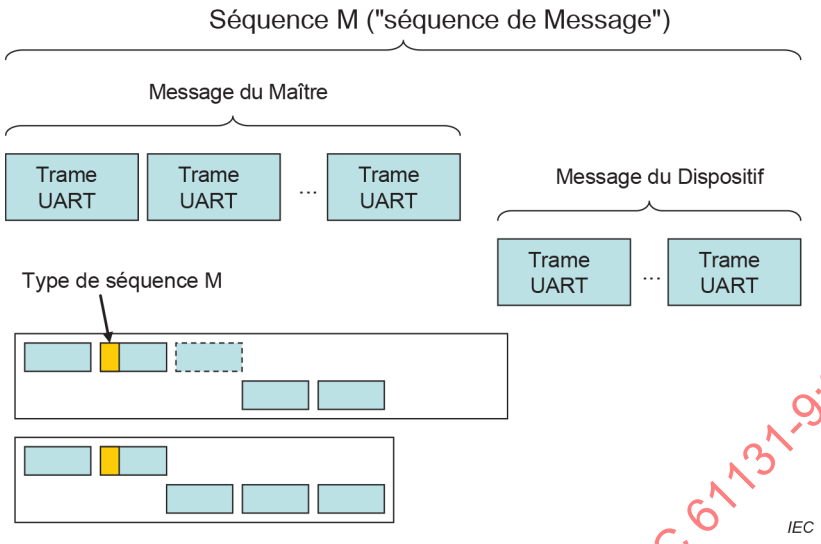


Figure 38 – Séquences de messages SDCI

Différents types de séquences M peuvent être sélectionnés pour satisfaire aux besoins particuliers d'un actionneur ou d'un capteur (vitesse de balayage, quantité de Données de Processus). La longueur des messages du Maître et du Dispositif peut varier en fonction du type de message et du sens de transmission des données, voir la Figure 38.

La Figure 39 donne une vue d'ensemble des types de séquences M définis. Les parties représentées par des lignes en pointillés dépendent du sens, lecture ou écriture, dans l'octet de contrôle de séquence M.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

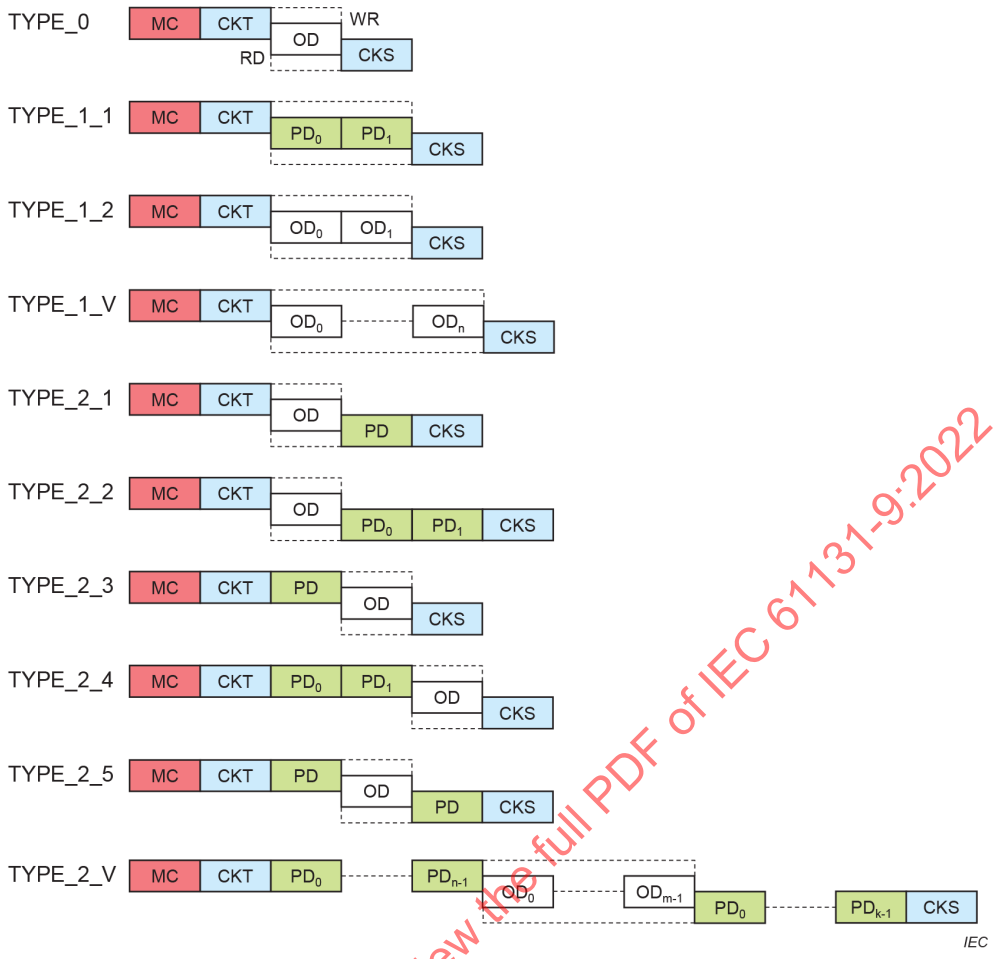


Figure 39 – Vue d'ensemble des types de séquence M

Les types de séquences M fixes sont le TYPE_0, le TYPE_1_1, le TYPE_1_2, ainsi que les TYPE_2_1 à TYPE_2_5. Attention: Le TYPE_2_6 n'est plus pris en charge. Les types de séquences M variables sont le TYPE_1_V et le TYPE_2_V.

Les différents types de séquences M satisfont aux différentes exigences des capteurs et actionneurs en matière de taille de Données de Processus et de conditions correspondantes. Pour plus de précisions sur les types de séquences M, voir l'Article A.2. Les contraintes temporelles applicables aux séquences M sont décrites à l'Article A.3.

7.3.3.3 Contraintes liées au MasterCycleTime

À l'état STARTUP et à l'état PREOPERATE, un Dispositif est capable de communiquer de manière acyclique. Pour détecter la déconnexion des Dispositifs, il est fortement recommandé que le Maître effectue à partir de là une communication périodique ("message d'entretien") par l'intermédiaire de séquences M acycliques à travers la couche de liaison de données. Les durées minimales de récupération pour la communication acyclique, spécifiées en A.2.6, doivent être prises en considération.

Après ces phases, la communication cyclique de Données de Processus peut être lancée par le Maître au moyen du service DL_SetMode (OPERATE). Les types de séquences M pour les échanges de données cycliques doivent être utilisés au cours de cette phase de communication pour échanger avec un Dispositif des Données de Processus (PD) et des Données à la Demande (voir le Tableau A.9 et le Tableau A.10).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le Maître doit utiliser pour le temps t_{CYC} la valeur indiquée dans le paramètre de Dispositif "MasterCycleTime" (voir le Tableau B.1) avec une tolérance relative comprise entre −1 % et +10 % (y compris la gigue).

Dans les cas où le Dispositif doit être ramené au mode SIO après paramétrage, le Maître doit envoyer une commande de "Repli" (voir le Tableau B.2), suivie par une confirmation du Dispositif.

7.3.3.4 Diagramme d'états du module de traitement des messages du Maître

La Figure 40 représente le diagramme d'états du module de traitement des messages du Maître. Trois sous-machines décrivant les réactions en cas d'erreurs de communication sont représentées à la Figure 41, à la Figure 42 et à la Figure 43.

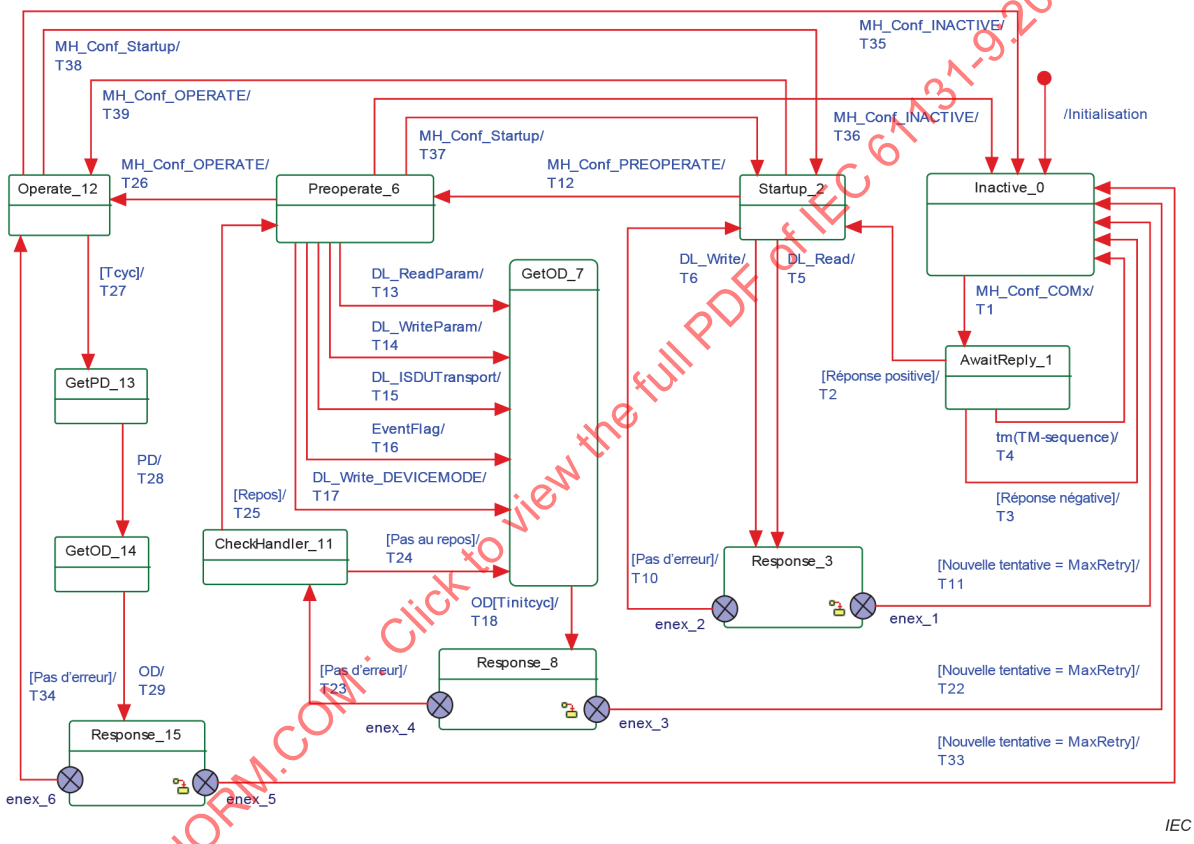


Figure 40 – Diagramme d'états du module de traitement des messages du Maître

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le module de traitement des messages prend en charge les exigences de communication particulières dans les états "EstablishCom", "Startup", "PreOperate" et "Operate" du module de traitement de mode DL. Une invocation réglementaire interne MH_Conf_COMx à l'état "Inactive_0" force le module de traitement des messages à envoyer des messages "test" avec une séquence M de TYPE_0 et différentes vitesses de transmission (COM3, COM2 ou COM1) au cours de la séquence d'établissement de communication.

L'état "Startup_2" fournit tous les moyens de communication nécessaires à la prise en charge des contrôles d'identité de la Gestion Système à l'aide des services DL_Read et DL_Write. Le module de traitement des messages attend l'invocation de ces services pour envoyer et recevoir des messages (communication acyclique). L'état "Preoperate_6" est le point de reprise de toutes les activités relatives aux Données à la Demande, telles que les ISDU, les commandes et les Événements pour le paramétrage du Dispositif. Le module de traitement des messages attend l'invocation des services représentés à la Figure 40 pour envoyer et recevoir des messages (communication acyclique). L'état "Operate_12" est le point de reprise de l'échange

cyclique de Données de Processus. En fonction du type de séquence M, le module de traitement des messages génère des messages du Maître avec les Données de Processus acquises auprès du module de traitement des Données de Processus, au moyen du service PD, et optionnellement les Données à la Demande acquises auprès du module de traitement des Données à la Demande, au moyen du service OD.

La Figure 41 représente la sous-machine de l'état "Response 3".

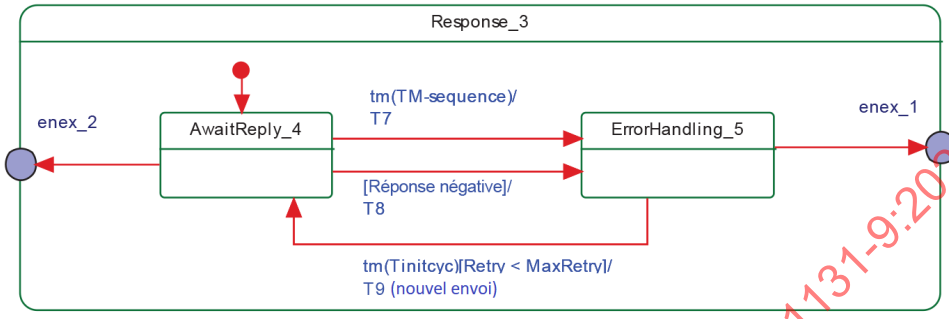


Figure 41 – Sous-machine "Response 3" du module de traitement des messages

La Figure 42 représente la sous-machine de l'état "Response 8".

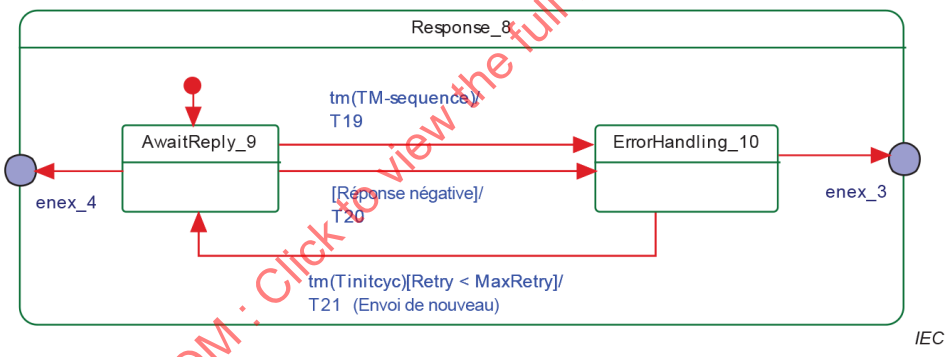


Figure 42 – Sous-machine "Response 8" du module de traitement des messages

La Figure 43 représente la sous-machine de l'état "Response 15".

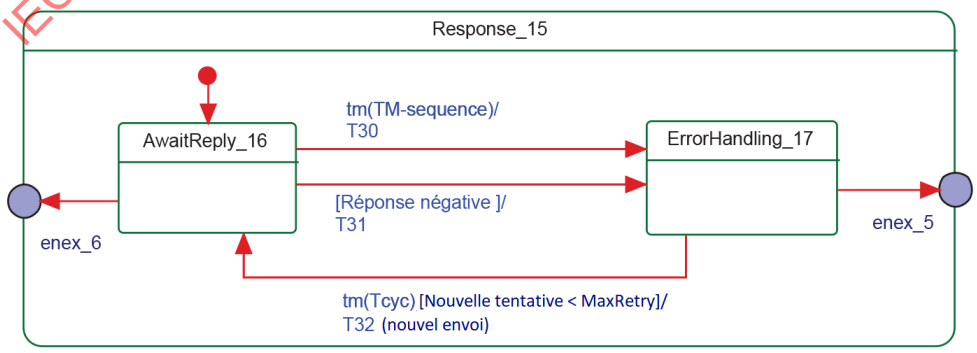


Figure 43 – Sous-machine "Response 15" du module de traitement des messages

Le Tableau 46 présente la table de transitions d'états du module de traitement des messages du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 46 – Table de transitions d'états du module de traitement des messages du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Inactive_0	Attente de demande d'un message "test" par l'intermédiaire de l'invocation de MH_Conf_COMx (voir la Figure 36 et le Tableau 44) en provenance du module de traitement de mode DL.
AwaitReply_1	Attente de réponse du Dispositif au message "test". Retour à l'état Inactive_0 si le temps de séquence $T_{\text{M-sequence}}$ s'est écoulé sans réponse du Dispositif ou si la réponse au message "test" n'a pas pu être décodée. En cas de réponse correcte du Dispositif, le module de traitement des messages passe à l'état Startup_2.
Startup_2	Lorsque cet état est atteint au moyen de la transition T2, il est chargé de contrôler l'échange acyclique de Données à la Demande selon les conditions spécifiées dans le Tableau A.7. Tout service DL_Write ou DL_Read en provenance de la Gestion Système génère une transition.
Response_3	Le service OD a entraîné l'envoi d'un message correspondant par le module de traitement de message. Dans ce pseudo-état, la sous-machine attend la réponse et vérifie qu'elle est correcte.
SM: AwaitReply_4	Cet état vérifie que le temps de séquence $T_{\text{M-sequence}}$ a expiré et que la réponse est correcte.
SM: ErrorHandling_5	Si elle est incorrecte, le module de traitement des messages renvoie le message au bout d'un temps d'attente T_{initcyc}. Si le nombre de nouvelles tentatives est trop élevé, le module de traitement des messages passe à l'état Inactive_0.
Preoperate_6	À la réception d'une invocation MH_Conf_PREOPERATE, le module de traitement des messages est passé dans cet état. Le module de traitement des messages est à présent chargé de contrôler l'échange acyclique de Données à la Demande selon les conditions spécifiées dans le Tableau A.8. Tout service DL_ReadParam, DL_WriteParam, DL_ISDUTransport, DL_Write ou EventFlag génère une transition.
GetOD_7	Le module de traitement des messages a utilisé le service ODTrig pour acquérir des OD auprès du module de traitement de Données à la Demande. Le module de traitement des messages attend le service OD pour envoyer un message au bout d'une période T_{initcyc}.
Response_8	Le service OD a entraîné l'envoi d'un message correspondant par le module de traitement des messages. Dans ce pseudo-état, la sous-machine attend la réponse et vérifie qu'elle est correcte.
SM: AwaitReply_9	Cet état permet de vérifier si le temps de séquence $T_{\text{M-sequence}}$ s'est écoulée et si la réponse est correcte.
SM: ErrorHandling_10	Si elle est incorrecte, le module de traitement des messages renvoie le message au bout d'un temps d'attente T_{initcyc}. Si le nombre de nouvelles tentatives est trop élevé, le module de traitement des messages passe à l'état Inactive_0.
CheckHandler_11	Certains services nécessitent plusieurs cycles d'acquisition d'OD pour échanger les OD. Lorsque le module de traitement des OD, des ISDU ou d'Événements concerné est revenu à l'état au repos (inactif), le module de traitement des messages peut quitter la boucle d'acquisition des OD.
Operate_12	À la réception d'une invocation MH_Conf_OPERATE, le module de traitement des messages est passé dans cet état, et au bout d'un temps initial T_{initcyc}, ce dernier est chargé de contrôler l'échange cyclique de Données de Processus et de Données à la Demande selon les conditions spécifiées dans le Tableau A.9 et le Tableau A.10. Le module de traitement des messages relance de lui-même un nouveau cycle de messages cycle après expiration de la période t_{CYC}.
GetPD_13	Le module de traitement des messages a utilisé le service PDTrig pour acquérir des PD auprès du module de traitement des Données de Processus. Le module de traitement des messages attend le service PD puis passe à l'état GetOD_14.
GetOD_14	Le module de traitement des messages a utilisé le service ODTrig pour acquérir des OD auprès du module de traitement de Données à la Demande. Le module de traitement des messages attend le service OD pour compléter les PD déjà acquises et envoyer un message contenant les PD/OD acquises.
Response_15	Le module de traitement des messages a envoyé un message contenant les PD/OD acquises. Dans ce pseudo-état, la sous-machine attend la réponse et vérifie qu'elle est correcte.
SM: AwaitReply_16	Cet état permet de vérifier si le temps de séquence $T_{\text{M-sequence}}$ s'est écoulée et si la réponse est correcte.
SM: ErrorHandling_17	Si elle est incorrecte, le module de traitement des messages renvoie le message au bout d'un temps d'attente t_{CYC}. Si le nombre de nouvelles tentatives est trop élevé, le module de traitement des messages passe à l'état Inactive_0.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Envoi d'un message à la vitesse de transmission COMx demandée et avec une séquence M de TYPE_0: Lecture de la page 1 de Paramètres Directs, adresse 0x02 ("MinCycleTime"), compilation en un contrôle de séquence M MC = 0xA2 (voir A.1.2). Démarrage du temporisateur avec le temps de séquence $T_{\text{M-sequence}}$
T2	1	2	Renvoi de la valeur de "MinCycleTime" par l'intermédiaire de la confirmation de service DL_Read.
T3	1	0	Réinitialisation du temporisateur ($T_{\text{M-sequence}}$).
T4	1	0	Réinitialisation du temporisateur ($T_{\text{M-sequence}}$).
T5	2	3	Envoi d'un message en utilisant la vitesse de transmission établie, le canal de communication de page, et l'option d'Accès en lecture (voir A.1.2). Démarrage du temporisateur avec le temps de séquence $T_{\text{M-sequence}}$.
T6	2	3	Envoi d'un message en utilisant la vitesse de transmission établie, le canal de communication de page, et l'option d'Accès en écriture (voir A.1.2). Démarrage du temporisateur avec le temps de séquence $T_{\text{M-sequence}}$.
T7	4	5	Réinitialisation du temporisateur ($T_{\text{M-sequence}}$).
T8	4	5	Réinitialisation du temporisateur ($T_{\text{M-sequence}}$).
T9	5	4	Retransmission du message au bout d'un temps $T_{\text{initcyc.}}$ Redémarrage du temporisateur avec le temps de séquence $T_{\text{M-sequence}}$.
T10	3	2	Renvoi de la confirmation de service DL_Read ou DL_Write respectivement à la Gestion Système.
T11	3	0	Le module de traitement des messages renvoie MH_Info (COMLOST) au module de traitement de mode DL.
T12	2	6	-
T13	6	7	Le module de traitement des messages invoque le service ODTrig pour le module de traitement des Données à la Demande (voir la Figure 48), qui est à l'état "ISDU_1". Dans cet état, il force le module de traitement des ISDU à fournir le service OD en correspondance au service DL_ReadParam (voir la Figure 51, Transition T13).
T14	6	7	Le module de traitement des messages invoque le service ODTrig pour le module de traitement des Données à la Demande (voir la Figure 48), qui est à l'état "ISDU_1". Dans cet état, il force le module de traitement des ISDU à fournir le service OD en correspondance au service DL_WriteParam (voir la Figure 51, Transition T13).
T15	6	7	Le module de traitement des messages invoque le service ODTrig pour le module de traitement des Données à la Demande (voir la Figure 48), qui est à l'état "ISDU_1". Dans cet état, il force le module de traitement des ISDU à fournir le service OD en correspondance au service DL_ISDUTransport (voir la Figure 51, Transition T2). Le module de traitement des messages peut nécessiter plusieurs cycles avant que le module de traitement des ISDU ne retourne à l'état de "repos".
T16	6	7	Le module de traitement des messages invoque le service ODTrig pour le module de traitement des Données à la Demande (voir la Figure 48), qui est à l'état "Event_4". Dans cet état, il force le module de traitement d'Événements à fournir le service OD en correspondance au service EventFlag (voir la Figure 55, Transition T2). Le module de traitement des messages peut nécessiter plusieurs cycles avant que le module de traitement d'Événements ne retourne à l'état de "repos".
T17	6	7	Le module de traitement des messages invoque le service ODTrig pour le module de traitement des Données à la Demande (voir la Figure 48), qui est à l'état "ISDU_1". Dans cet état, il force le module de traitement des ISDU à fournir le service OD en correspondance au service DL_Write (voir la Figure 51, Transition T13).
T18	7	8	Envoi d'un message au bout d'un temps de récupération T_{initcyc} généré par le service OD.req. Démarrage du temporisateur avec le temps de séquence $T_{\text{M-sequence}}$.
T19	9	10	Réinitialisation du temporisateur ($T_{\text{M-sequence}}$).
T20	9	10	Réinitialisation du temporisateur ($T_{\text{M-sequence}}$).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T21	10	9	Retransmission du message au bout d'un temps $T_{initcyc}$. Redémarrage du temporisateur avec le temps de séquence $T_{M\text{-}sequence}$.
T22	8	0	Le module de traitement des messages passe à l'état Inactive_0 et renvoie MH_Info (COMLOST) au module de traitement de DL-mode.
T23	8	11	-
T24	11	7	Acquisition des OD par invocation du service ODTrig vers le module de traitement de Données à la Demande, qui à son tour déclenche le module de traitement qui en a actuellement la charge, par l'intermédiaire de l'invocation de ISDU ou d'EventTrig.
T25	11	6	Renvoi du résultat par l'intermédiaire de la primitive de service OD.cnf
T26	6	12	Le module de traitement des messages passe à l'état Operate_12.
T27	12	13	Démarrage du temporisateur t_{CYC}. Acquisition des PD par invocation du service PDTrig vers le module de traitement des Données de Processus (voir la Figure 46).
T28	13	14	Acquisition des OD par invocation du service ODTrig vers le module de traitement des Données à la Demande (voir la Figure 48).
T29	14	15	PD et OD prêtes par l'intermédiaire du service PD.req du module de traitement des PD et du service OD.req au moyen du module de traitement des OD. Le module de traitement des messages envoie le message. Démarrage du temporisateur avec le temps de séquence $T_{M\text{-}sequence}$.
T30	16	17	Réinitialisation du temporisateur ($T_{M\text{-}sequence}$).
T31	16	17	Réinitialisation du temporisateur ($T_{M\text{-}sequence}$).
T32	17	16	Retransmission du message au bout d'un temps t_{CYC}. Redémarrage du temporisateur avec le temps de séquence $T_{M\text{-}sequence}$.
T33	15	0	Le module de traitement des messages passe à l'état Inactive_0 et renvoie MH_Info (COMLOST) au module de traitement de DL-mode.
T34	15	12	Le message de réponse du Dispositif est correct. Renvoi de PD au moyen du service PD.cnf et par l'intermédiaire de l'invocation de PDTrig vers le module de traitement des PD (voir le Tableau 48). Renvoi des OD au moyen du service OD.cnf et par l'intermédiaire de l'invocation de ODTrig vers le module de traitement de Données à la Demande, qui les redirige vers le module de traitement des ISDU (voir le Tableau 53), le module de traitement des commandes (voir le Tableau 56), ou le module de traitement d'Événements (voir le Tableau 59) concerné.
T35	12	0	Le module de traitement des messages passe à l'état Inactive_0 et renvoie MH_Info (COMLOST) au module de traitement de DL-mode.
T36	6	0	Le module de traitement des messages passe à l'état Inactive_0 et renvoie MH_Info (COMLOST) au module de traitement de DL-mode.
T37	6	2	-
T38	12	2	-
T39	2	12	-
ÉLÉMENTS INTERNES	**TYPE**	**DÉFINITION**	
Retry (Nouvelle tentative)	Variable	Compteur du nombre de tentatives	
MaxRetry	Constant	MaxRetry = 2, voir le Tableau 102	
$t_{M\text{-}sequence}$	Temps	Voir l'équation (A.6)	
t_{CYC}	Temps	Le service DL_SetMode fournit cette valeur avec son paramètre "M-sequenceTime". Voir l'équation (A.7)	
$t_{initcyc}$	Temps	Voir A.2.6	
MH_Conf_xxx	Invocation	Voir le Tableau 44	

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.3.5 Diagramme d'états du module de traitement des messages du Dispositif

La Figure 44 représente le diagramme d'états du module de traitement des messages du Dispositif.

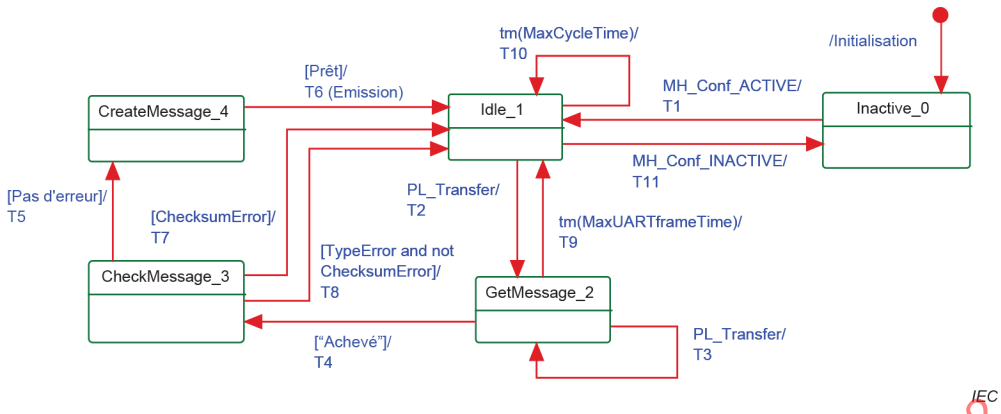


Figure 44 – Diagramme d'états du module de traitement des messages du Dispositif

Le Tableau 47 présente la table de transitions d'états du module de traitement des messages du Dispositif.

Tableau 47 – Table de transitions d'états du module de traitement des messages du Dispositif

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Inactive_0	Attente de l'activation par le module de traitement de mode DL du Dispositif par l'intermédiaire de MH_Conf_ACTIVE (voir le Tableau 45, Transition T1).
Idle_1	Attente de la première trame UART du message de Maître par l'intermédiaire de l'indication de service PL_Transfer. Vérification de l'expiration du temps "MaxCycleTime".
GetMessage_2	Réception d'une trame UART de message de Maître. Vérification du nombre de trames UART reçues (Le Dispositif détecte le type de séquence M au moyen des deux premiers octets reçus en fonction de l'état de la communication en cours, et il connaît donc le nombre de trames UART). Vérification de l'expiration du temps "MaxUARTframeTime".
CheckMessage_3	Vérification du type de séquence M et de la somme de contrôle du message reçu.
CreateMessage_4	Compilation du message à partir des services OD.rsp, PD.rsp, EventFlag et PDStatus.

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	-
T2	1	2	Démarrage de "MaxUARTframeTime" et "MaxCycleTime" en l'état OPERATE.
T3	2	2	Redémarrage du temporisateur "MaxUARTframeTime".
T4	2	3	Redémarrage du temporisateur "MaxUARTframeTime".
T5	3	4	Invocation des indications de service OD.ind et PD.ind
T6	4	1	Compilation et invocation de la réponse de service PL_Transfer.rsp (Le Dispositif envoie le message de réponse)
T7	3	1	-
T8	3	1	Indication d'erreur au module de traitement de mode DL par l'intermédiaire de MHInfo (ILLEGAL_MESSAGETYPE)
T9	2	1	Réinitialisation des deux temporisateurs "MaxUARTframeTime" et "MaxCycleTime".
T10	1	1	Indication d'erreur à la technologie d'actionneur qui doit observer cette information et prendre les mesures correspondantes (voir 10.2 et 10.8.3).
T11	1	0	Le module de traitement des messages du Dispositif passe à l'état Inactive_0.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
MaxUARTFrameTime	Temps	Temps de transmission d'une trame UART (11 T_{BIT}) plus la valeur maximale de t_1 (1 T_{BIT}) = 11 T_{BIT}.
MaxCycleTime	Temps	Le but du temporisateur "MaxCycleTime" est de vérifier si l'échange cyclique de Données de Processus a pris trop de temps ou s'il a été interrompu. Le MaxCycleTime doit être > au MasterCycleTime (voir A.3.7).
TypeError	Protection	Une des détections d'erreurs possibles: ILLEGAL_MESSAGETYPE, ou COMLOST
ChecksumError	Protection	Erreur de somme de contrôle de message détectée

7.3.4 Module de traitement des Données de Processus

7.3.4.1 Généralités

Le transport des Données de Processus de sortie est réalisé en utilisant les services DL_OutputUpdate et celui des Données de Processus d'entrée en utilisant les services DL_InputTransport (voir la Figure 28). Un cycle de Données de Processus est terminé lorsque l'ensemble des Données de Processus a été transféré entre le Maître et le Dispositif dans le sens demandé. Ce cycle peut durer pendant plusieurs séquences M.

Toutes les Données de Processus sont transmises dans une séquence M lorsque des séquences M de TYPE_2_x (voir la Figure 39) sont utilisées. Dans ce cas, la durée d'exécution d'un cycle de Données de Processus est égale à la durée de cycle t_{CYC}.

7.3.4.2 Mode d'entrelacement

Toutes les Données de Processus et les Données à la Demande sont transmises dans ce cas en utilisant plusieurs séquences M alternantes de TYPE_1_1 (Données de Processus) et de TYPE_1_2 (Données à la Demande), comme cela est représenté à la Figure 45. Cette figure représente l'écriture de messages du Maître contenant des Données de Processus de sortie vers un Dispositif. Le paramètre de service PDOutAddress indique le partitionnement des PD de sortie à transmettre (voir 7.2.2.3). Pour les Données de Processus d'entrée, le paramètre de service PDInAddress indique quant à lui le partitionnement des PD d'entrée. Dans un cycle de Données de Processus, toutes les PD d'entrée doivent être lues en premier, puis suivies par toutes les PD de sortie à écrire. Un cycle de Données de Processus comprend toutes les durées de cycle nécessaires à la transmission de l'ensemble des Données de Processus.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

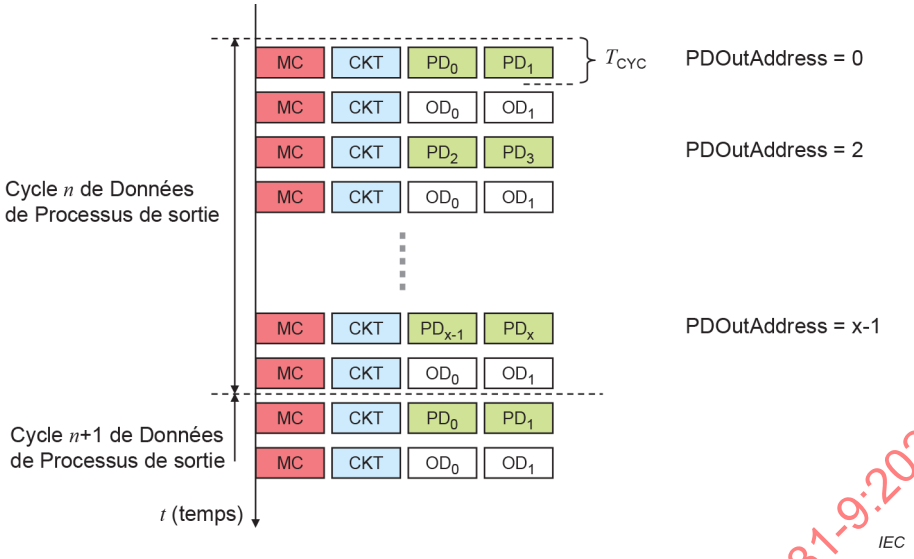


Figure 45 – Mode d'entrelacement pour la transmission segmentée des Données de Processus

Le mode d'entrelacement est destiné aux seuls Dispositifs hérités.

7.3.4.3 Diagramme d'états du module de traitement des Données de Processus du Maître

La Figure 46 représente le diagramme d'états du module de traitement des Données de Processus du Maître.

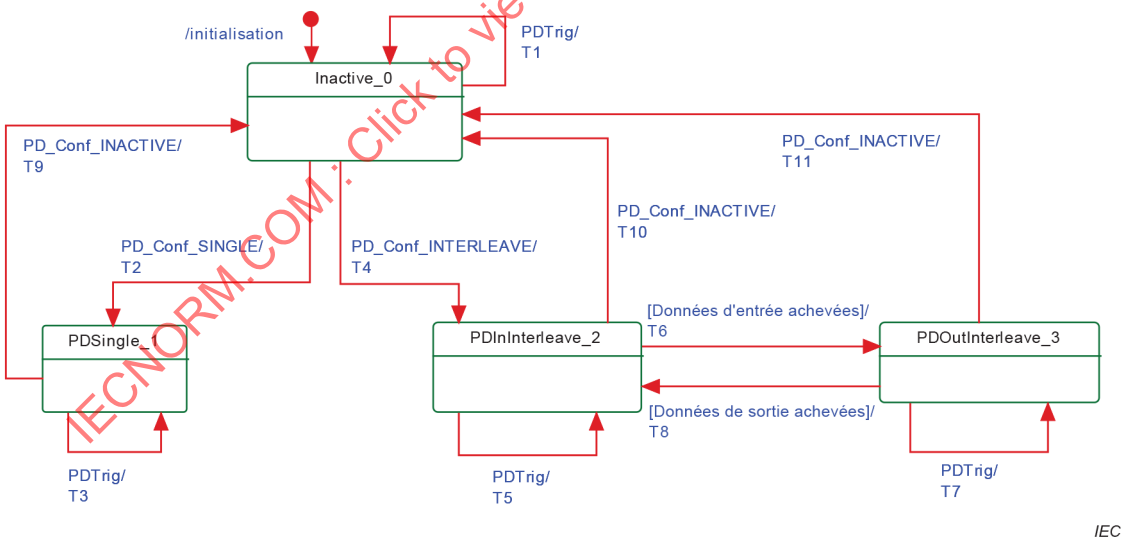


Figure 46 – Diagramme d'états du module de traitement des Données de Processus du Maître

Le Tableau 48 présente la table de transitions d'états du module de traitement des Données de Processus du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 48 – Table de transitions d'états du module de traitement des Données de Processus du Maître

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation	
PDSingle_1		Communication des Données de Processus en une séquence M unique	
PDInInterleave_2		Communication des Données de Processus d'entrée en mode d'entrelacement	
PDOutInterleave_3		Communication des Données de Processus de sortie en mode d'entrelacement	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	0	Invocation de PD.req sans Données de Processus
T2	0	1	NOTE Le module de traitement de mode DL a configuré le module de traitement des Données de Processus pour une transmission de PD en une seule fois (voir le Tableau 44, T10 ou T11)
T3	1	1	Récupération de données auprès du service DL_PDOutputUpdate et invocation de PD.req pour propager les PD résultantes au module de traitement des messages. Récupération de données à partir de PD.cnf et invocation de DL_PDInputTransport.ind et DL_PDCycle.ind pour propager les PD d'entrée vers la couche AL.
T4	0	2	NOTE Configuré pour transmission des PD en entrelacement (voir le Tableau 44, T10 ou T11).
T5	2	2	Invocation de PD.req et utilisation de PD.cnf pour préparer DL_PDInputTransport.ind.
T6	2	3	Invocation de DL_PDInputTransport.ind et de DL_PDCycle.ind pour propager les PD d'entrée vers la couche AL (voir 7.2.1.11).
T7	3	3	Récupération de données auprès du service DL_PDOutputUpdate et invocation de PD.req pour propager les PD de sortie au module de traitement des messages.
T8	3	2	Invocation de DL_PDCycle.ind pour indiquer la fin du cycle de Données de Processus à la couche AL (voir 7.2.1.12).
T9	1	0	-
T10	2	0	-
T11	3	0	-
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
<Aucun>			

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.4.4 Diagramme d'états du module de traitement des Données de Processus du Dispositif

La Figure 47 représente le diagramme d'états du module de traitement des Données de Processus du Dispositif.

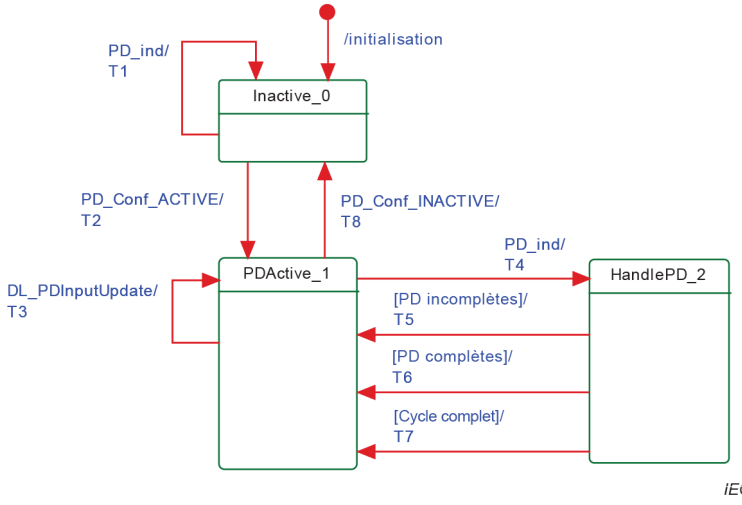


Figure 47 – Diagramme d'états du module de traitement des Données de Processus du Dispositif

Dans ce contexte, se reporter aux diagrammes de séquences à la Figure 67 et à la Figure 68.

Le Tableau 49 présente la table de transitions d'états du module de traitement des Données de Processus du Dispositif.

Tableau 49 – Table de transitions d'états du module de traitement des Données de Processus du Dispositif

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation	
PDActive_1		Module de traitement actif et en attente de la demande suivante du module de traitement des messages au moyen du service PD ou le service DL_PDInputUpdate en provenance de la couche AL.	
HandlePD_2		Vérification de l'exhaustivité des Données de Processus en mode d'entrelacement	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	0	Les Données de Processus sont ignorées
T2	0	1	-
T3	1	1	Préparation des Données de Processus d'entrée pour PD.rsp à la demande suivante du module de traitement des messages
T4	1	2	Le module de traitement des messages demande des PD d'entrée au moyen d'un service PD.ind et remet des PD de sortie ou des segments de PD de sortie. Invocation de PD.rsp avec des Données de Processus d'entrée en mode sans entrelacement (voir 7.2.2.3).
T5	2	1	-
T6	2	1	Invocation de DL_PDOutputTransport.ind (voir 7.2.1.9)
T7	2	1	Invocation de DL_PDCycle.ind (voir 7.2.1.12)
T8	1	0	-
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
PD_ind		Étiquette	Le service PD.ind a été invoqué par le module de traitement des messages

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.5 Module de traitement de Données à la Demande

7.3.5.1 Généralités

Le module de traitement de Données à la Demande du Maître est un diagramme d'états subordonné actif à l'état "Startup_2", "PreOperate_3" et "Operate_4" du module de traitement de mode DL (voir la Figure 35). Il contrôle trois autres diagrammes d'états appelés module de traitement des ISDU, module de traitement des commandes et module de traitement d'Événements. Par défaut, il commence toujours en module de traitement des ISDU.

Dès qu'un EventFlag.ind est reçu, le diagramme d'états passe en module de traitement d'Événements. Après avoir lu entièrement les informations d'Événement, il retourne à l'état de module de traitement des ISDU.

Dès qu'un service DL_Control.req ou PDInStatus.ind est reçu lorsqu'il est en module de traitement des ISDU ou en module de traitement d'Événements, le diagramme d'états passe en module de traitement des commandes. Dès lors que la commande est servie, le diagramme d'états retourne à l'état précédemment actif (ISDU ou Événement).

7.3.5.2 Diagramme d'états du module de traitement de Données à la Demande du Maître

La Figure 48 représente le diagramme d'états du module de traitement de Données à la Demande du Maître.

Le module de traitement de Données à la Demande redirige la primitive de service ODTrig.ind pour le contenu du message suivant vers le module de traitement subsidiaire couramment actif (ISDU, commandes ou Événements). Ceci est réalisé par l'intermédiaire d'une des invocations ISDUTrig, CommandTrig ou EventTrig.

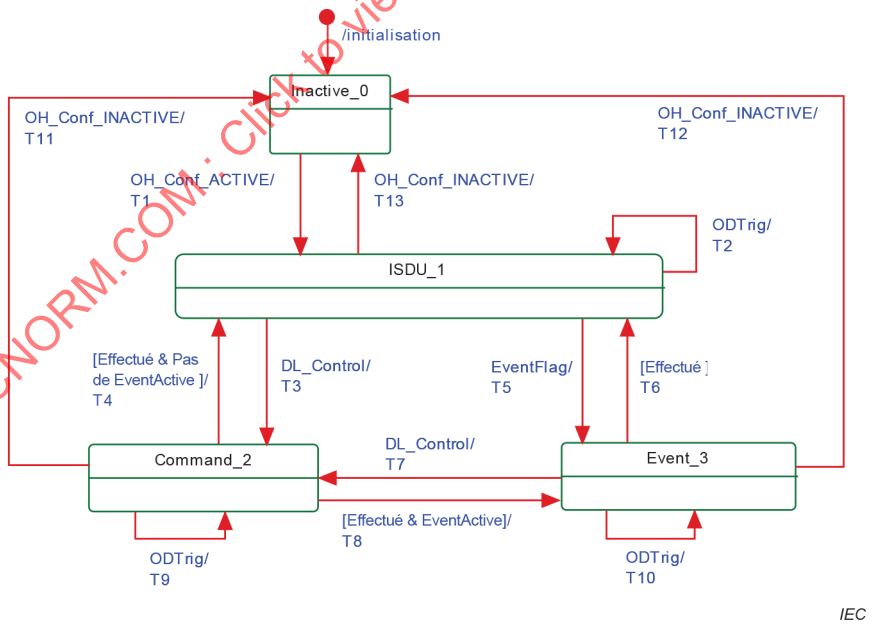


Figure 48 – Diagramme d'états du module de traitement de Données à la Demande du Maître

Le Tableau 50 présente la table de transitions d'états du module de traitement des Données à la Demande du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 50 – Table de transitions d'états du module de traitement des Données à la Demande du Maître

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation	
ISDU_1		État par défaut du module de traitement de Données à la Demande (non prioritaire)	
Command_2		État pour le contrôle du Dispositif par l'intermédiaire de commandes prioritaires	
Event_3		État pour l'acheminement des informations d'Événements (erreurs, avertissements, notifications) prioritaires	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	1	–
T2	1	1	Le module de traitement de Données à la Demande propage le service ODTrig.ind à présent désigné ISDUTrig vers le module de traitement des ISDU (voir la Figure 51). Dans le cas des services DL_Read, DL_Write, DL_ReadParam ou DL_WriteParam, le module de traitement des ISDU utilise une transition séparée (voir la Figure 51, T13).
T3	1	2	–
T4	2	1	–
T5	1	3	EventActive = VRAI
T6	3	1	EventActive = FAUX
T7	3	2	–
T8	2	3	–
T9	2	2	Le module de traitement de Données à la Demande propage le service ODTrig.ind à présent désigné CommandTrig vers le module de traitement des commandes (voir la Figure 53)
T10	3	3	Le module de traitement de Données à la Demande propage le service ODTrig.ind à présent désigné EventTrig vers le module de traitement d'Événements (voir la Figure 55)
T11	2	0	–
T12	3	0	–
T13	1	0	–
T14	1	2	–
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
EventActive		Bool	Drapeau indiquant le sens de reprise après une interruption du traitement d'Événements pour une demande de commande prioritaire

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.5.3 Diagramme d'états du module de traitement de Données à la Demande du Dispositif

La Figure 49 représente le diagramme d'états du module de traitement de Données à la Demande du Dispositif.

Le module de traitement de Données à la Demande du Dispositif obtient des informations sur le canal de communication et le paramètre ou l'adresse de FlowCTRL au moyen du service OD.ind. Les canaux de communication sont totalement indépendants. Si l'Accès est valide, le diagramme d'états correspondant (des ISDU, des commandes ou des Événements) est sollicité au moyen du canal de communication associé.

Le Dispositif doit répondre aux demandes de lecture vers des plages d'adresses non utilisées de valeur "0". Il doit ignorer les demandes d'écriture vers des plages d'adresses non utilisées.

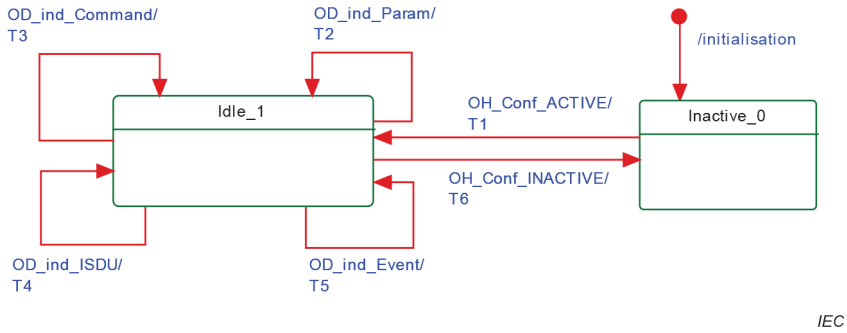


Figure 49 – Diagramme d'états du module de traitement de Données à la Demande du Dispositif

Dans le cas d'un Accès ISDU dans un Dispositif sans prise en charge des ISDU, le Dispositif doit répondre par "Pas de Service" (voir le Tableau A.12). Il n'est pas généré de message d'erreur.

NOTE OD.ind (R, ISDU, FlowCTRL = IDLE) est le message par défaut s'il n'y a pas de Données à la Demande en attente de transmission.

Le Tableau 51 présente la table de transitions d'états du module de traitement des Données à la Demande du Dispositif.

Tableau 51 – Table de transitions d'états du module de traitement des Données à la Demande du Dispositif

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation	
Idle_1		Attente de messages avec des Données à la Demande au moyen de l'indication de service OD. Décomposition et analyse.	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	1	–
T2	1	1	Fournit le contenu des données du paramètre demandé ou procède à une action d'écriture appropriée.
T3	1	1	Redirection vers le module de traitement des commandes
T4	1	1	Redirection vers le module de traitement des ISDU
T5	1	1	Redirection vers le module de traitement d'Événements
T6	1	0	–
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
OD_ind_Param		Service	Alias du service OD.ind (Lecture/Écriture, PAGE, 1 à 31, Données) dans le cas de DL_ReadParam ou DL_WriteParam
OD_ind_Command		Service	Alias du service OD.ind (Écriture, PAGE, 0, MasterCommand)
OD_ind_ISDU		Service	Alias du service OD.ind (Lecture/Écriture, ISDU, FlowCtrl, Données)
OD_ind_Event		Service	Alias du service OD.ind (Lecture/Écriture, DIAGNOSTIC, n, Données)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.6 Module de traitement des ISDU

7.3.6.1 Unité de données de service indexée (ISDU)

La structure générale d'une ISDU est représentée à la Figure 50 et décrite de manière détaillée à l'Article A.5.

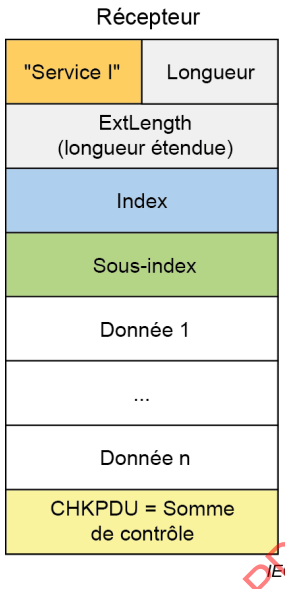


Figure 50 – Structure de l'ISDU

La séquence des éléments correspond à la séquence de transmission. Les éléments d'une ISDU peuvent prendre différentes formes en fonction du type de "Service I" (voir A.5.2 et le Tableau A.12).

L'ISDU permet d'accéder à des objets de données (paramètres et commandes) à transmettre (voir la Figure 6). Les objets de données doivent être adressés par l'élément "Index".

Tous les types de données à octets multiples doivent être transmis sous forme de séquence "gros-boutiste", c'est-à-dire que l'octet de poids fort (MSO) doit être envoyé en premier, suivie par des octets de poids faible en ordre décroissant, l'octet de poids faible (LSO) étant envoyé en dernier, comme cela est représenté à la Figure 2.

7.3.6.2 Transmission des ISDU

Une ISDU est transmise au moyen du canal de communication des ISDU (voir la Figure 8 et A.1.2). Un certain nombre de messages est généralement exigé pour effectuer cette transmission (segmentation). Le Maître transfère une ISDU en envoyant une demande de "Service I" (Lecture/Écriture) au Dispositif au moyen du canal de communication des ISDU. Il reçoit ensuite la réponse du Dispositif par le même canal.

Sur le canal de communication des ISDU, l'élément "Adresse" dans l'octet de contrôle de séquence M inclut un compteur (= FlowCTRL). FlowCTRL contrôle le flux de données segmentées (voir A.1.2) en comptant les séquences M nécessaires à la transmission d'une ISDU.

Le récepteur d'une ISDU attend un FlowCTRL + 1 dans le message de texte en cas de communication non perturbée. Si FlowCTRL ne change pas, le message déjà transmis est répété. Dans tous les autres cas, la structure ISDU n'est pas respectée.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Le Maître utilise l'élément "Longueur" de l'ISDU et FlowCTRL pour vérifier l'accomplissement de la transmission complète.

Les valeurs admissibles de FlowCTRL sont spécifiées dans le Tableau 52.

Tableau 52 – Définitions du FlowCTRL

FlowCTRL	Définition
0x00 à 0x0F	COUNT Compteur de séquences M dans une ISDU. S'incrémente à partir de 1 après un DÉBUT D'ISDU. Revient de 15 à 0 en cas d'Événement de débordement.
0x10	START Début d'un "Service I" de ISDU, c'est-à-dire le début d'une demande ou d'une réponse. Pour le début d'une demande, tous les services précédemment incomplets peuvent être rejetés. Pour une demande de démarrage associée à une réponse, un Dispositif doit envoyer "Pas de Service" jusqu'à ce que son application renvoie des données de réponse (voir le Tableau A.12).
0x11	IDLE 1 Pas de demande de transmission de ISDU.
0x12	IDLE 2: Réservé pour usage ultérieur Pas de demande de transmission de ISDU.
0x13 à 0x1E	Réservé
0x1F	ABORT Abandon de l'ensemble du service. Le Maître répond en rejetant les données de réponse reçues. Le Dispositif répond en rejetant les données de demande reçues et peut générer un abandon.

Dans l'état Idle_1, les valeurs 0x12 à 0x1F ne doivent pas donner lieu à une erreur de communication.

7.3.6.3 Diagramme d'états du module de traitement des ISDU du Maître

La Figure 51 représente le diagramme d'états du module de traitement des ISDU du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

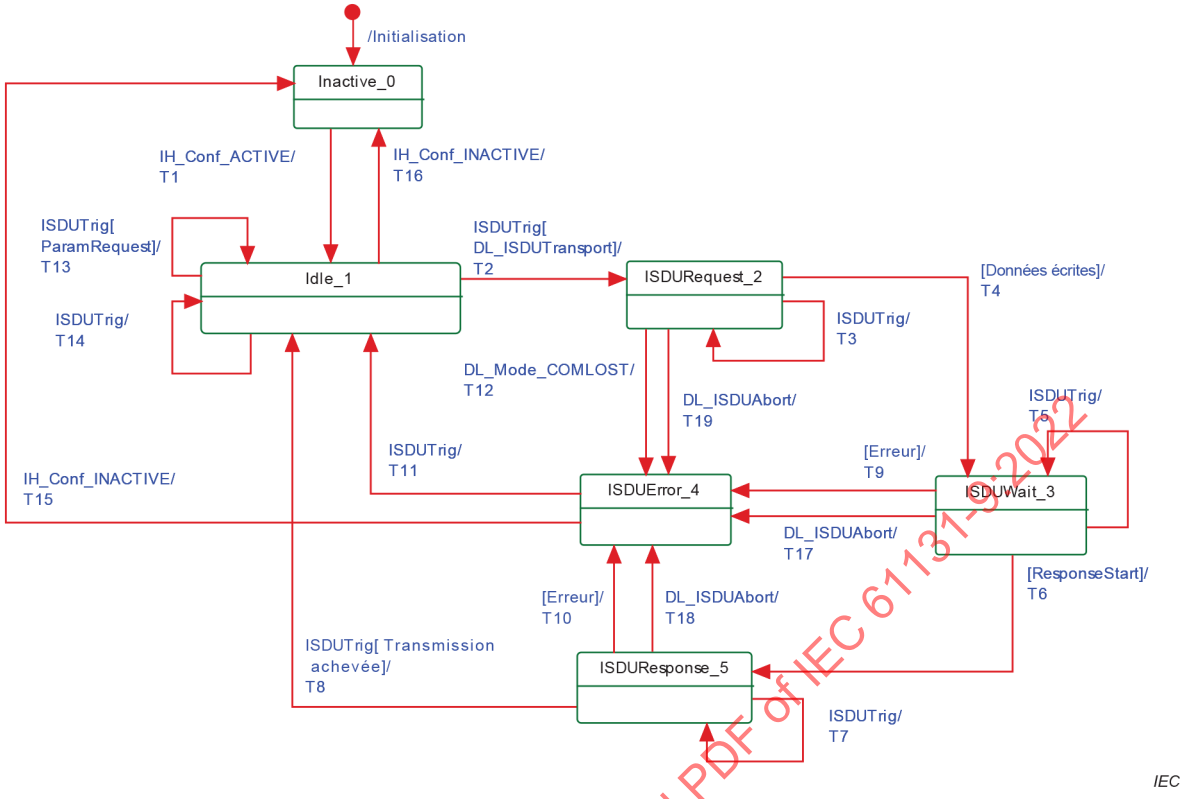


Figure 51 – Diagramme d'états du module de traitement des ISDU du Maître

Le Tableau 53 présente la table de transitions d'états du module de traitement des ISDU du Maître.

Tableau 53 – Table de transitions d'états du module de traitement des ISDU du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Inactive_0	Attente d'activation
Idle_1	Attente de transmission des prochaines Données à la Demande
ISDURequest_2	Transmission des données de demande d'ISDU
ISDUWait_3	Attente de réponse du Dispositif. Respect de ISDUTime
ISDUError_4	Gestion des erreurs après détection: Invocation de la réponse DL_ISDU_Transport négative avec ISDUTransportErrorInfo
ISDUResponse_5	Obtention des données de réponse du Dispositif

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	–
T2	1	2	Invocation de OD.req avec condition de début d'écriture des ISDU: OD.req (Écriture, ISDU, flowCtrl = DÉBUT, données)
T3	2	2	Invocation de OD.req avec écriture de données ISDU et FlowCTRL dans les conditions définies dans le Tableau 52
T4	2	3	Démarrage du temporisateur (ISDUTime)
T5	3	3	Invocation de OD.req avec condition de début de lecture des ISDU: OD.req (Lecture, ISDU, flowCtrl = DÉBUT)
T6	3	5	Arrêt du temporisateur (ISDUTime)
T7	5	5	Invocation de OD.req avec lecture de données ISDU et FlowCTRL dans les conditions définies dans le Tableau 52

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T8	5	1	OD.req (R, ISDU, flowCtrl = IDLE) Invocation de confirmation positive de DL_ ISDUTransport
T9	3	4	–
T10	5	4	–
T11	4	1	Invocation de OD.req avec abandon des ISDU: OD.req (Lecture, ISDU, flowCtrl = ABANDON) Invocation de confirmation négative de DL_ ISDUTransport
T12	2	4	–
T13	1	1	Invocation de OD.req avec des données appropriées Invocation de confirmation positive de DL_ReadParam/DL_WriteParam
T14	1	1	Invocation de OD.req avec message au repos: OD.req (Lecture, ISDU, flowCtrl = REPOS)
T15	4	1	En cas de perte de communication, le module de traitement des messages informe le module de traitement de Mode DL qui à son tour utilise l'invocation réglementaire IH_Conf_INACTIVE. Aucune action n'est exigée au cours de cette transition.
T16	1	0	–
T17	3	4	–
T18	5	4	–
T19	2	4	–

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
ISDUTime	Temps	Mesurage du temps de réponse du Dispositif (chien de garde, voir le Tableau 102)
ResponseStart	Service	OD.cnf sans l'indication "occupé" (voir le Tableau A.14)
ParamRequest	Service	DL_ReadParam ou DL_WriteParam
Erreur	Variable	Toute erreur détectable dans la transmission de ISDU ou demandes DL_ISDUAbort, ou toute violation du délai d'acquittement des ISDU (voir le Tableau 102)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.6.4 Diagramme d'états du module de traitement des ISDU du Dispositif

La Figure 52 représente le diagramme d'états du module de traitement des ISDU du Dispositif.

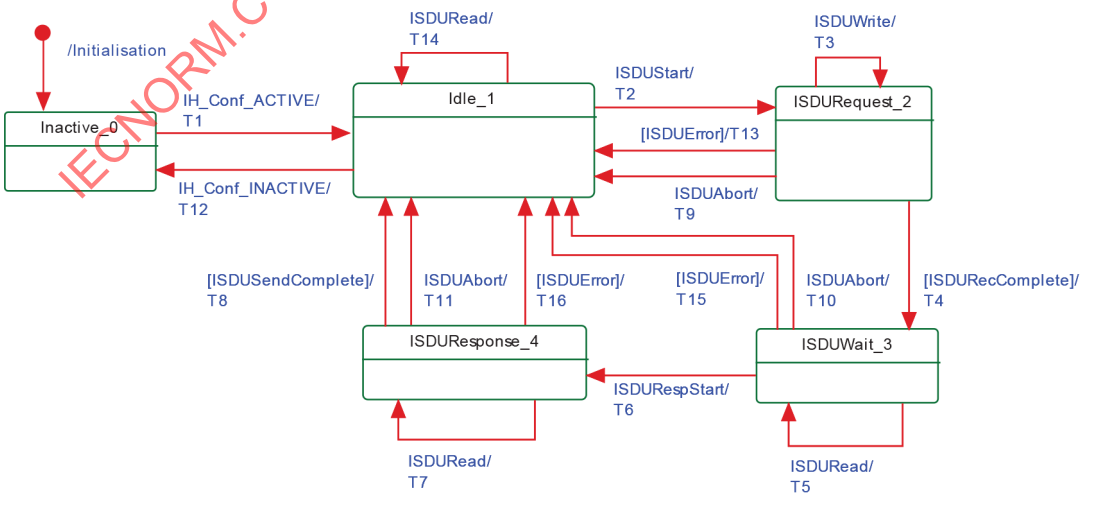


Figure 52 – Diagramme d'états du module de traitement des ISDU du Dispositif

Le Tableau 54 présente la table de transitions d'états du module de traitement des ISDU du Dispositif.

Tableau 54 – Table de transitions d'états du module de traitement des ISDU du Dispositif

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation	
Idle_1		Attente de transmission des ISDU suivantes	
ISDURequest_2		Réception de demande d'ISDU	
ISDUWait_3		Attente de données à transmettre en provenance de la couche d'application (voir DL_ISDUTransport)	
ISDUResponse_4		Transmission des données de réponse d'ISDU	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	1	–
T2	1	2	Début de réception des données de demande d'ISDU
T3	2	2	Réception des données de demande d'ISDU
T4	2	3	Invocation de DL_ ISDUTransport.ind vers la couche AL (voir 7.2.1.6)
T5	3	3	Invocation de OD.rsp avec l'indication "occupé" (voir le Tableau A.14)
T6	3	4	–
T7	4	4	Invocation de OD.rsp avec données de réponse d'ISDU
T8	4	1	–
T9	2	1	–
T10	3	1	Invocation de DL_ ISDUAbort
T11	4	1	Invocation de DL_ ISDUAbort
T12	1	0	–
T13	2	1	Invocation de DL_ISDUAbort
T14	1	1	Invocation de OD.rsp avec l'indication "Pas de Service" (voir le Tableau A.12 et le Tableau A.14)
T15	3	1	Invocation de DL_ISDUAbort
T16	4	1	Invocation de DL_ISDUAbort
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
ISDUStart		Service	OD.ind(Écriture, ISDU, Début, Données)
ISDUWrite		Service	OD.ind(Écriture, ISDU, FlowCtrl, Données)
ISDURecComplete		Protection	Si OD.ind(Lecture, ISDU, Début,…) reçue
ISDURespStart		Service	DL_ ISDUTransport.rsp()
ISDURead		Service	OD.ind(Lecture, ISDU, Début ou FlowCtrl,…)
ISDUSendComplete		Protection	Si OD.ind(Lecture, ISDU, REPOS,…) reçue
ISDUAbort		Service	OD.ind(Lecture/Écriture, ISDU, Abandon,…)
ISDUError		Protection	Si la structure des ISDU est incorrecte ou qu'une erreur FlowCTRL est détectée

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.7 Module de traitement des commandes

7.3.7.1 Généralités

Le module de traitement des commandes passe le code de contrôle (PDOUTVALID ou PDOUTINVALID) contenu dans la primitive de service DL_Control.req au module de traitement des messages à fonctionnement cyclique, au moyen du service OD.req et MasterCommands. Le module de traitement des messages utilise le canal de communication de page.

Les codes de contrôle admissibles pour les Données de Processus de sortie sont énumérés dans le Tableau 55.

Tableau 55 – Codes de contrôle

Code de contrôle	MasterCommand	Description
PDOUTVALID	ProcessDataOutputOperate	Données de Processus de sortie valides
PDOUTINVALID	DeviceOperate	Données de Processus de sortie non valides ou manquantes

Le module de traitement des commandes reçoit des informations d'état des Données de Processus d'entrée, au moyen du service PDInStatus et les propage dans une primitive de service DL_Control.ind.

En outre, le module de traitement des commandes traduit les demandes de modification de mode du Dispositif provenant de la Gestion Système vers des commandes MasterCommands correspondantes (voir le Tableau B.2).

7.3.7.2 Diagramme d'états du module de traitement des commandes du Maître

La Figure 53 représente le diagramme d'états du module de traitement des commandes du Maître.

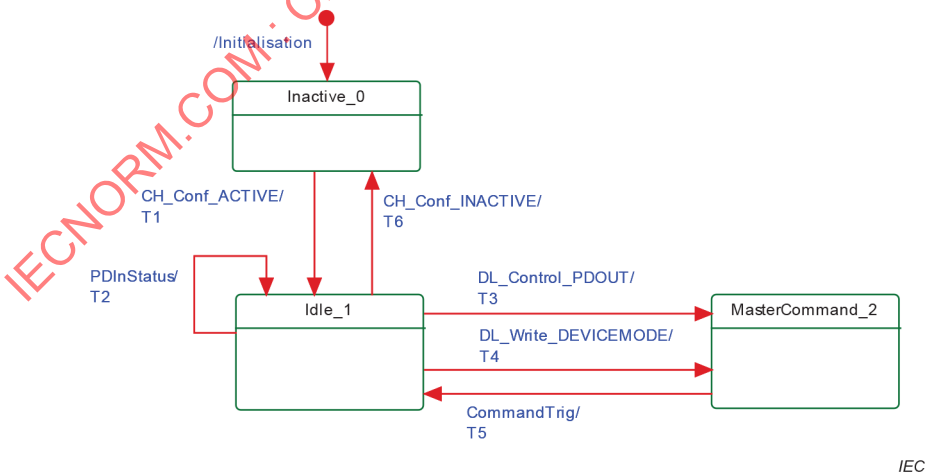


Figure 53 – Diagramme d'états du module de traitement des commandes du Maître

Le Tableau 56 présente la table de transitions d'états du module de traitement des commandes du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 56 – Table de transitions d'états du module de traitement des commandes du Maître

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation par le module de traitement de mode DL	
Idle_1		Attente de nouvelle commande en provenance de la couche AL: DL_Control (état des PD de sortie) ou en provenance de la SM: DL_Write (changement de mode du Dispositif, par exemple en OPERATE), ou attente de la primitive de service PDInStatus.ind.	
MasterCommand_2		Préparation de données pour la primitive de service OD.req. Attente de demande de la part du module de traitement des OD (CommandTrig).	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	1	-
T2	1	1	Si le service PDInStatus.ind = VALID, invocation de DL_Control.ind (VALID) pour signaler des Données de Processus d'entrée valides vers AL. Si le service PDInStatus.ind = INVALID, invocation de DL_Control.ind (INVALID) pour signaler des Données de Processus d'entrée non valides vers AL.
T3	1	1	Si le service DL_Control.req = PDOUTVALID, invocation de OD.req (ÉCRITURE, PAGE, 0, 1, MasterCommand = 0x98). Si le service DL_Control.req = PDOUTINVALID, invocation de OD.req (ÉCRITURE, PAGE, 0, 1, MasterCommand = 0x99). Voir le Tableau B.2.
T4	1	2	Les services DL_Write DEVICEMODE traduisent en: INACTIVE: OD.req (ÉCRITURE, PAGE, 0, 1, MasterCommand = 0x5A) STARTUP: OD.req (ÉCRITURE, PAGE, 0, 1, MasterCommand = 0x97) PREOPERATE: OD.req (ÉCRITURE, PAGE, 0, 1, MasterCommand = 0x9A) OPERATE: OD.req (ÉCRITURE, PAGE, 0, 1, MasterCommand = 0x99)
T5	2	1	Une invocation CommandTrig à partir du module de traitement des OD force le module de traitement des commandes à invoquer la primitive de service OD.req et par conséquent le module de traitement des messages à envoyer au Dispositif la MasterCommand appropriée.
T6	1	0	-
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
DEVICEMODE		Étiquette	N'importe quel mode du Dispositif: INACTIVE, STARTUP, PREOPERATE, ou OPERATE
PDout		Étiquette	Un des deux codes de contrôle de sortie: PDOUTVALID ou PDOUTINVALID (voir le Tableau 55)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.7.3 Diagramme d'états du module de traitement des commandes du Dispositif

La Figure 54 représente le diagramme d'états du module de traitement des commandes du Dispositif. Il est principalement piloté par des MasterCommands en provenance du module de traitement des commandes du Maître pour contrôler les modes du Dispositif et l'état des Données de Processus de sortie. Il contrôle également l'état des Données de Processus d'entrée au moyen du service PDInStatus.

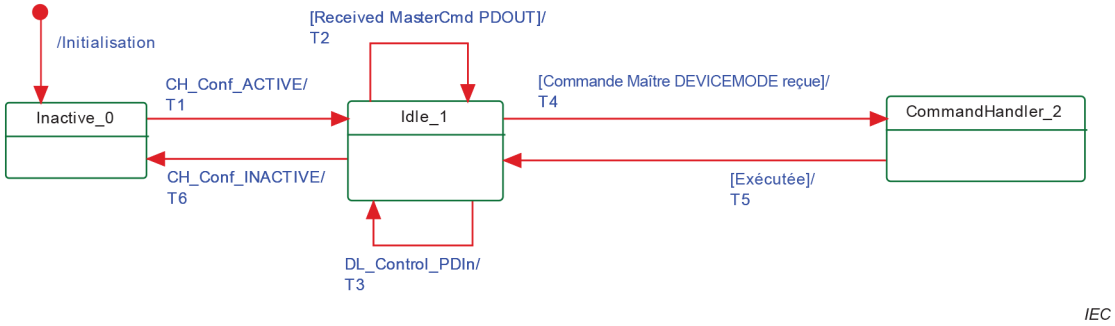


Figure 54 – Diagramme d'états du module de traitement des commandes du Dispositif

Le Tableau 57 présente la table de transitions d'états du module de traitement des commandes du Dispositif.

Tableau 57 – Table de transitions d'états du module de traitement des commandes du Dispositif

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Inactive_0		Attente d'activation	
Idle_1		Attente de la MasterCommand suivante	
CommandHandler_2		Décomposition de la MasterCommand et invocation d'actions spécifiques (voir B.1.2): Si MasterCommand = 0x5A, passage du Dispositif à l'état INACTIVE. Si MasterCommand = 0x97, passage du Dispositif à l'état STARTUP. Si MasterCommand = 0x9A, passage du Dispositif à l'état PREOPERATE. Si MasterCommand = 0x99, passage du Dispositif à l'état OPERATE.	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	1	
T2	1	1	Invocation de DL_Control.ind (PDOUTVALID) si MasterCommand reçue = 0x98. Invocation de DL_Control.ind (PDOUTINVALID) si MasterCommand reçue = 0x99.
T3	1	1	Si service DL_Control.req (VALID), invocation de PDInStatus.req (VALID). Si service DL_Control.req (INVALID), invocation de PDInStatus.req (INVALID). Le module de traitement des messages utilise le service PDInStatus pour initialiser/réinitialiser le drapeau d'état des PD (voir A.1.5)
T4	1	2	–
T5	2	1	–
T6	1	0	–
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
<aucun>			

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.8 Module de traitement d'Événements

7.3.8.1 Événements

Il existe deux types d'Événements, les uns détaillés et les autres non détaillés. Des événements non détaillés peuvent avoir été utilisés dans des Dispositifs hérités, mais ils ne doivent pas être utilisés pour des Dispositifs conformes au présent document. Cependant, tous les Maîtres doivent prendre en charge le traitement des deux types d'Événements, détaillés et non détaillés.

La structure générale et le codage des Événements sont définis à l'Article A.6. Les codes d'événements non détaillés sont donnés dans le Tableau A.16. Les codes d'événements détaillés sont spécifiés à l'Annexe D. La structure de la mémoire d'Événements pour des codes d'événements détaillés dans un Dispositif donné est définie dans le Tableau 58.

Tableau 58 – Mémoire d'Événements

Adresse	Numéro d'emplacement de l'événement	Nom du paramètre	Description
0x00		StatusCode	Récapitulatif des informations d'état et d'erreurs. Également utilisé pour contrôler l'Accès en lecture pour des messages individuels.
0x01	1	EventQualifier 1	Type, mode et origine de l'Événement
0x02		EventCode 1	Code d'Événement sur 16 bits de l'Événement
0x03			
0x04	2	EventQualifier 2	Type, mode et origine de l'Événement
0x05		EventCode 2	Code d'Événement sur 16 bits de l'Événement
0x06			
...			
0x10	6	EventQualifier 6	Type, mode et origine de l'Événement
0x11		EventCode 6	Code d'Événement sur 16 bits de l'Événement
0x12			
0x13 à 0x1F			Réservé pour un usage futur

7.3.8.2 Traitement des Événements

La couche AL du Dispositif écrit un Événement dans la mémoire d'Événements, puis établit le bit "drapeau d'Événement" qui est ensuite envoyé au Maître avec le message suivant dans l'octet CKS (voir 7.3.3.2 et A.1.5).

À la réception d'un message de réponse du Dispositif avec le bit "drapeau d'Événement" = 1, le Maître doit faire passer le module de traitement des ISDU en module de traitement d'Événements. Le module de traitement d'Événements commence à lire le StatusCode.

Si le bit "Détails de l'Événement" est établi (voir la Figure A.22), le Maître doit lire les détails des Événements indiqués dans le StatusCode à partir de la mémoire d'Événements. Après avoir lu les détails d'un Événement, il doit invoquer le service DL_Event.ind. Après réception du service DL_EventConf, le Maître doit écrire dans le StatusCode les éventuelles données nécessaires pour réinitialiser le bit "drapeau d'Événement". La gestion des événements du côté Maître doit être menée à terme quel que soit le contenu des données d'Événements reçues (EventQualifier, EventCode).

Si le bit "Détails d'Événement" n'est pas établi (voir la Figure A.21), le module de traitement d'Événements du Maître doit générer les Événements normalisés conformément au Tableau A.16 commençant par le bit de poids fort dans le EventCode.

L'Accès en écriture au StatusCode indique au Dispositif la fin du traitement des Événements. Le Dispositif doit ignorer les données de cet Accès en écriture du Maître. Le Dispositif réinitialise alors le bit "Drapeau d'Événement" et peut désormais modifier le contenu des champs de la mémoire d'Événements.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

7.3.8.3 Diagramme d'états du module de traitement d'Événements du Maître

La Figure 55 représente le diagramme d'états du module de traitement d'Événements du Maître.

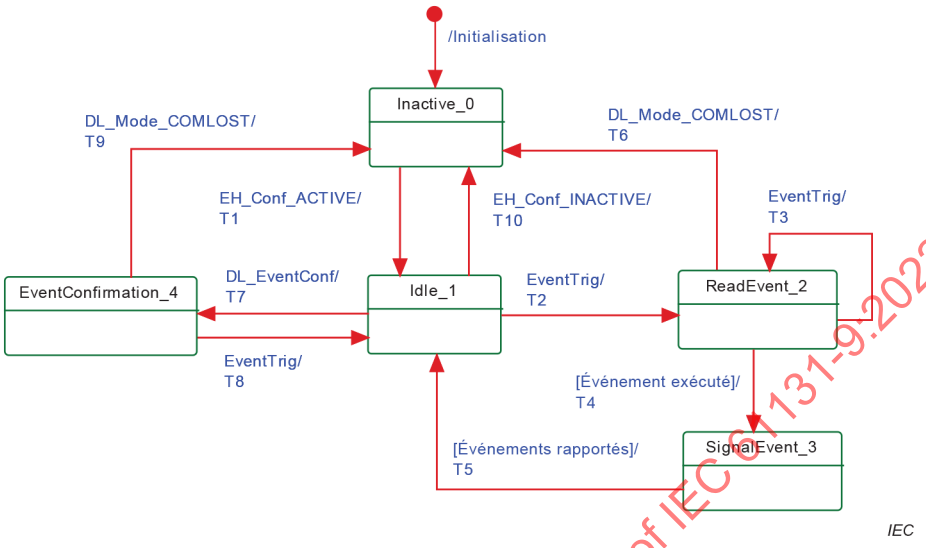


Figure 55 – Diagramme d'états du module de traitement d'Événements du Maître

Le Tableau 59 présente la table de transitions d'états du module de traitement d'Événements du Maître.

Tableau 59 – Table de transitions d'états du module de traitement d'Événements du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Inactive_0	Attente d'activation
Idle_1	Attente de l'indication d'Événement suivante ("EventTrig" par l'intermédiaire du module de traitement de Données à la Demande) ou confirmation d'Événement par l'intermédiaire du service DL_EventConf en provenance de la couche AL du Maître.
ReadEvent_2	Lecture, message par message, du jeu de données d'Événement du Dispositif par l'intermédiaire de l'adresse de la mémoire d'Événements. Vérification du Code d'État pour un certain nombre d'Événements activés (voir le Tableau 58).
SignalEvent_3	Analyse des données d'Événement et invocation de l'indication DL_Event vers la couche AL du Maître (voir 7.2.1.15) pour chaque Événement disponible.
EventConfirmation_4	Attente de la transmission de la confirmation d'Événement au moyen du service OD.req vers le Dispositif

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	–
T2	1	2	Lecture de l'octet de Code d'État d'Événement au moyen du service OD.req (Lecture, DIAGNOSTIC, adresse de mémoire d'Événements = 0, 1)
T3	2	2	Lecture des octets en provenance de la mémoire d'Événements au moyen du service OD.req (Lecture, DIAGNOSTIC, adresse de mémoire d'Événements incrémentée, 1)
T4	2	3	–
T5	3	1	–
T6	2	0	–

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T7	1	4	–
T8	4	1	Invocation de OD.req (Écriture, DIAGNOSTIC, 0, 1, toutes données) avec Accès en écriture au "Code d'État" (voir le Tableau 58) pour confirmation de la lecture de l'Événement au Dispositif
T9	4	0	–
T10	1	0	–
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
<Aucun>			

7.3.8.4 Diagramme d'états du module de traitement d'Événements du Dispositif

La Figure 56 représente le diagramme d'états du module de traitement d'Événements du Dispositif.

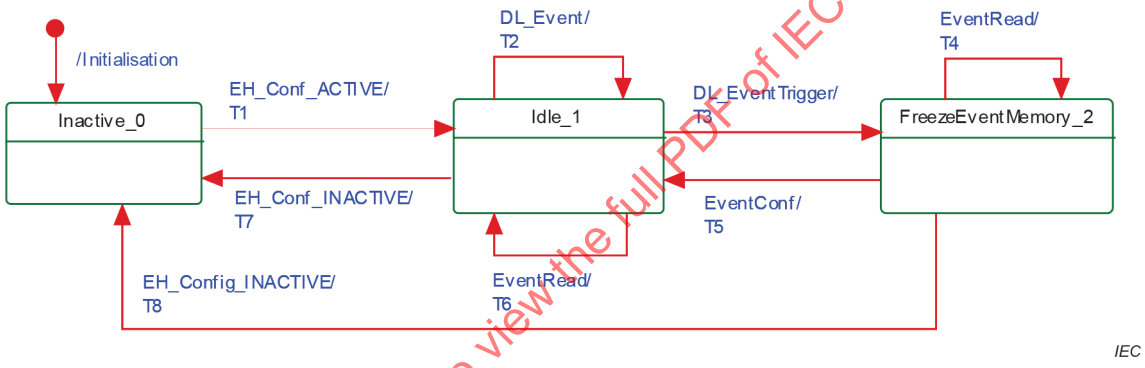


Figure 56 – Diagramme d'états du module de traitement d'Événements du Dispositif

Le Tableau 60 présente la table de transitions d'états du module de traitement d'Événements du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 60 – Table de transitions d'états du module de traitement d'Événements du Dispositif

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Inactive_0	Attente d'activation
Idle_1	Attente du service DL-Event de la couche AL qui fournit les données d'Événement et du service DL_EventTrigger pour établir le bit "drapeau d'Événement" (voir A.1.5)
FreezeEventMemory_2	Attente de la lecture de la mémoire d'Événements et de la confirmation de lecture de la mémoire d'Événements par Accès en écriture au Code d'État

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	-
T2	1	1	Remplacement des entrées de la mémoire d'Événements par de nouvelles données d'Événement (voir le Tableau 58)
T3	1	2	Invocation du service EventFlag.req (Drapeau = VRAI) pour indiquer l'activation d'Événements au Maître au moyen du bit "drapeau d'Événement". Marquage de tous les emplacements d'événements en mémoire comme non modifiables.
T4	2	2	Le Maître demande des données de la mémoire d'Événements par l'intermédiaire de EventRead (= OD.ind). Envoi de données d'Événement par invocation de OD.rsp avec des données d'Événement de l'adresse de la mémoire des Événements demandés.
T5	2	1	Invocation du service EventFlag.req (Drapeau = FAUX) pour indiquer la désactivation d'Événements au Maître par l'intermédiaire du bit "drapeau d'Événement". Marquage de tous les emplacements d'événements en mémoire comme non valides selon A.6.3.
T6	1	1	Envoi du contenu de la mémoire d'Événements par invocation de OD.rsp avec des données d'Événement
T7	1	0	-
T8	2	0	Vidage des données de la mémoire d'Événements

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
EventRead	Service	OD.ind (R, DIAGNOSIS, adresse de mémoire d'Événements, longueur, données)
EventConf	Service	OD.ind (W, DIAGNOSIS, adresse = 0, données = sans effet)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8 Couche d'application (AL)

8.1 Généralités

La Figure 57 donne une vue d'ensemble de la structure et des services de la couche d'application (AL) du Maître.

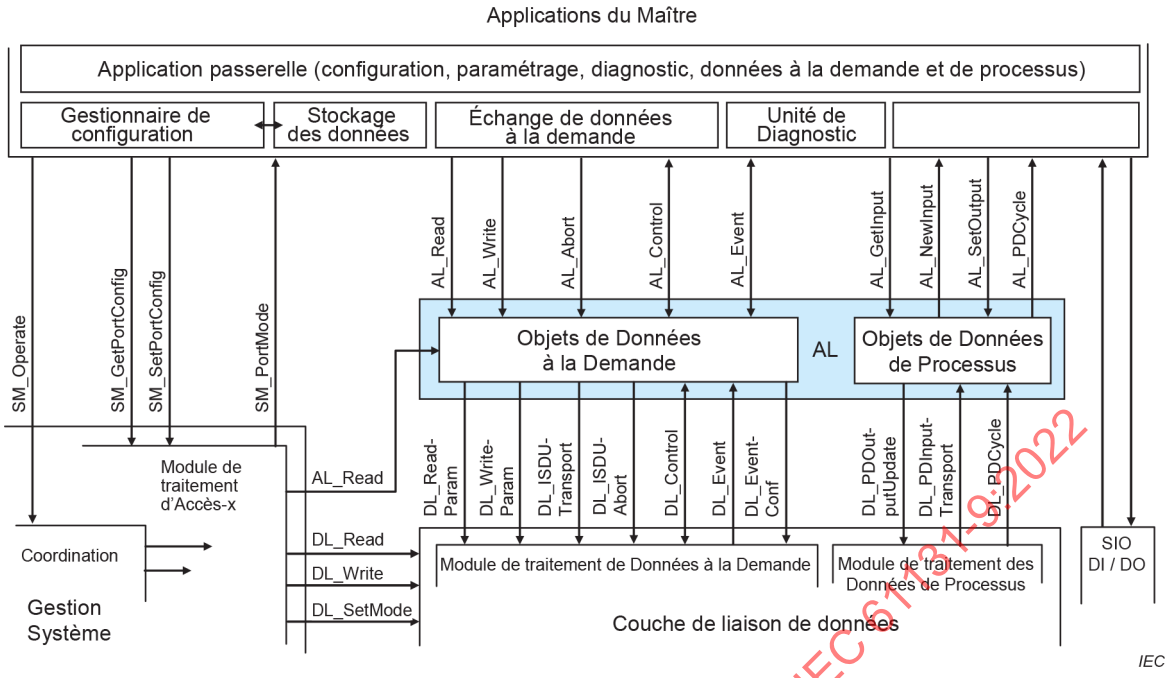
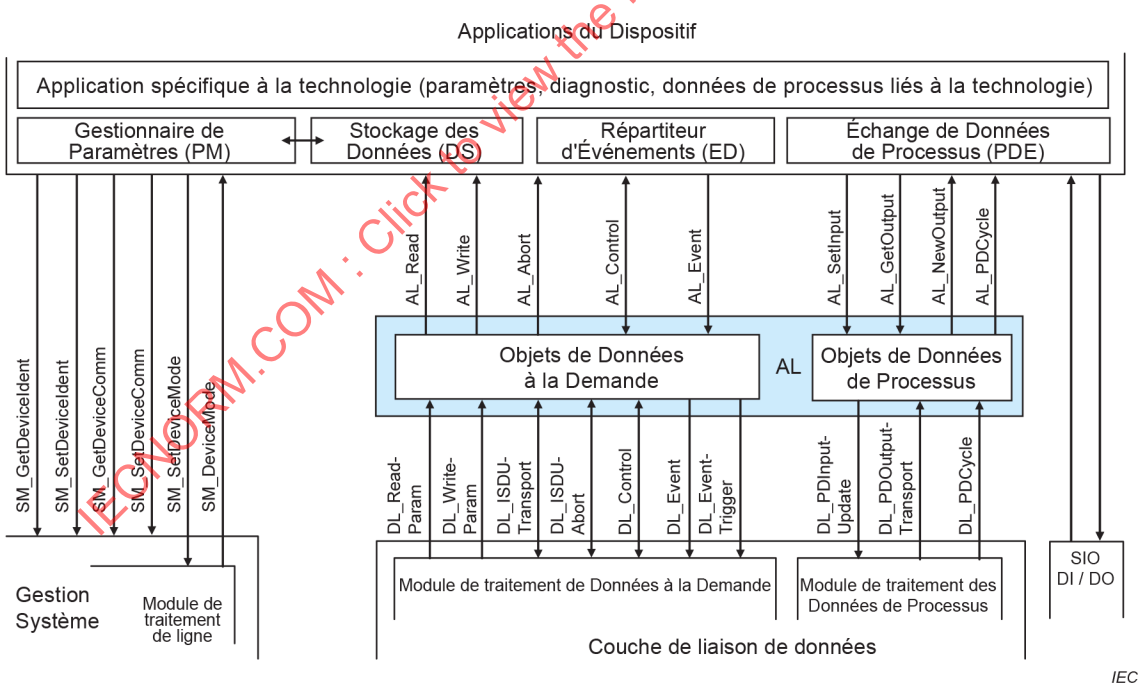


Figure 57 – Structure et services de la couche d'application (Maître)

La Figure 58 donne une vue d'ensemble de la structure et des services de la couche d'application (AL) du Dispositif.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 58 – Structure et services de la couche d'application (Dispositif)

8.2 Services de la Couche d'Application

8.2.1 Services AL au sein du Maître et du Dispositif

Le présent Article 8.2 définit les services de la couche d'application (AL) à fournir au Maître, aux applications de Dispositif et à la Gestion Système par l'intermédiaire de ses interfaces externes. Le Tableau 61 énumère les attributions du Maître et du Dispositif quant à leur rôle

d'initiateur ou de destinataire des services individuels de couche d'application (AL). Des champs vides indiquent la non-disponibilité de ce service au niveau du Maître ou du Dispositif.

Tableau 61 – Services AL au sein du Maître et du Dispositif

Désignation du service	Maître	Dispositif
AL_Read	R	I
AL_Write	R	I
AL_Abort	R	I
AL_GetInput	R	
AL_NewInput	I	
AL_SetInput		R
AL_PDCycle	I	I
AL_GetOutput		R
AL_NewOutput		I
AL_SetOutput	R	
AL_Event	I / R	R
AL_Control	I / R	R / I
Légende (voir 3.3.4) I Initiateur du service R Destinataire (Répondeur) du service		

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.2.2 Services d'AL

8.2.2.1 AL_Read

Le service AL_Read est utilisé pour lire des Données à la Demande en provenance d'un Dispositif connecté à un Accès spécifique. Les paramètres des primitives de service sont énumérés dans le Tableau 62.

Tableau 62 – AL_Read

Nom du paramètre	.req	.ind	.rsp	.cnf
Argument	M	M		
Accès	M			
Index	M	M		
Sous-index	M	M		
Résultat (+)			S	S(=)
Accès				M
Données			M	M(=)
Résultat (−)			S	S(=)
Accès				M
ErrorInfo				M(=)

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès pour les Données à la Demande à lire.

Type de paramètre: Unsigned8.

Index

Ce paramètre indique l'adresse des objets de Données à la Demande à lire en provenance du Dispositif. L'Index 0 conjointement au Sous-index 0 couvre l'ensemble du jeu de Paramètres Directs de 0 à 15 (voir la page 1 de Paramètres Directs dans le Tableau B.1) ou, conjointement aux Sous-index 1 à 16, il couvre les paramètres individuels de 0 à 15. L'Index 1 conjointement au Sous-index 0 couvre l'ensemble du jeu de Paramètres Directs des adresses 16 à 31 (voir la page 2 de Paramètres Directs dans le Tableau B.1) ou, conjointement aux Sous-index 1 à 16, il couvre les paramètres individuels de 16 à 31. Il utilise le canal de communication de page (voir la Figure 7) dans les deux cas et renvoie toujours un résultat positif. Le canal de communication des ISDU est utilisé pour tous les autres index (voir l'Article B.2).

Valeurs admises: 0 à 65 535 (voir B.2.1 pour les contraintes).

Sous-index

Ce paramètre indique le numéro d'élément dans un objet structuré de Données à la Demande. Une valeur 0 indique l'ensemble du jeu d'éléments.

Valeurs admises: 0 à 255.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Accès

Ce paramètre contient le numéro d'Accès des Données à la Demande requises.

Données

Ce paramètre contient les valeurs lues des Données à la Demande.

Type de paramètre: Chaîne d'octets.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Accès

Ce paramètre contient le numéro d'Accès pour les Données à la Demande requises.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises: voir l'Annexe C.

NOTE La couche AL met en correspondance les ErrorInfos de couche DL avec ses propres ErrorInfos de couche AL en utilisant l'Annexe C.

8.2.2.2 AL_Write

Le service AL_Write est utilisé pour écrire des Données à la Demande vers un Dispositif connecté à un Accès spécifique. Les paramètres des primitives de service sont énumérés dans le Tableau 63.

Tableau 63 – AL_Write

Nom du paramètre	.req	.ind	.rsp	.cnf
Argument	M	M		
Accès	M			
Index	M	M		
Sous-index	M	M		
Données	M	M(=)		
Résultat (+)			S	S(=)
Accès				M
Résultat (−)			S	S(=)
Accès				M
ErrorInfo			M	M(=)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès pour les Données à la Demande à écrire.

Type de paramètre: Unsigned8.

Index

Ce paramètre indique l'adresse des objets de Données à la Demande à écrire vers le Dispositif. L'Index 0 renvoie toujours un résultat négatif, sauf en cas d'utilisation conjointement avec le Sous-index 16 sur des Dispositifs sans support ISDU. L'Index 1 conjointement au Sous-index 0 couvre l'ensemble du jeu de Paramètres Directs des adresses 16 à 31 (voir la page 2 de Paramètres Directs dans le Tableau B.1) ou, conjointement aux Sous-index 1 à 16, il couvre les paramètres individuels de 16 à 31. Il utilise le canal de communication de page (voir la Figure 7) dans le cas de l'index 1 et renvoie toujours un résultat positif. Le canal de communication des ISDU est utilisé pour tous les autres index (voir l'Article B.2).

Valeurs admises: 1 à 65 535 (voir le Tableau 102).

Sous-index

Ce paramètre indique le numéro d'élément dans un objet structuré de Données à la Demande. Une valeur 0 indique l'ensemble du jeu d'éléments.

Valeurs admises: 0 à 255.

Données

Ce paramètre contient les valeurs des Données à la Demande.

Type de paramètre: Chaîne d'octets.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Accès

Ce paramètre contient le numéro d'Accès des Données à la Demande.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Accès

Ce paramètre contient le numéro d'Accès des Données à la Demande.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises: voir l'Annexe C.

8.2.2.3 AL_Abort

Le service AL_Abort est utilisé pour abandonner un service AL_Read ou AL_Write en cours sur un Accès spécifique. L'invocation de ce service abandonne la réponse à un service AL_Read ou AL_Write en cours au niveau du Maître. Les paramètres des primitives de service sont énumérés dans le Tableau 64.

Tableau 64 – AL_Abort

Nom du paramètre	.req	.ind
Argument Accès	M M	M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Le paramètre spécifique au service est transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès du service à abandonner.

8.2.2.4 AL_GetInput

Le service AL_GetInput lit les données d'entrée dans les Données de Processus fournies par la couche de liaison de données d'un Dispositif connecté à un Accès spécifique. Les paramètres des primitives de service sont énumérés dans le Tableau 65.

Tableau 65 – AL_GetInput

Nom du paramètre	.req	.cnf
Argument	M	
Accès	M	
Résultat (+)		S
Accès		M
InputData		M
Résultat (−)		S
Accès		M
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès pour les Données de Processus à lire.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Accès

Ce paramètre contient le numéro d'Accès des Données de Processus.

InputData

Ce paramètre contient les valeurs des Données de Processus d'entrée demandées de l'Accès spécifié.

Type de paramètre: Chaîne d'octets.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Accès

Ce paramètre contient le numéro d'Accès des Données de Processus.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_DATA (La DL n'a pas fourni de Données de Processus).

8.2.2.5 AL_NewInput

Le service local AL_NewInput indique la réception de données d'entrée mises à jour dans les Données de Processus d'un Dispositif connecté à un Accès spécifique. Les paramètres des primitives de service sont énumérés dans le Tableau 66.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 66 – AL_NewInput

Parameter name	.ind
Argument Accès	M M

Argument

Le paramètre spécifique au service est transmis dans l'argument.

Accès

Ce paramètre spécifie le numéro d'Accès des Données de Processus reçues.

8.2.2.6 AL_SetInput

Le service local AL_SetInput met à jour les données d'entrée dans les Données de Processus d'un Dispositif. Les paramètres des primitives de service sont énumérés dans le Tableau 67.

Tableau 67 – AL_SetInput

Nom du paramètre	.req	.cnf
Argument InputData	M M	
Résultat (+)		S
Résultat (−) ErrorInfo		S M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

InputData

Ce paramètre contient les valeurs de Données de Processus des données d'entrée à transmettre.

Type de paramètre: Chaîne d'octets.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (Service non disponible en l'état courant).

8.2.2.7 AL_PDCycle

Le service local AL_PDCycle indique la fin d'un cycle de Données de Processus. L'application du Dispositif peut utiliser ce service pour transmettre de nouvelles données d'entrée à la couche d'application par l'intermédiaire de AL_SetInput. Les paramètres des primitives de service sont énumérés dans le Tableau 68.

Tableau 68 – AL_PDCycle

Nom du paramètre	.ind
Argument Accès	 O

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Le paramètre spécifique au service est transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès des nouvelles Données de Processus reçues (Maître uniquement).

8.2.2.8 AL_GetOutput

Le service AL_GetOutput lit les données de sortie dans les Données de Processus fournies par la couche de liaison de données du Dispositif. Les paramètres des primitives de service sont énumérés dans le Tableau 69.

Tableau 69 – AL_GetOutput

Nom du paramètre	.req	.cnf
Argument	M	
Résultat (+)		S
OutputData		M
Résultat (−)		S
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

OutputData

Ce paramètre contient les valeurs de Données de Processus des données de sortie demandées.

Type de paramètre: Chaîne d'octets.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
NO_DATA (La DL n'a pas fourni de Données de Processus).

8.2.2.9 AL_NewOutput

Le service local AL_NewOutput indique la réception de données de sortie mises à jour dans les Données de Processus d'un Dispositif. Ce service n'a pas de paramètres. Les primitives de service sont présentées dans le Tableau 70.

Tableau 70 – AL_NewOutput

Nom du paramètre	.ind
<aucun>	

8.2.2.10 AL_SetOutput

Le service local AL_SetOutput met à jour les données de sortie dans les Données de Processus d'un Maître. Les paramètres des primitives de service sont énumérés dans le Tableau 71.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 71 – AL_SetOutput

Nom du paramètre	.req	.cnf
Argument	M	
Accès	M	
OutputData	M	
Résultat (+)		S
Accès		M
Résultat (−)		S
Accès		M
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès des Données de Processus à écrire.

OutputData

Ce paramètre contient les données de sortie à écrire à l'Accès spécifié.

Type de paramètre: Chaîne d'octets.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Accès

Ce paramètre contient le numéro d'Accès des Données de Processus.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Accès

Ce paramètre contient le numéro d'Accès des Données de Processus.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (Service non disponible en l'état courant).

8.2.2.11 AL_Event

Le service AL_Event indique jusqu'à six états ou messages d'erreur en cours. L'origine d'un Événement peut être locale (Maître) ou distante (Dispositif). L'Événement peut être déclenché par une couche de communication ou par une application. Les paramètres des primitives de service sont énumérés dans le Tableau 72.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 72 – AL_Event

Nom du paramètre		.req	.ind	.rsp	.cnf
Argument		M	M	M	M
Accès			M	M	M
EventCount		M	M		
Événement(1)	Instance	M	M		
	Mode	M	M		
	Type	M	M		
	Origin		M		
	EventCode	M	M		
…					
Événement (n)	Instance	M	M		
	Mode	M	M		
	Type	M	M		
	Origin		M		
	EventCode	M	M		

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Accès

Ce paramètre contient le numéro d'Accès des données d'Événement.

EventCount

Ce paramètre indique le nombre n (1 à 6) d'Événements dans la mémoire d'Événements.

Event(x)

En fonction de EventCount, ce paramètre existe n fois. Chaque instance contient les éléments suivants:

Instance

Ce paramètre indique l'origine de l'Événement.

Valeurs admises: Application (voir le Tableau A.17).

Mode

Ce paramètre indique le mode de l'Événement.

Valeurs admises: SINGLESHOT, APPEARS, DISAPPEARS (voir le Tableau A.20).

Type

Ce paramètre indique la catégorie de l'Événement.

Valeurs admises: ERROR, WARNING, NOTIFICATION (voir le Tableau A.19).

Origine

Ce paramètre indique si l'Événement a été généré dans la section de communication locale ou à distance (dans le Dispositif).

Valeurs admises: LOCAL, DISTANT.

EventCode

Ce paramètre contient un code identifiant un certain Événement.

Valeurs admises: voir l'Annexe D.

8.2.2.12 AL_Control

Le service AL_Control contient les informations d'état du qualificatif des Données de Processus transmises à et provenant de l'application du Dispositif. Ce service doit être synchronisé avec AL_GetInput et AL_SetOutput respectivement (voir 11.7.2.1). Les paramètres des primitives de service sont énumérés dans le Tableau 73.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 73 – AL_Control

Nom du paramètre	.req	.ind

Argument
  Accès
  ControlCode
	M
	C
	M
		M
		C
		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Accès

Ce paramètre contient le numéro de l'Accès concerné.

ControlCode

Ce paramètre contient l'état du qualificatif des Données de Processus (PD).

Valeurs admises:

VALID	(Données de Processus d'entrée valides),
INVALID	(Données de Processus d'entrée non valides),
PDOUTVALID	(Données de Processus de sortie valides; voir le Tableau 55),
PDOUTINVALID	(Données de Processus de sortie non valides, voir le Tableau 55).

8.3 Protocole de couche d'application

8.3.1 Vue d'ensemble

La Figure 8 indique que la couche d'application offre des services aux objets de données qui sont transformés en canaux de communication spéciaux de la couche de liaison de données.

La couche d'application gère le transfert de données avec tous les Accès qui lui sont attribués. Ce qui signifie qu'il est nécessaire que les invocations de service AL identifient l'Accès particulier auquel elles sont associées.

8.3.2 Transfert de Données à la Demande

8.3.2.1 Diagramme d'états d'OD de la couche AL du Maître

La Figure 59 représente le diagramme d'états utilisé pour le traitement des Données à la Demande (OD) dans la couche d'application.

L'expression "AL_Service" se réfère à tout service AL du Tableau 61 et concernant les OD. "Portx" indique un numéro d'Accès particulier.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

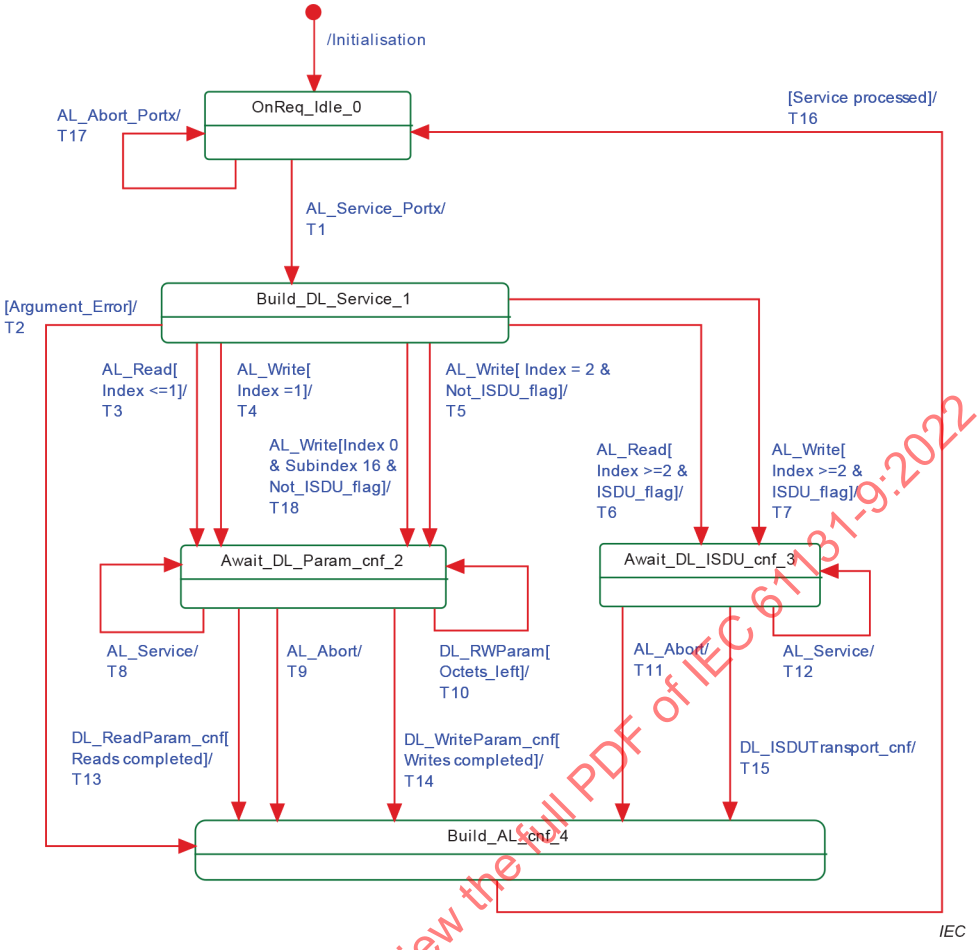


Figure 59 – Diagramme d'états d'OD de la couche AL du Maître

Le Tableau 74 présente les états et transitions pour le diagramme d'états d'OD de la couche AL du Maître.

Tableau 74 – États et transitions pour le diagramme d'états d'OD de la couche AL du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
OnReq_Idle_0	Les invocations de services AL en provenance des applications du Maître ou du module de traitement de Portx de la SM (voir la Figure 57) peuvent être acceptées dans cet état.
Build_DL_Service_1	Dans cet état, les invocations de service AL sont vérifiées et les services DL correspondants sont générés dans les états ultérieurs. En cas d'erreur dans les arguments du service de couche AL, une confirmation AL négative est générée et renvoyée.
Await_DL_Param_cnf_2	Dans cet état, l'invocation de service AL est transformée en une séquence comportant autant d'invocations DL_ReadParam ou DL_WriteParam que nécessaire (Accès à la page de Paramètres Directs; voir le canal de communication de page à la Figure 7). Toutes les invocations de service AL en asynchrone, à l'exception de AL_Abort sont rejetées (voir 3.3.8).
Await_DL_ISDU_cnf_3	Dans cet état, l'invocation de service AL est transformée en une invocation de service DL_ISDUTransport (voir canal de communication des ISDU à la Figure 7). Toutes les invocations de service AL en asynchrone, à l'exception de AL_Abort sont rejetées (voir 3.3.8).
Build_AL_cnf_4	Dans cet état, une confirmation de service AL est générée en fonction d'une erreur d'argument, de la confirmation de service DL ou d'un service AL_Abort.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Mémorisation du numéro d'Accès "Accèsx".
T2	1	4	Préparation d'une confirmation de service AL négative.
T3	1	2	Préparation de DL_ReadParam pour l'Index 0 ou 1.
T4	1	2	Préparation de DL_WriteParam pour l'Index 1.
T5	1	2	Préparation de Write pour l'Adresse 0x0F si le Dispositif ne prend pas en charge les ISDU.
T6	1	3	Préparation de DL_ ISDUTransport (read)
T7	1	3	Préparation de DL_ ISDUTransport (write)
T8	2	2	Renvoi d'une confirmation de service AL négative sur cette invocation de service asynchrone.
T9	2	4	Toutes les actions courantes de service DL sont abandonnées et une confirmation de service AL négative est préparée.
T10	2	2	Invocation du service DL_ReadParam ou DL_WriteParam suivant si les OD n'ont pas toutes été transférées.
T11	3	4	Toutes les actions courantes de service DL sont abandonnées et une confirmation de service AL négative est préparée.
T12	3	3	Renvoi d'une confirmation de service AL négative sur cette invocation de service asynchrone.
T13	2	4	Préparation d'une confirmation de service AL positive.
T14	2	4	Préparation d'une confirmation de service AL positive.
T15	3	4	Préparation d'une confirmation de service AL positive.
T16	4	0	Renvoi d'une confirmation de service AL positive avec le numéro d'Accès "Accèsx".
T17	0	0	Renvoi d'une confirmation de service AL négative avec le numéro d'Accès "Accèsx".
T18	1	2	Préparation de DL_Write pour l'Adresse 0x0F si le Dispositif ne prend pas en charge les ISDU.

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
Argument_Error	Bool	Valeurs illicites dans le corps du service, par exemple "Numéro d'Accès ou Index hors limites"
DL_RWParam	Étiquette	"DL_RWParam": DL_WriteParam_cnf ou DL_ReadParam_cnf
Completed	Bool	Il ne reste plus d'autres OD à transférer
Octets_left	Bool	Il reste d'autres OD à transférer
Portx	Variable	Variable du corps du service indiquant le numéro d'Accès
ISDU_Flag	Bool	Le Dispositif prend en charge les ISDU
AL_Service	Étiquette	L'expression "AL_Service" se réfère à tout service AL du Tableau 61 et concernant les OD.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.2.2 Diagramme d'états d'OD de la couche AL du Dispositif

La Figure 60 représente le diagramme d'états utilisé pour le traitement des Données à la Demande (OD) dans la couche d'application d'un Dispositif.

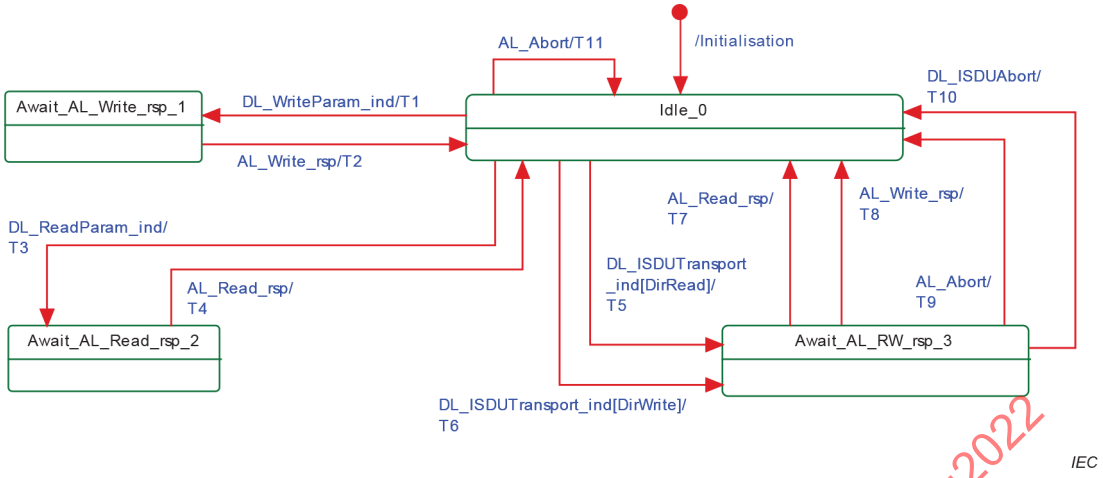


Figure 60 – Diagramme d'états d'OD de la couche AL du Dispositif

Le Tableau 75 présente les états et transitions pour le diagramme d'états d'OD de la couche AL du Dispositif.

Tableau 75 – États et transitions pour le diagramme d'états d'OD de la couche AL du Dispositif

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Idle_0	La couche AL du Dispositif est en attente d'invocations de service DL subordonnées déclenchées par des messages du Maître.
Await_AL_Write_rsp_1	La couche AL du Dispositif est en attente d'une réponse de l'application spécifique à la technologie (Accès en écriture à la page de Paramètres Directs).
Await_AL_Read_rsp_2	La couche AL du Dispositif est en attente d'une réponse de l'application spécifique à la technologie (Accès en lecture à la page de Paramètres Directs).
Await_AL_RW_rsp_3	La couche AL du Dispositif est en attente d'une réponse de l'application spécifique à la technologie (Accès en lecture ou en écriture par l'intermédiaire des ISDU).

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Invocation de AL_Write.
T2	1	0	Invocation de DL_WriteParam (16 à 31).
T3	0	2	Invocation de AL_Read.
T4	2	0	Invocation de DL_ReadParam (0 à 31).
T5	0	3	Invocation de AL_Read.
T6	0	3	Invocation de AL_Write.
T7	3	0	Invocation de DL_ ISDUTransport (read)
T8	3	0	Invocation de DL_ ISDUTransport (write)
T9	3	0	AL_Read ou AL_Write en cours abandonné sur cette invocation de service asynchrone AL_Abort. Renvoi de DL_ ISDUTransport négatif (voir 3.3.8).
T10	3	0	Attente en cours de AL_Read ou AL_Write abandonnée.
T11	0	0	DL_ ISDUTransport courant abandonné. Toutes les OD sont réglées à "0".

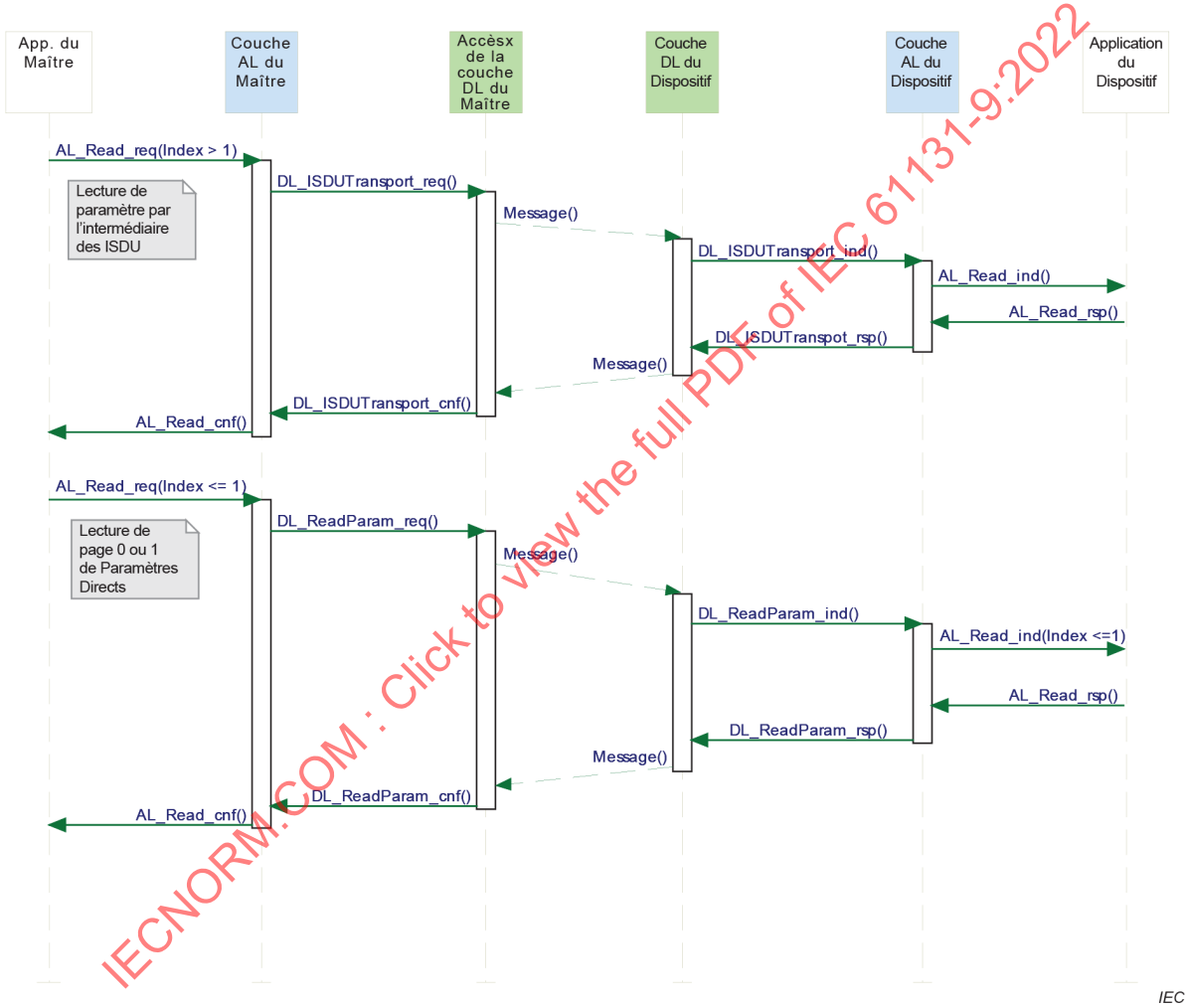
ÉLÉMENTS INTERNES	TYPE	DÉFINITION
DirRead	Bool	Sens d'Accès: DL_ ISDUTransport (read) force un AL_Read
DirWrite	Bool	Sens d'Accès: DL_ ISDUTransport (write) force un AL_Read

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.2.3 Diagrammes de séquence pour les Données à la Demande

Les figures de la Figure 61 à la Figure 63 représentent l'ensemble des interactions entre le Maître et le Dispositif pour plusieurs cas d'utilisation de l'échange de Données à la Demande.

La Figure 61 représente deux exemples d'échange de Données à la Demande. Pour des index > 1, cet échange est réalisé à l'aide d'unités ISDU et des services DL correspondants (canal de communication des ISDU, conformément à la Figure 7). Les Accès aux pages 0 et 1 de Paramètres Directs utilisent des services DL différents (canal de communication de page, conformément à la Figure 7).



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 61 – Diagramme de séquence pour la transmission de Données à la Demande

La Figure 62 représente le comportement de l'échange de Données à la Demande en cas d'erreur, par exemple lorsque l'Index demandé n'est pas disponible (voir le Tableau C.1).

Une autre erreur possible a lieu lorsque l'application du Maître (passerelle) tente de lire un Index > 1 à partir d'un Dispositif qui ne prend pas en charge les ISDU. La couche AL du Maître répond immédiatement par "NO_ISDU_SUPPORTED" (PAS DE PRISE EN CHARGE D'ISDU) étant donné que les fonctionnalités du Dispositif sont acquises pendant le démarrage par lecture de la page 1 de Paramètres Directs au moyen du paramètre "Capacité de séquence M" (voir le Tableau B.1).

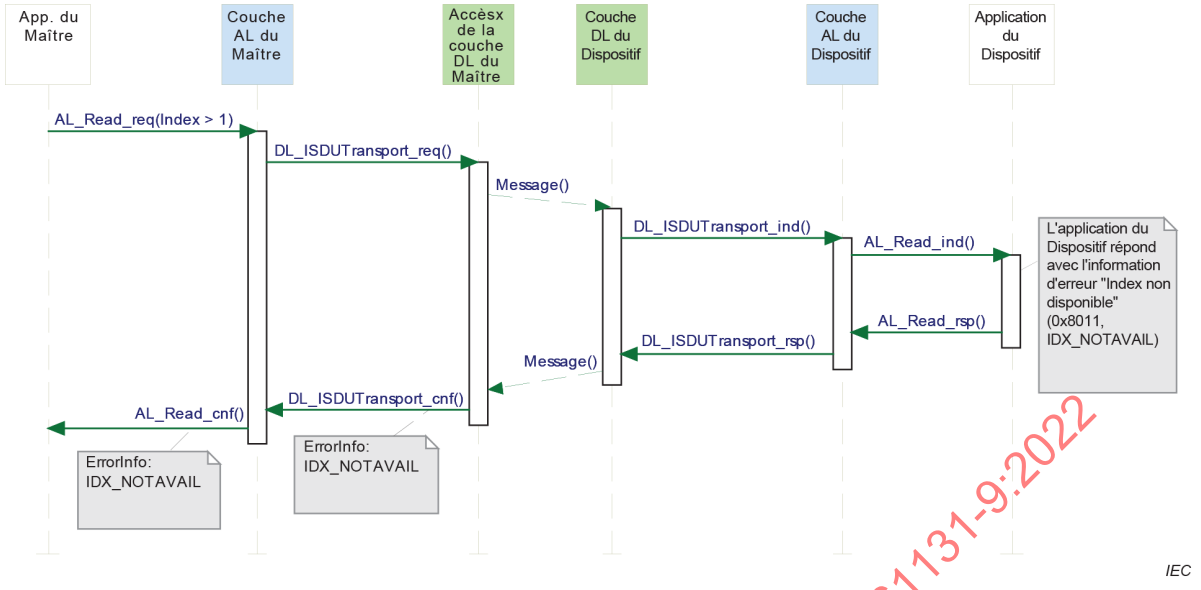


Figure 62 – Diagramme de séquence pour des Données à la Demande en cas d'erreurs

La Figure 63 représente le comportement de l'échange de Données à la Demande en cas d'expiration du délai des ISDU (5 500 ms). Un Dispositif doit répondre dans un délai inférieur au "délai d'acquittement des ISDU" (voir 10.8.5).

NOTE Pour les constantes du système telles que le "délai d'acquittement des ISDU", voir le Tableau 102.

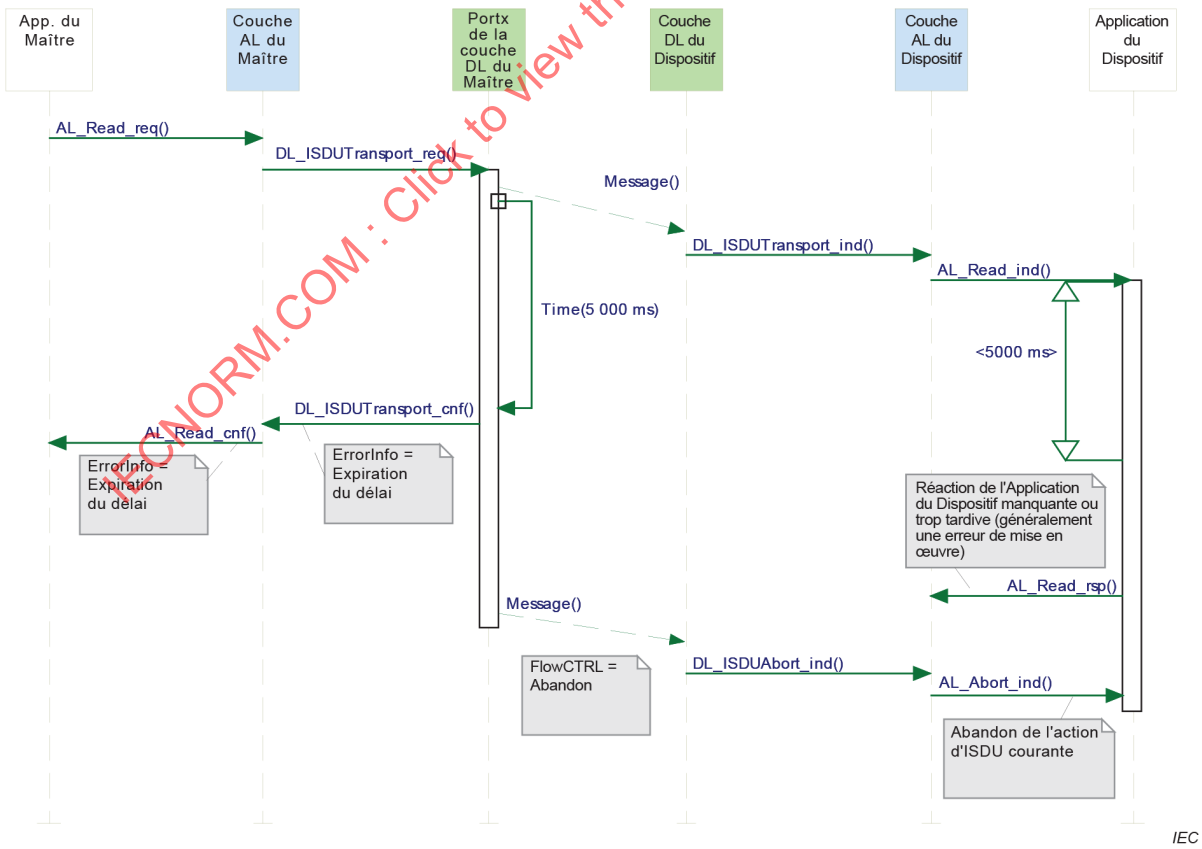


Figure 63 – Diagramme de séquence pour des Données à la Demande en cas d'expiration de délai

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

8.3.3 Traitement des Événements

8.3.3.1 Diagramme d'états d'Événements de la couche AL du Maître

La Figure 64 représente le diagramme d'états d'Événements de la couche d'application du Maître.

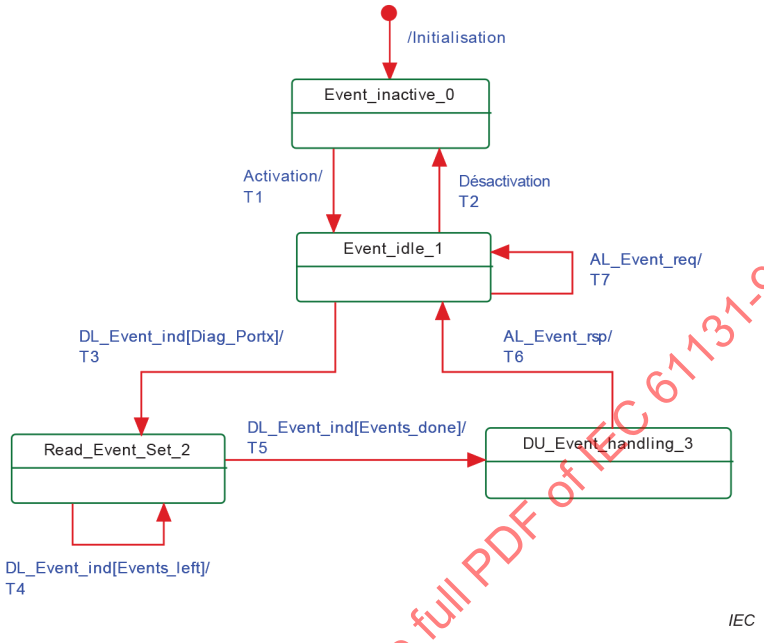


Figure 64 – Diagramme d'états d'Événements de la couche AL du Maître

Le Tableau 76 définit les états et transitions du diagramme d'états d'Événements de la couche d'application du Maître.

Tableau 76 – États et transitions du diagramme d'états d'Événements de la couche AL du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Event_inactive_0	Le traitement d'Événements de couche AL du Maître est inactif.
Event_idle_1	La couche AL du Maître est prête à accepter des Événements DL (informations de diagnostic) à partir de la couche DL.
Read_Event_Set_2	La couche AL du Maître a reçu une DL_Event_ind avec des informations de diagnostic. Après cette première DL_Event.ind, la couche AL recueille l'ensemble du jeu (1 à 6) d'Événements DL du EventTrigger courant (voir 11.6).
DU_Event_handling_3	La couche AL du Maître reste dans cet état tant que l'Unité de Diagnostic (voir 11.6) n'a pas acquitté la AL_Event.ind.

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	–
T2	1	0	–
T3	1	2	–
T4	2	2	–
T5	2	3	AL_Event.ind
T6	3	1	DL_EventConf.req
T7	1	1	AL_Event.ind

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
Diag_Portx	Bool	Le jeu d'Événements contient des informations de diagnostic détaillées.
Events_done	Bool	Le jeu d'Événements est traité.
Events_left	Bool	Le jeu d'Événements n'est pas encore terminé

8.3.3.2 Diagramme d'états d'Événements de la couche AL du Dispositif

La Figure 65 représente le diagramme d'états d'Événements de la couche d'application du Dispositif.

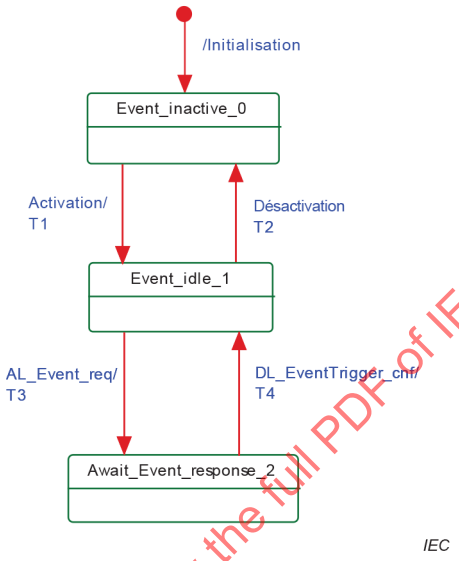


Figure 65 – Diagramme d'états d'Événements de la couche AL du Dispositif

Le Tableau 77 définit les états et transitions du diagramme d'états d'Événements de la couche d'application du Dispositif.

Tableau 77 – États et transitions du diagramme d'états d'Événements de la couche AL du Dispositif

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
Event_inactive_0	Le traitement d'Événements de couche AL du Dispositif est inactif.
Event_idle_1	La couche AL du Dispositif est prête à accepter des Événements AL (informations de diagnostic) à partir des applications spécifiques à la technologie du Dispositif pour transfert à la couche DL. Pendant ce temps, les applications du Dispositif peuvent générer de nouveaux Événements.
Await_event_response_2	La couche AL du Dispositif a propagé un AL_Event avec des informations de diagnostic et attend une confirmation DL_EventTrigger de la couche DL. Pendant ce temps, la couche AL du Dispositif ne doit pas accepter un quelconque AL_Event nouveau.

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	–
T2	1	0	–
T3	1	2	Une demande AL_Event déclenche un DL_Event et le service DL_EventTrigger correspondant. Le DL_Event achemine les informations de diagnostic de la couche AL à la couche DL. Le service DL_EventTrigger établit le drapeau d'Événement dans l'échange des données cycliques (voir A.1.5)
T4	2	1	Une confirmation de DL_EventTrigger déclenche une confirmation d'AL_Event.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
aucun		

8.3.3.3 Planification d'Événements uniques

La Figure 66 représente la manière dont est traité un Événement unique en provenance d'un Dispositif, conformément aux diagrammes d'états correspondants.

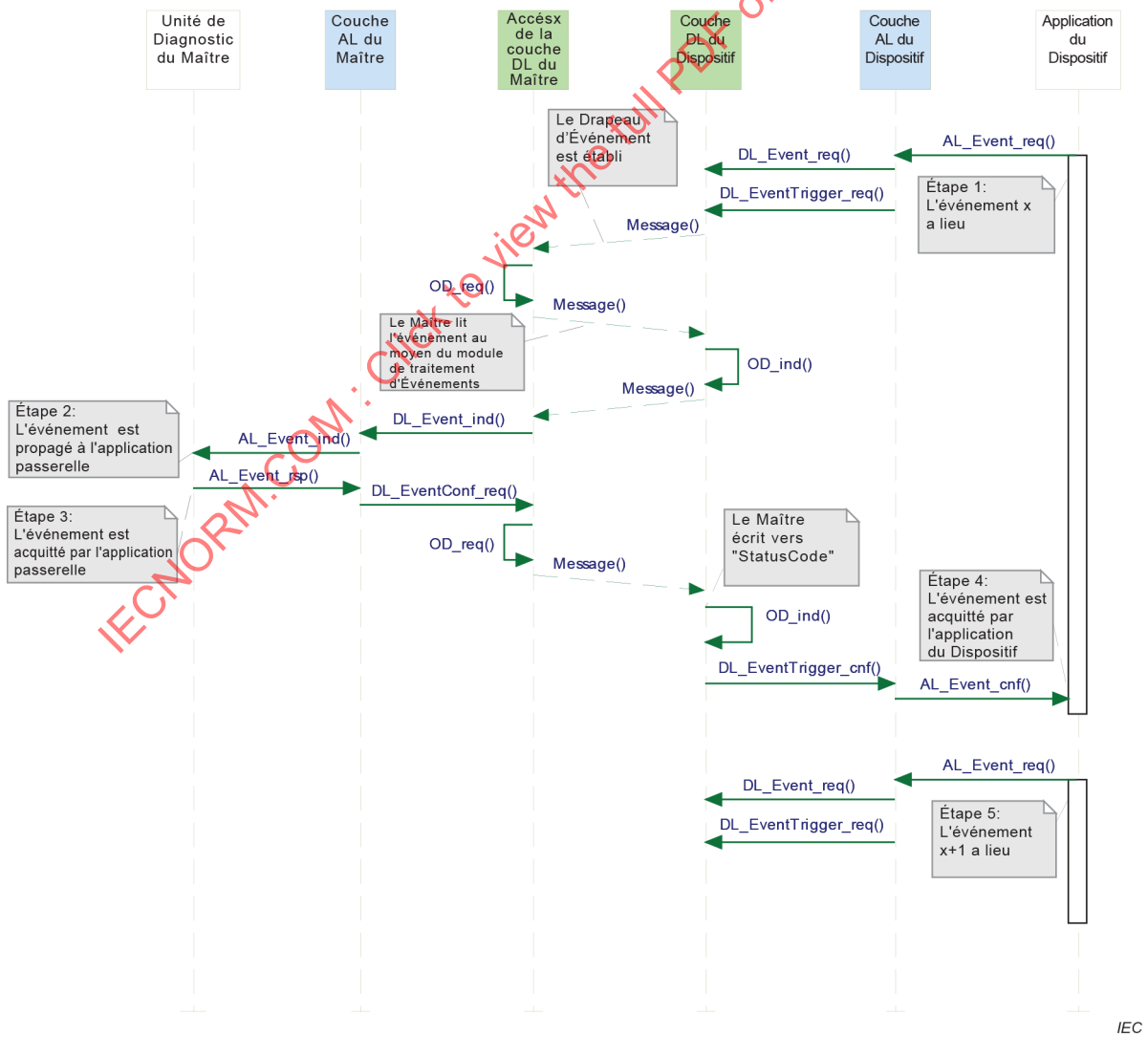
– L'application du Dispositif génère une demande d'Événement (Étape 1), qui est acheminée de la couche AL à la couche DL et mise en mémoire-tampon d'Événements (voir Tableau 58).

– La couche AL du Dispositif active le service EventTrigger pour établir le drapeau d'Événement et de ce fait le Maître lit l'Événement de la mémoire d'Événements.

– Le Maître propage ensuite cet Événement à l'application passerelle (Étape 2), et attend l'acquittement de l'Événement.

– Dès la réception de l'acquittement de l'Événement (Étape 3), il est indiqué au Dispositif par écriture dans le Code d'État (Étape 4).

– Le Dispositif confirme la demande d'Événement initiale à son application (Étape 5), et celle-ci peut à présent lancer une nouvelle demande d'Événement.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 66 – Planification d'Événements uniques

8.3.3.4 Transport d'Événements multiples (Dispositifs hérités uniquement)

Outre la méthode décrite en 8.3.3.3 dans le cadre de laquelle chaque Événement unique est acheminé par les couches et acquitté par l'application passerelle, tous les Maîtres doivent prendre en charge ce qu'on appelle le "Transport d'Événements multiples" qui permet de transférer en une seule fois jusqu'à six Événements. La couche AL du Maître transfère le jeu d'Événements comme une indication de diagnostic unique à l'application passerelle et renvoie à l'application du Dispositif hérité un acquittement unique pour l'ensemble du jeu.

La Figure 66 s'applique également au transport d'Événements multiples, à l'exception du fait que ce transport utilise une indication DL_Event pour chaque emplacement de mémoire d'Événement et une indication unique AL_Event pour l'ensemble du jeu d'Événements.

Une demande AL_Event.req transporte jusqu'à six Événements et une AL_Event.ind indique jusqu'à six Événements en cours. Les services AL_Event.rsp et AL_Event.cnf font référence à l'ensemble du jeu d'Événements indiqué.

8.3.4 Cycles de Données de Processus

La Figure 67 et la Figure 68 représentent l'ensemble des interactions entre le Maître et le Dispositif pour plusieurs cas d'utilisation de Données de Processus de sortie et d'entrée.

La Figure 67 représente la manière dont les services AL et DL du Maître et du Dispositif sont impliqués dans l'échange cyclique de Données de Processus de sortie. L'application du Dispositif est en mesure d'acquérir les valeurs en cours des PD de sortie au moyen du service AL_GetOutput.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

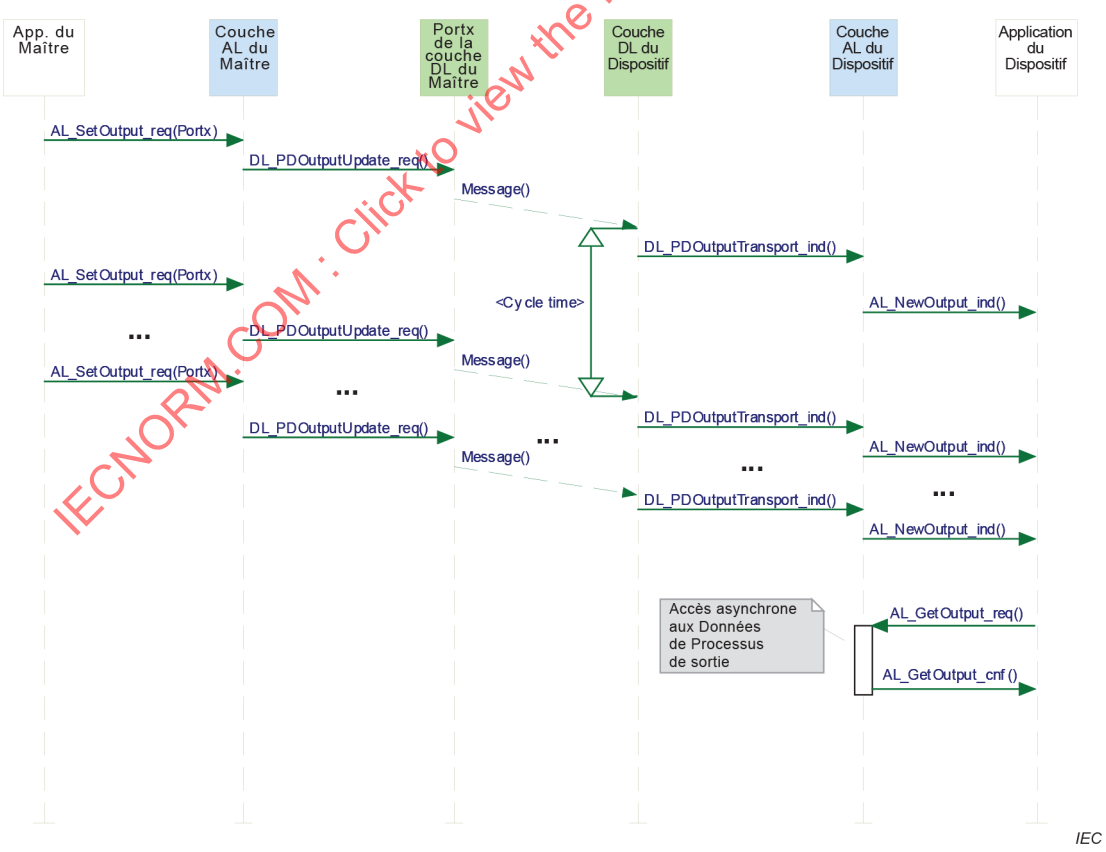


Figure 67 – Diagramme de séquence pour des Données de Processus de sortie

La Figure 68 représente la manière dont les services AL et DL du Maître et du Dispositif sont impliqués dans l'échange cyclique de Données de Processus d'entrée. L'application du Maître

est en mesure d'acquérir les valeurs en cours des PD d'entrée au moyen du service AL_GetInput.

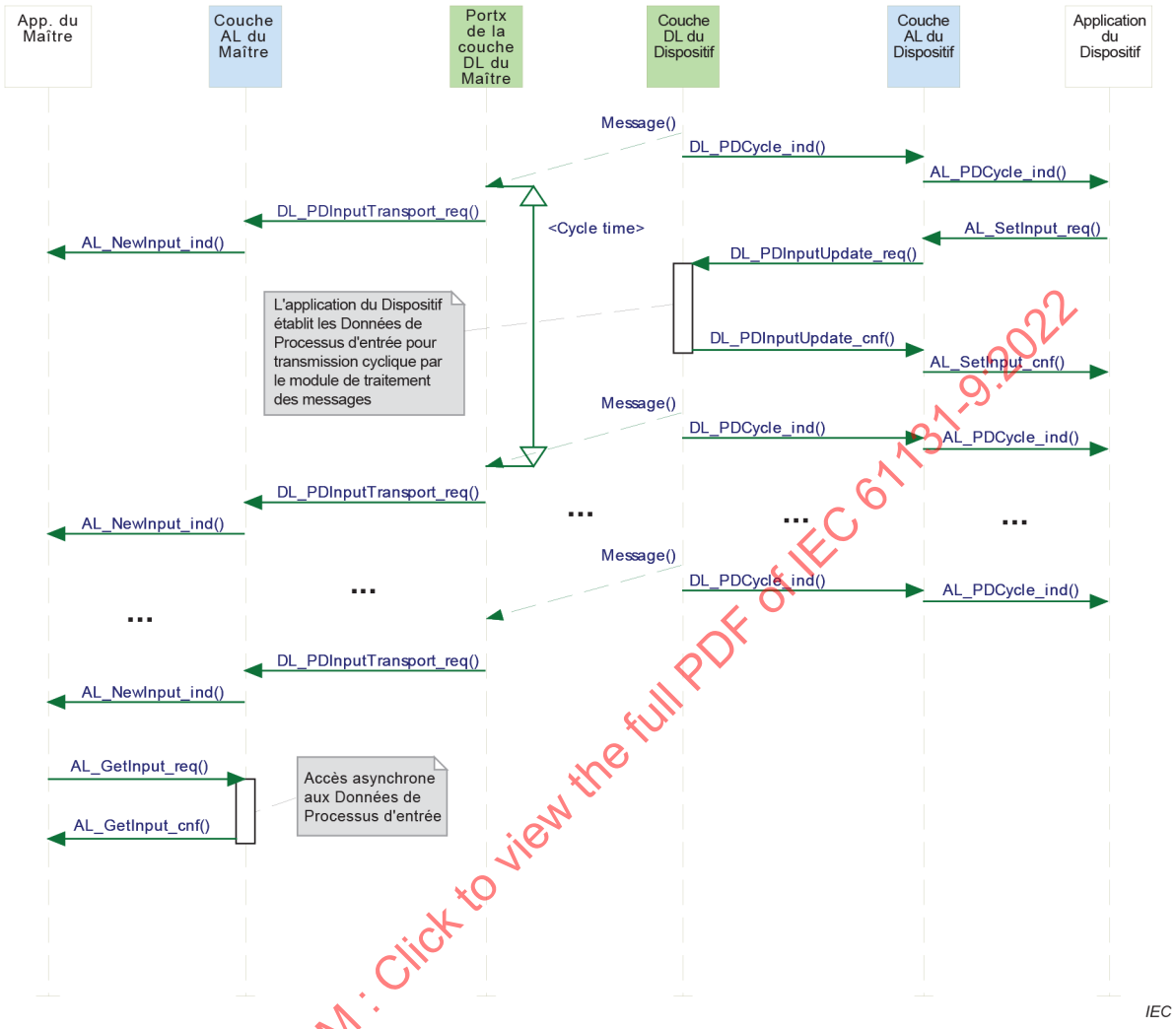


Figure 68 – Diagramme de séquence pour des Données de Processus d'entrée

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

9 Gestion Système (SM)

9.1 Généralités

La Gestion Système de l'interface SDCI est chargée d'assurer le démarrage coordonné des Accès du Maître, ainsi que les opérations correspondantes dans les Dispositifs connectés. La différence entre la SM du Maître et celle du Dispositif est plus significative que pour les autres couches. En conséquence, la structure du présent Article 9 traite de manière séparée les services et protocoles du Maître et du Dispositif.

9.2 Gestion Système du Maître

9.2.1 Vue d'ensemble

Les services de Gestion Système du Maître sont utilisés pour configurer les Accès du Maître ainsi que le système pour tous les modes opérationnels possibles.

La SM du Maître règle les Accès:

– en mettant en place la version exigée des protocoles de communication;

– en vérifiant la compatibilité du Dispositif (correspondance entre les identifications réelles du Dispositif et les valeurs prévues);

– en réglant les types de séquences M appropriés du Maître et les MasterCycleTimes.

Pour ce faire, la SM utilise les services suivants représentés à la Figure 69.

– SM_SetPortConfig transfère les paramètres de Dispositif nécessaires (données de configuration) de la Gestion de la Configuration (CM) à la Gestion Système (SM). L'Accès démarre ensuite implicitement.

– SM_PortMode rend compte du résultat positif de la configuration de l'Accès à la CM si la configuration et la vérification de l'Accès sont correctes. Il rend compte également du résultat négatif à la CM, par l'intermédiaire d'"erreurs" correspondantes en cas de discordance des versions et d'incompatibilité des Dispositifs.

– SM_GetPortConfig lit les paramètres réels et effectifs.

– SM_Operate commute un seul Accès en mode "OPERATE".

La Figure 69 donne une vue d'ensemble de la structure et des services de la Gestion Système du Maître.

La Gestion Système du Maître nécessite un service de couche d'application (AL_Read) pour acquérir, à des fins de vérification, les données (paramètre d'identification) à partir d'index spéciaux.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

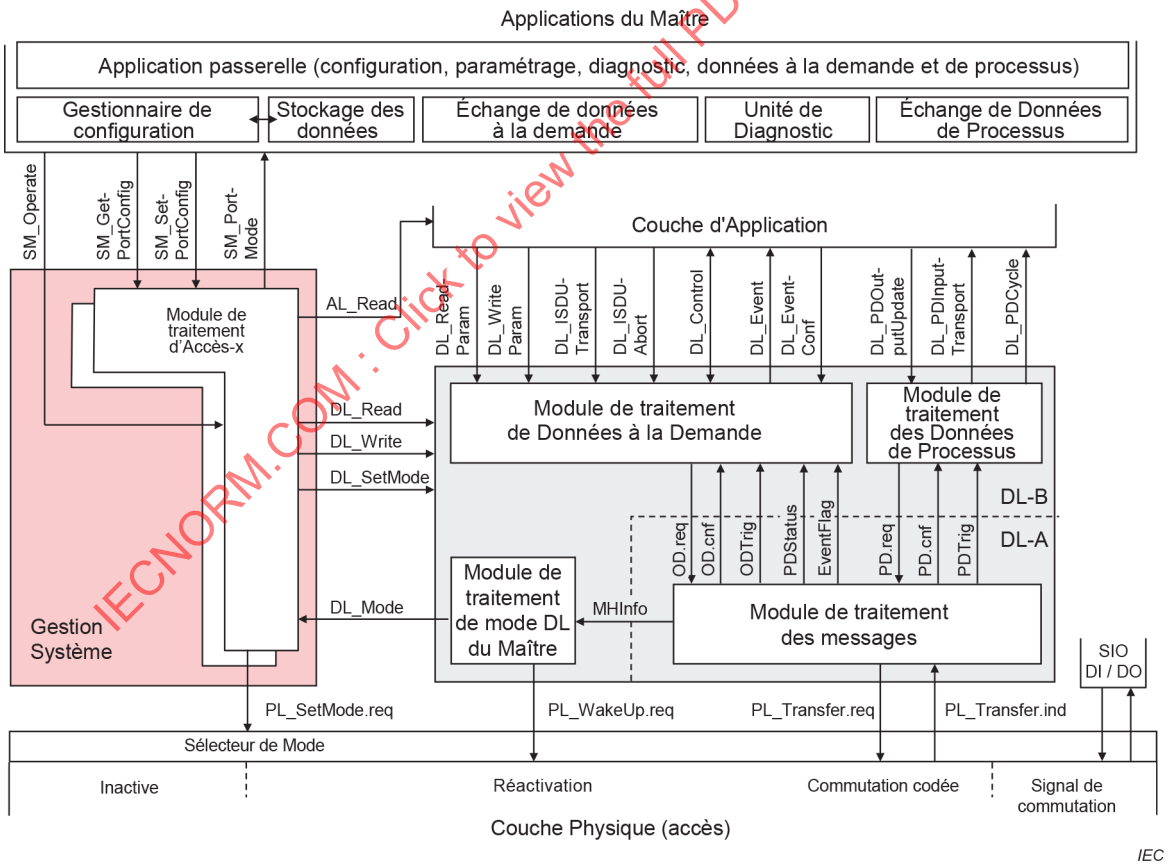


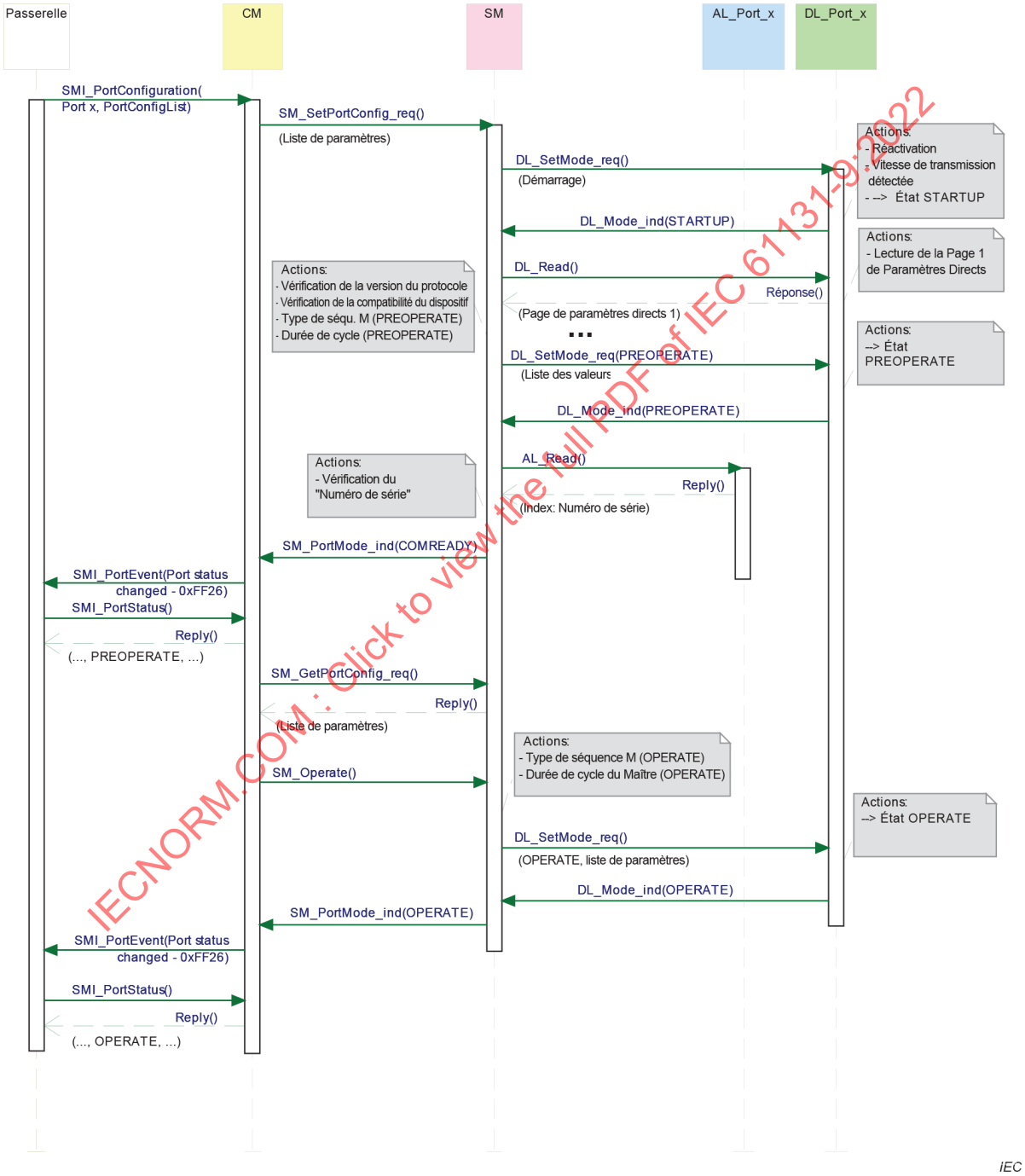
Figure 69 – Structure et services de la Gestion Système du Maître

La Figure 70 représente les actions entre les couches d'application du Maître (App Maître), de Gestion de la Configuration (CM), de Gestion Système (SM), de liaison des données (DL) et la couche d'application (AL) en cas d'utilisation en démarrage d'un Accès particulier.

Ce cas d'utilisation particulier est caractérisé par les postulats suivants:

– le Dispositif pour la configuration disponible est connecté et la vérification est réussie;
– le Dispositif utilise la version de protocole correcte conformément au présent document;
– l'InspectionLevel configuré est "type compatible" (Le SerialNumber est lu à partir du Dispositif et non vérifié).

Les flèches en pointillés de la Figure 70 représentent les services de réponse à un service initial.



IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Figure 70 – Organigramme séquentiel du cas d'utilisation "Configuration d'Accès x"

9.2.2 Services SM du Maître

9.2.2.1 Vue d'ensemble

La Gestion Système fournit à l'utilisateur les services SM du Maître par l'intermédiaire de son interface supérieure. Le Tableau 78 énumère les attributions du Maître quant à son rôle d'initiateur ou de destinataire des services individuels de la Gestion Système.

Tableau 78 – Services SM au sein du Maître

Désignation du service	Maître
SM_SetPortConfig	R
SM_GetPortConfig	R
SM_PortMode	I
SM_Operate	R
Légende (voir 3.3.4) I Initiateur du service R Destinataire (Répondeur) du service	

9.2.2.2 SM_SetPortConfig

Le service SM_SetPortConfig est utilisé pour définir la configuration demandée du Dispositif. Les paramètres des primitives de service sont énumérés dans le Tableau 79.

Tableau 79 – SM_SetPortConfig

Nom du paramètre	.req	.cnf
Argument	M	
ParameterList	M	
Résultat (+)		S
Numéro d'Accès		M
Résultat (−)		S
Numéro d'Accès		M
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

ParameterList

Ce paramètre contient l'Accès configuré et les paramètres de Dispositif d'un Accès du Maître.

Type de paramètre: Enregistrement.

Enregistrement d'éléments:

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

ConfiguredCycleTime

Ce paramètre contient la durée de cycle demandée pour le mode OPERATE.

Valeurs admises:
0 (FreeRunning),
Temps (voir le Tableau B.3).

TargetMode

Ce paramètre indique le mode de fonctionnement demandé de l'Accès.

Valeurs admises: INACTIVE, DI, DO, CFGCOM, AUTOCOM (voir le Tableau 81).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ConfiguredRevisionID (CRID):

Longueur de données: 1 octet pour la version de protocole (voir B.1.5).

InspectionLevel:

Valeurs admises: NO_CHECK, TYPE_COMP, IDENTICAL (voir le Tableau 80).

ConfiguredVendorID (CVID)

Longueur de données: 2 octets.

NOTE Les VendorID (identifiants de fournisseurs) sont attribués par le "IO-Link community".

ConfiguredDeviceID (CDID)

Longueur de données: 3 octets.

ConfiguredFunctionID (CFID)

Longueur de données: 2 octets.

ConfiguredSerialNumber (CSN)

Longueur de données: jusqu'à 16 octets (voir le Tableau 80).

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
PARAMETER_CONFLICT (cohérence du jeu de paramètres non respectée).

Le Tableau 80 spécifie le codage des différents niveaux de vérification (valeurs du paramètre InspectionLevel). Voir 9.2.3.2 et 11.3.2.

Tableau 80 – Définition de InspectionLevel (IL)

Paramètre	Niveau de vérification (IL)		
	NO_CHECK	TYPE_COMP	IDENTICAL[a]
DeviceID (DID) (compatible)	–	Oui (RDID = CDID)	Oui (RDID = CDID)
VendorID (VID)	–	Oui (RVID = CVID)	Oui (RVID = CVID)
SerialNumber (SN)	–	–	Oui (RSN = CSN)
[a] "IDENTICAL" = facultatif (non recommandé pour les nouveaux développements).			

Le Tableau 81 spécifie le codage des différents Modes Cibles.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 81 – Définitions des Modes Cibles

Mode Cible	Définition
CFGCOM	Dispositif communiquant en mode CFGCOM après inspection réussie
AUTOCOM	Dispositif communiquant en mode AUTOCOM sans inspection
INACTIVE	Communication désactivée, pas de DI, pas de DO
DI	Accès en mode d'entrée numérique (SIO)
DO	Accès en mode de sortie numérique (SIO)

CFGCOM est un Mode Cible fondé sur une configuration utilisateur (par exemple, au moyen d'un IODD) et sur un contrôle de cohérence de RID, VID et DID.

AUTOCOM est un Mode Cible sans configuration. Ceci signifie qu'il n'y a aucun contrôle de CVID et de CDID. Le CRID est établi sur la version la plus élevée prise en charge par le Maître. Il convient qu'AUTOCOM ne puisse être sélectionné qu'avec le niveau de vérification "NO_CHECK" (voir le Tableau 80).

9.2.2.3 SM_GetPortConfig

Le service SM_GetPortConfig est utilisé pour acquérir la configuration courante (réelle) du Dispositif. Les paramètres des primitives de service sont énumérés dans le Tableau 82.

Tableau 82 – SM_GetPortConfig

Nom du paramètre	.req	.cnf
Argument	M	
Numéro d'Accès	M	
Résultat (+)		S(=)
Parameterlist		M
Résultat (–)		S(=)
Numéro d'Accès		M
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ParameterList

Ce paramètre contient l'Accès et le paramètre de Dispositif configurés d'un Accès du Maître.

Type de paramètre: Enregistrement.

Enregistrement d'éléments:

PortNumber

Ce paramètre contient le numéro d'Accès.

TargetMode

Ce paramètre indique le mode de fonctionnement.

Valeurs admises: INACTIVE, DI, DO, CFGCOM, AUTOCOM (voir le Tableau 81).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

RealBaudrate

Ce paramètre indique la vitesse de transmission réelle

Valeurs admises:
COM1 (vitesse de transmission de COM1),
COM2 (vitesse de transmission de COM2),
COM3 (vitesse de transmission de COM3).

RealCycleTime

Ce paramètre contient la durée de cycle courant (réelle).

RealRevision (RRID)

Longueur de données: 1 octet pour la version de protocole (voir B.1.5).

RealVendorID (RVID)

Longueur de données: 2 octets.

NOTE Les VendorID (identifiants de fournisseurs) sont attribués par le "IO-Link community".

RealDeviceID (RDID)

Longueur de données: 3 octets.

RealFunctionID (RFID)

Longueur de données: 2 octets.

RealSerialNumber (RSN)

Longueur de données: jusqu'à 16 octets.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
PARAMETER_CONFLICT (cohérence du jeu de paramètres non respectée).

Tous les paramètres doivent être réglés sur "0" s'il n'y a pas d'information disponible.

9.2.2.4 SM_PortMode

Le service SM_PortMode est utilisé pour indiquer des changements ou des défauts du mode de communication local. Ceux-ci doivent être notifiés à l'application du Maître. Les paramètres des primitives de service sont énumérés dans le Tableau 83.

Tableau 83 – SM_PortMode

Nom du paramètre	.ind
Argument	M
Numéro d'Accès	M
Mode	M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Mode

Valeurs admises:
INACTIVE (Communication désactivée, pas de DI, pas de DO),
DI (Accès en mode d'entrée numérique (SIO)),
DO (Accès en mode de sortie numérique (SIO)),
COMREADY (Communication établie et inspection réussie),
SM_OPERATE (L'Accès est prêt à échanger des Données de Processus),
COMLOST (Échec de la communication, nouvelle procédure de réactivation exigée),
REVISION_FAULT (Version de protocole incompatible),
COMP_FAULT (Dispositif incompatible ou Dispositif hérité selon l'InspectionLevel),
SERNUM_FAULT (SerialNumber non concordant selon l'InspectionLevel),
CYCTIME_FAULT (Le Dispositif ne prend pas en charge la durée de cycle configurée).

9.2.2.5 SM_Operate

Le service SM_Operate sollicite la Gestion Système pour calculer le MasterCycleTime des Accès si le service fait l'objet d'un acquittement positif de Résultat (+). Ce service est en vigueur au niveau de l'Accès indiqué. Les paramètres des primitives de service sont énumérés dans le Tableau 84.

Tableau 84 – SM_Operate

Nom du paramètre	.req	.cnf
Argument	M	
Numéro d'Accès	M	
Résultat (+)		S
Résultat (−)		S
Numéro d'Accès		M
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

Numéro d'Accès

Ce paramètre contient le numéro d'Accès.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (service indisponible à l'état actuel, par exemple si l'Accès est déjà à l'état OPERATE).

9.2.3 Protocole SM du Maître

9.2.3.1 Vue d'ensemble

Du fait des caractéristiques exhaustives de la configuration, du paramétrage et des fonctionnalités opérationnelles du SDCI, la description du comportement à l'aide de diagrammes d'états devient quelque peu complexe. Comme pour les diagrammes d'états de la

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

couche DL, le 9.2.3 utilise la possibilité de disposer de sous-machines dans les diagrammes d'états principaux.

Les méthodes globales de vérification de la compatibilité sont réalisées au sein des états des sous-machines. Ces méthodes sont indiquées par les champs "*Appliquer méthode*" dans les graphes d'états, par exemple à la Figure 72.

La logique décisionnelle correspondante est représentée par des diagrammes d'activités (voir la Figure 73, la Figure 74, la Figure 75 et la Figure 78).

9.2.3.2 Diagramme d'états de la SM du Maître

La Figure 71 représente le diagramme d'états principal de la Gestion Système du Maître.

Deux sous-machines pour la vérification de la compatibilité et du numéro de série sont décrites en 9.2.3.3 et en 9.2.3.4.

En cas d'interruption de la communication, la Gestion Système est informée au moyen du service DL_Mode (COMLOST).

Seul le service SM_SetPortConfig permet de reconfigurer un Accès.

Le service SM_Operate n'a d'effet dans aucun état, sauf dans l'état "wait_4".

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

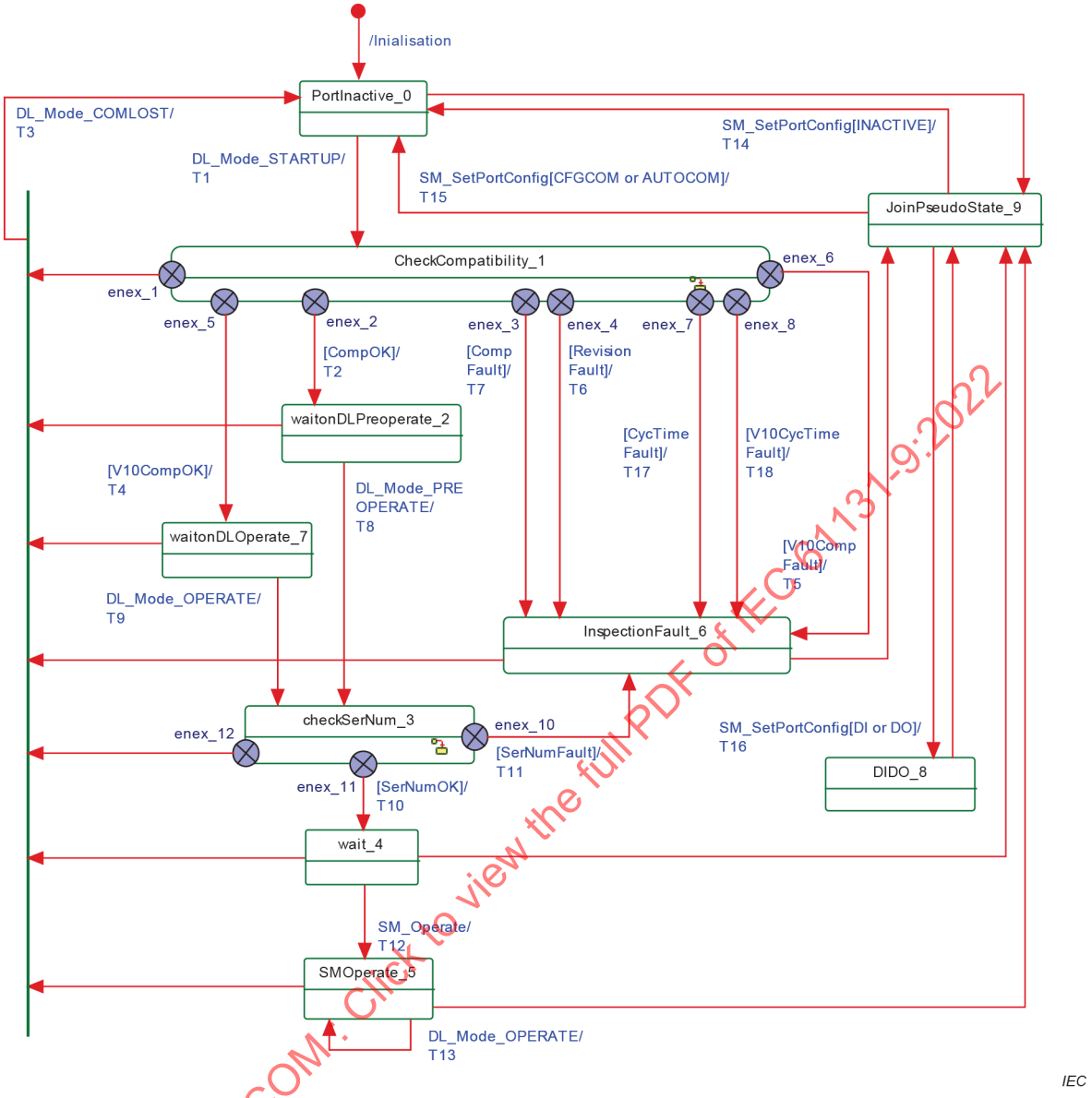


Figure 71 – Diagramme d'états principal de la Gestion Système du Maître

Le Tableau 85 présente la table de transitions d'états de la Gestion Système du Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 85 – Table de transitions d'états de la Gestion Système du Maître

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
PortInactive_0	Pas de communication
CheckCompatibility_1	L'Accès démarre et la version ainsi que la compatibilité du Dispositif sont vérifiées. Voir la Figure 72.
waitonDLPreoperate_2	Attendre jusqu'à ce que l'état PREOPERATE soit établi et que tous les modules de traitement des Données à la Demande aient démarré. L'Accès est prêt pour la communication.
checkSerNum_3	Le numéro de série est vérifié en fonction du niveau de vérification (IL). Voir la Figure 77.
wait_4	L'Accès est prêt à communiquer et attend le service SM_Operate de la CM.
SM Operate_5	L'Accès est à l'état OPERATE et effectue l'échange cyclique de données de processus.
InspectionFault_6	L'Accès est prêt pour la communication. Cependant, l'échange cyclique des Données de Processus ne peut pas être effectué à cause d'incompatibilités.
waitonDLOperate_7	Attente de l'état OPERATE demandé dans le cas où le Maître est connecté à un Dispositif hérité. Le Numéro de série peut être lu par la suite.
DIDO_8	L'Accès passe en mode DI ou DO (SIO, pas de communication)
JoinPseudoState_9	Ce pseudo-état est utilisé à la place d'un lien de jointure UML. Il permet l'exécution de services individuels SM_SetPortConfig en fonction de l'état du système (INACTIVE, CFGCOM, AUTOCOM, DI, ou DO)

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	CompRetry = 0
T2	1	2	DL_SetMode.req (PREOPERATE, ValueList)
T3	1,2,3,4,5,6,7	0	DL_SetMode.req (INACTIVE) et SM_Mode.ind (COMLOST) due à un défaut de communication
T4	1	7	DL_SetMode.req (OPERATE, ValueList)
T5	1	6	SM_PortMode.ind (COMP_FAULT), DL_SetMode.req (OPERATE, ValueList)
T6	1	6	SM_PortMode.ind (REVISION_FAULT), DL_SetMode.req (PREOPERATE, ValueList)
T7	1	6	SM_PortMode.ind (COMP_FAULT), DL_SetMode.req (PREOPERATE, ValueList)
T8	2	3	-
T9	7	3	-
T10	3	4	SM_PortMode.ind (COMREADY)
T11	3	6	SM_PortMode.ind (SERNUM_FAULT)
T12	4	5	DL_SetMode.req (OPERATE, ValueList)
T13	5	5	-
T14	0,4,5,6,8	0	SM_PortMode.ind (INACTIVE), DL_SetMode.req (INACTIVE)
T15	0,4,5,6,8	0	DL_SetMode.req (STARTUP, ValueList), PL_SetMode.req (SDCI)
T16	0,4,5,6,8	8	PL_SetMode.req (SIO), SM_Mode.ind (DI ou DO), DL_SetMode.req (INACTIVE)
T17	1	6	SM_PortMode.ind (CYCTIME_FAULT), DL_SetMode.req (PREOPERATE, ValueList)
T18	1	6	SM_PortMode.ind (CYCTIME_FAULT), DL_SetMode.req (OPERATE, ValueList), ValueList.M-sequenceTime = MinCycleTime du Dispositif

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
CompOK	Bool	Voir la Figure 75
CompFault	Bool	Voir la Figure 75; Variable d'erreur COMP_FAULT

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
CycTimeFault	Bool	Voir la Figure 75; Variable d'erreur CYCTIME_FAULT
RevisionFault	Bool	Voir la Figure 73; Variable d'erreur REVISION_FAULT
SerNumFault	Bool	Voir la Figure 78; Variable d'erreur SERNUM_FAULT
SerNumOK	Bool	Voir la Figure 78
V10CompFault	Bool	Voir la Figure 74; Variable d'erreur COMP_FAULT
V10CompOK	Bool	Voir la Figure 74
V10CycTimeFault	Bool	Voir la Figure 74; Variable d'erreur CYCTIME_FAULT
INACTIVE	Variable	Un Mode Cible dans le service SM_SetPortConfig
CFGCOM, AUTOCOM	Variables	Des Modes Cibles dans le service SM_SetPortConfig

9.2.3.3 Sous-machine de la SM du Maître "Vérification de la compatibilité"

La Figure 72 représente la sous-machine de la SM du Maître checkCompatibility_1.

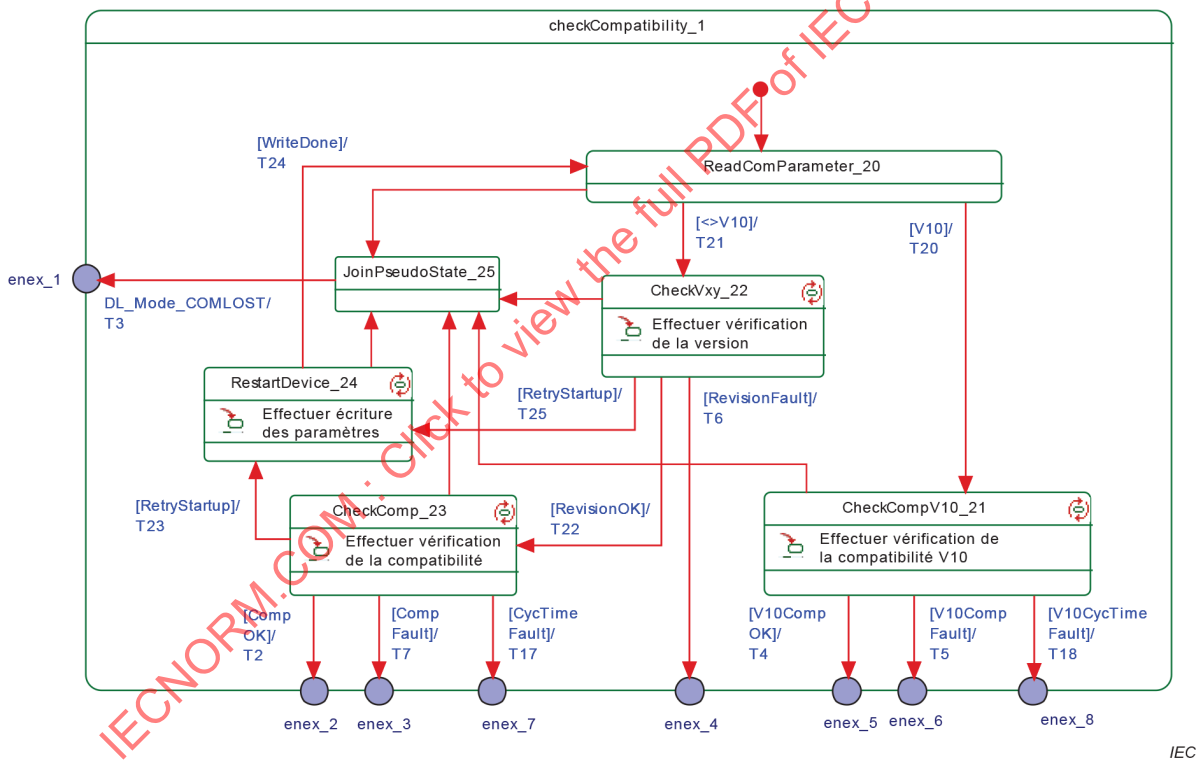


Figure 72 – Sous-machine de la SM du Maître CheckCompatibility_1

Le Tableau 86 présente la table de transitions d'états de la sous-machine du Maître checkCompatibility_1.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 86 – Table de transitions d'états de la sous-machine du Maître CheckCompatibility_1

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
ReadComParameter_20	Acquisition des paramètres de communication de la page 1 de Paramètres Directs (0x02 à 0x06) au moyen du service DL_Read (voir le Tableau B.1).
CheckCompV10_21	Acquisition des paramètres d'identification de la page 1 de Paramètres Directs (0x07 à 0x0D) au moyen du service DL_Read (voir le Tableau B.1). Le niveau de vérification (IL) configuré définit la logique décisionnelle de la vérification de compatibilité ultérieure "CheckCompV10" avec les paramètres RVID, RDID et RFID conformément à la Figure 74.
CheckVxy_22	Un contrôle est réalisé afin de vérifier que la version (CRID) configurée correspond à la version (RRID) en cours (réelle) conformément à la Figure 73.
CheckComp_23	Acquisition des paramètres d'identification de la page 1 de Paramètres Directs (0x07 à 0x0D) au moyen du service DL_Read (voir le Tableau B.1). Le niveau de vérification (IL) configuré définit la logique décisionnelle de la vérification de compatibilité ultérieure "CheckComp" conformément à la Figure 75.
RestartDevice_24	Écriture des paramètres de compatibilité de la version de protocole configurée (CRID) et de l'identifiant de Dispositif configuré (CDID) dans le Dispositif en fonction du Mode Cible de communication CFGCOM ou AUTOCOM (voir le Tableau 81) conformément à la Figure 76.
JoinPseudoState_25	Ce pseudo-état est utilisé à la place d'un lien de jointure UML. Pas de protections.

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T20	20	21	–
T21	20	22	DL_Write (0x00, MCmd_MASTERIDENT), voir le Tableau B.2
T22	22	23	–
T23	23	24	–
T24	24	20	–
T25	22	24	CompRetry = CompRetry +1

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
CompOK	Bool	Voir la Figure 75
CompFault	Bool	Voir la Figure 75; Variable d'erreur COMP_FAULT
RevisionFault	Bool	Voir la Figure 73; Variable d'erreur REVISION_FAULT
RevisionOK	Bool	Voir la Figure 73
SerNumFault	Bool	Voir la Figure 78; Variable d'erreur SERNUM_FAULT
SerNumOK	Bool	Voir la Figure 78
V10	Bool	La version de protocole réelle du Dispositif connecté est une version héritée (V1.0, voir B.1.5)
<>V10	Bool	La version de protocole réelle du Dispositif connecté est conforme au présent document
V10CompFault	Bool	Voir la Figure 74; Variable d'erreur COMP_FAULT
V10CompOK	Bool	Voir la Figure 74
RetryStartup	Bool	Voir la Figure 73 et la Figure 75
CompRetry	Variable	Compteur interne
WriteDone	Bool	Finalisation de la séquence du service de redémarrage
MCmd_XXXXXXX	Invocation	Voir le Tableau 45

Certains états comportent des logiques complexes permettant de traiter les vérifications de compatibilité et de validité. Les figures de la Figure 73 à la Figure 76 représentent ce contexte.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

La Figure 73 représente la logique décisionnelle de la vérification de la version de protocole dans l'état "CheckVxy". Dans le cas de Dispositifs configurés, la règle suivante s'applique: si la version configurée (CRID) et la version réelle (RRID) ne correspondent pas, le CRID est transmis au Dispositif. Si le Dispositif ne l'accepte pas, le Maître renvoie une indication au moyen du service SM_Mode avec REV_FAULT.

Dans le cas de Dispositifs non configurés, le mode de fonctionnement AUTOCOM doit être utilisé. Pour les abréviations de noms de paramètres, voir 9.2.2.2 et 9.2.2.3.

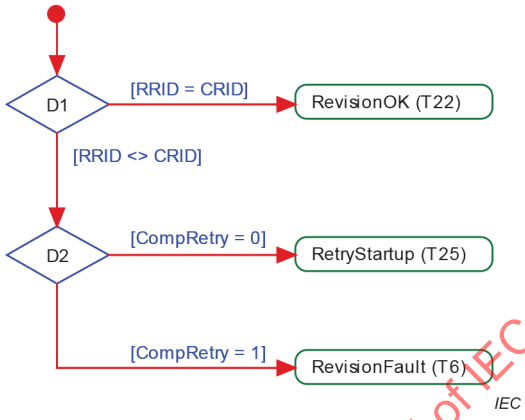
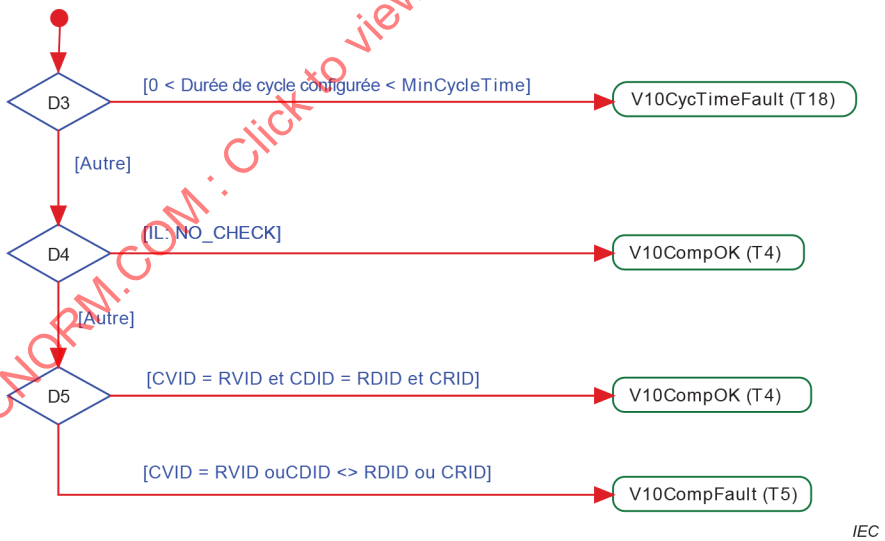


Figure 73 – Activité pour l'état "CheckVxy"

La Figure 74 représente la logique décisionnelle pour la vérification de compatibilité de versions héritées dans l'état "CheckCompV10".



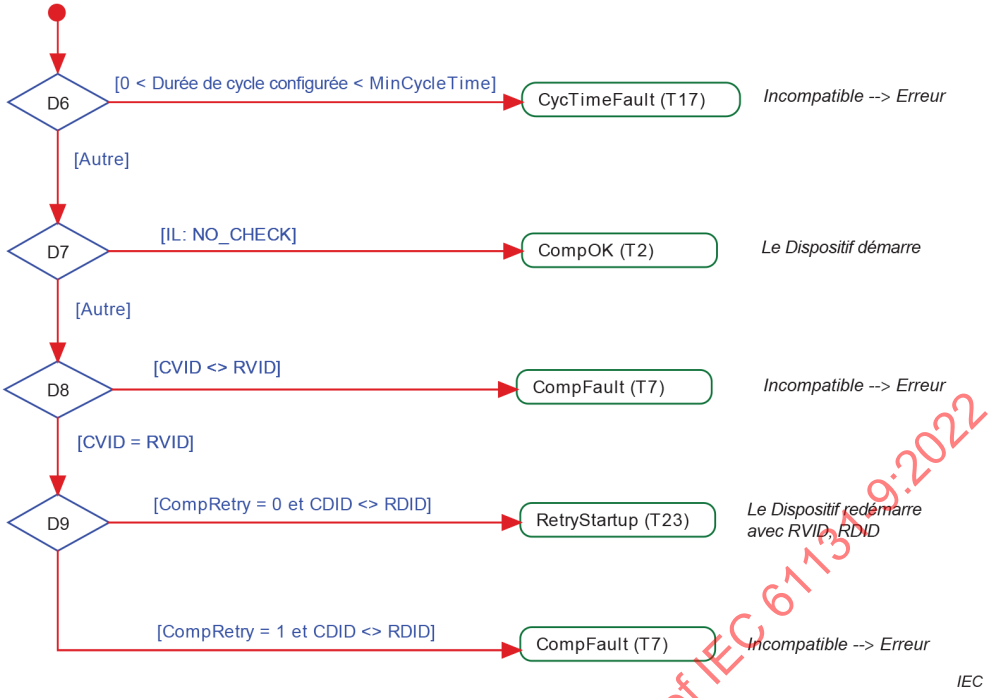
Légende

IL = Niveau de vérification

Figure 74 – Activité pour l'état "CheckCompV10"

La Figure 75 représente la logique décisionnelle pour la vérification de compatibilité dans l'état "CheckComp".

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022



Légende

IL = Niveau de vérification

Figure 75 – Activité pour l'état "CheckComp"

La Figure 76 représente l'activité (écriture de paramètres) dans l'état "RestartDevice".

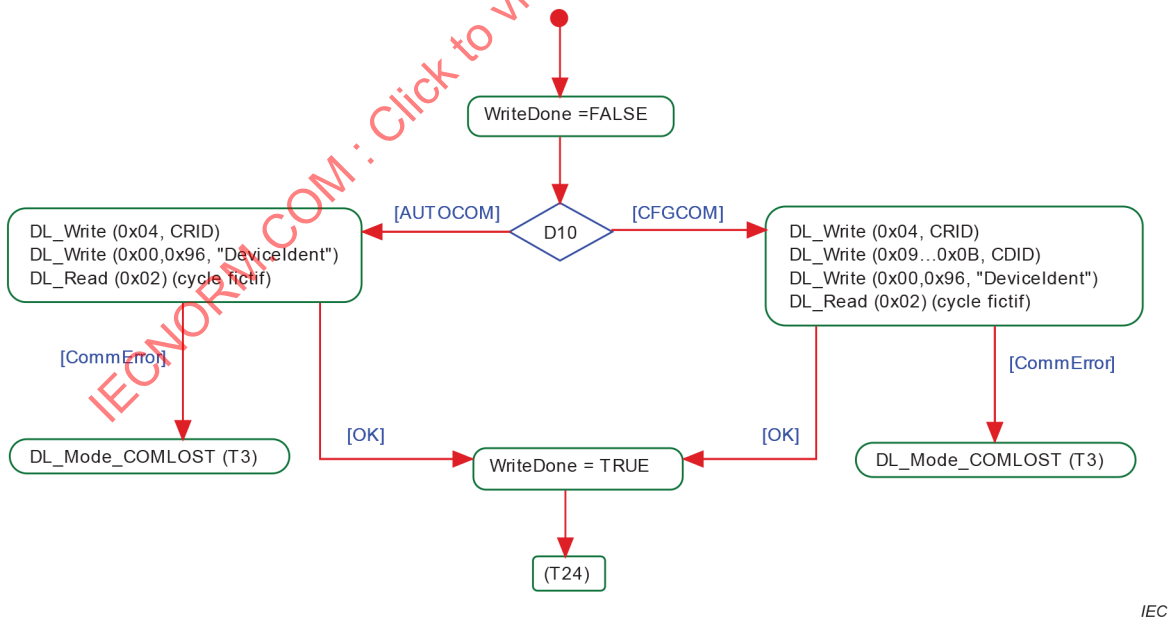


Figure 76 – Activité (écriture de paramètres) dans l'état "RestartDevice"

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

9.2.3.4 Sous-machine de la SM du Maître "Vérification du numéro de série"

La Figure 77 représente la sous-machine de la SM du Maître "checkSerNum_3". L'état CheckSerNum_31 peut être ignoré (option).

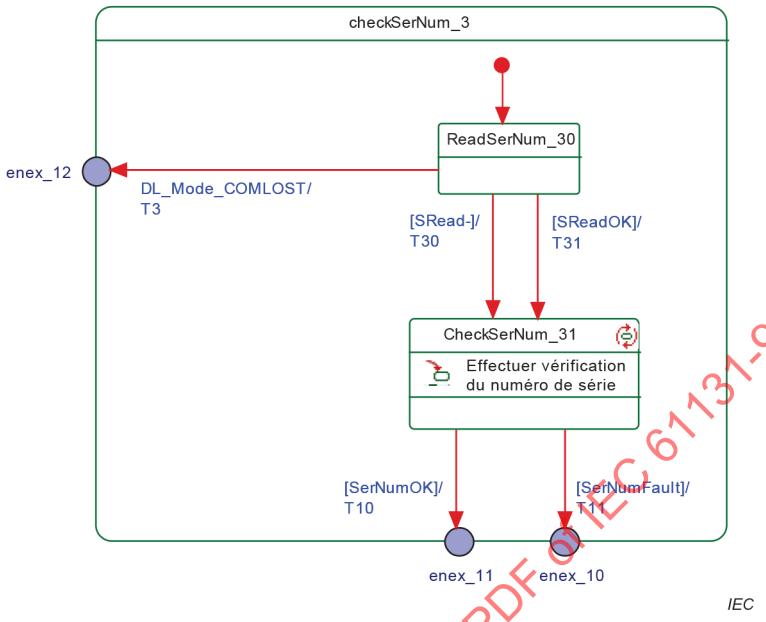


Figure 77 – Sous-machine de la SM du Maître checkSerNum_3

Le Tableau 87 présente la table de transitions d'états de la sous-machine du Maître checkSerNum_3.

Tableau 87 – Table de transitions d'états de la sous-machine du Maître checkSerNum_3

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
ReadSerNum_30		Acquisition du Numéro de série en provenance du Dispositif par l'intermédiaire de AL_Read.req (Index: 0x0015). Une réponse positive (AL_Read(+)) donne lieu à SReadOK = vrai. Une réponse négative (AL_Read(-)) donne lieu à SRead- = vrai.	
CheckSerNum_31		Facultatif: Vérification de SerialNumber ignorée ou vérifiée correctement.	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T30	40	41	–
T31	40	41	–
ÉLÉMENTS INTERNES		**TYPE**	**DÉFINITION**
SRead-		Bool	Réponse négative du service AL_Read (Index 0x0015)
SReadOK		Bool	Lecture de Numéro de série correcte
SerNumOK		Bool	Voir la Figure 78
SerNumFault		Bool	Voir la Figure 78

La Figure 78 représente la logique décisionnelle (activité) pour l'état CheckSerNum_31.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

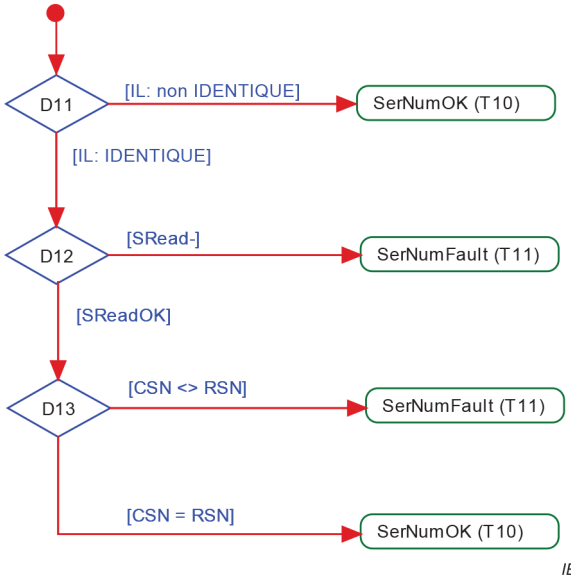


Figure 78 – Activité (vérification du SerialNumber) pour l'état CheckSerNum_31

9.2.3.5 Règles d'utilisation des types de séquence M

La Gestion Système est chargée de configurer les types de séquences M corrects. Cette configuration a lieu après les actions de vérification de compatibilité (transition vers PREOPERATE) et avant le passage à OPERATE.

Différents types de séquences M doivent être utilisés dans les différents états opérationnels (voir A.2.6). Par exemple, lors de la commutation à l'état OPERATE, le type de séquence M pertinent pour le fonctionnement cyclique doit être utilisé. Le type de séquence M à utiliser à l'état opérationnel OPERATE est déterminé par la taille des Données de Processus d'entrée et de sortie. Les types de séquences M disponibles pour les trois modes STARTUP, PREOPERATE et OPERATE, ainsi que le codage correspondant du paramètre M-sequenceCapability sont spécifiés en A.2.6. Les formats de données d'entrée et de sortie doivent être obtenus à partir du Dispositif connecté afin d'ajuster le type de séquence M. Il est obligatoire qu'un Maître mette en œuvre tous les types de séquences M spécifiés en A.2.6.

9.3 Gestion Système du Dispositif

9.3.1 Vue d'ensemble

La Figure 79 donne une vue d'ensemble de la structure et des services de la Gestion Système du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

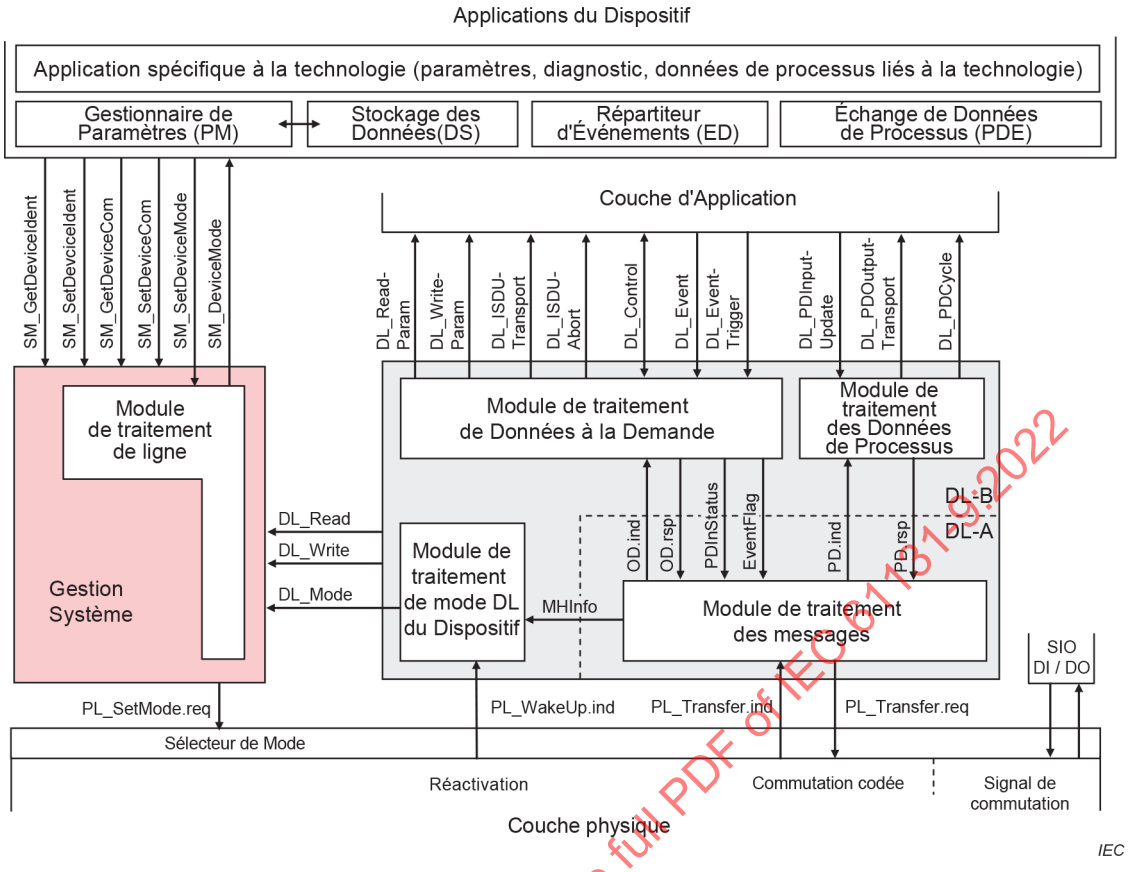


Figure 79 – Structure et services de la Gestion Système (Dispositif)

La Gestion Système (SM) du Dispositif fournit l'instance de contrôle centralisé par l'intermédiaire du Module de traitement de ligne pendant toutes les phases d'initialisation, d'état par défaut (SIO), de démarrage de la communication, de communication et de repli au mode SIO.

La SM du Dispositif interagit avec la couche PL afin d'établir les réglages nécessaires du circuit d'attaque de la ligne et du récepteur (voir la Figure 16), en s'appuyant sur la couche DL pour obtenir les informations nécessaires en provenance du Maître (réactivation, vitesses de transmission, etc.), ainsi que sur les applications du Dispositif pour en vérifier l'identité et la compatibilité (paramètres d'identification).

Les transitions entre états du Module de traitement de Ligne (voir la Figure 81) sont initiées par les activités de l'Accès du Maître (réactivation et communication) et déclenchées par la Couche de Liaison de Données du Dispositif au moyen des indications DL_Mode et des demandes DL_Write (commandes).

La SM fournit les paramètres d'identification du Dispositif par l'intermédiaire de l'interface d'applications du Dispositif.

L'organigramme séquentiel de la Figure 80 représente une séquence de Dispositif type, depuis l'initialisation au mode SIO par défaut, en passant par la demande de réactivation en provenance du Maître jusqu'à la communication finale. L'organigramme séquentiel est complété par le cas d'utilisation d'une erreur de communication, telle que l'expiration de T_{DSIO}, un défaut de communication ou une demande de Maître, par exemple un repli (provoqué par un Événement).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

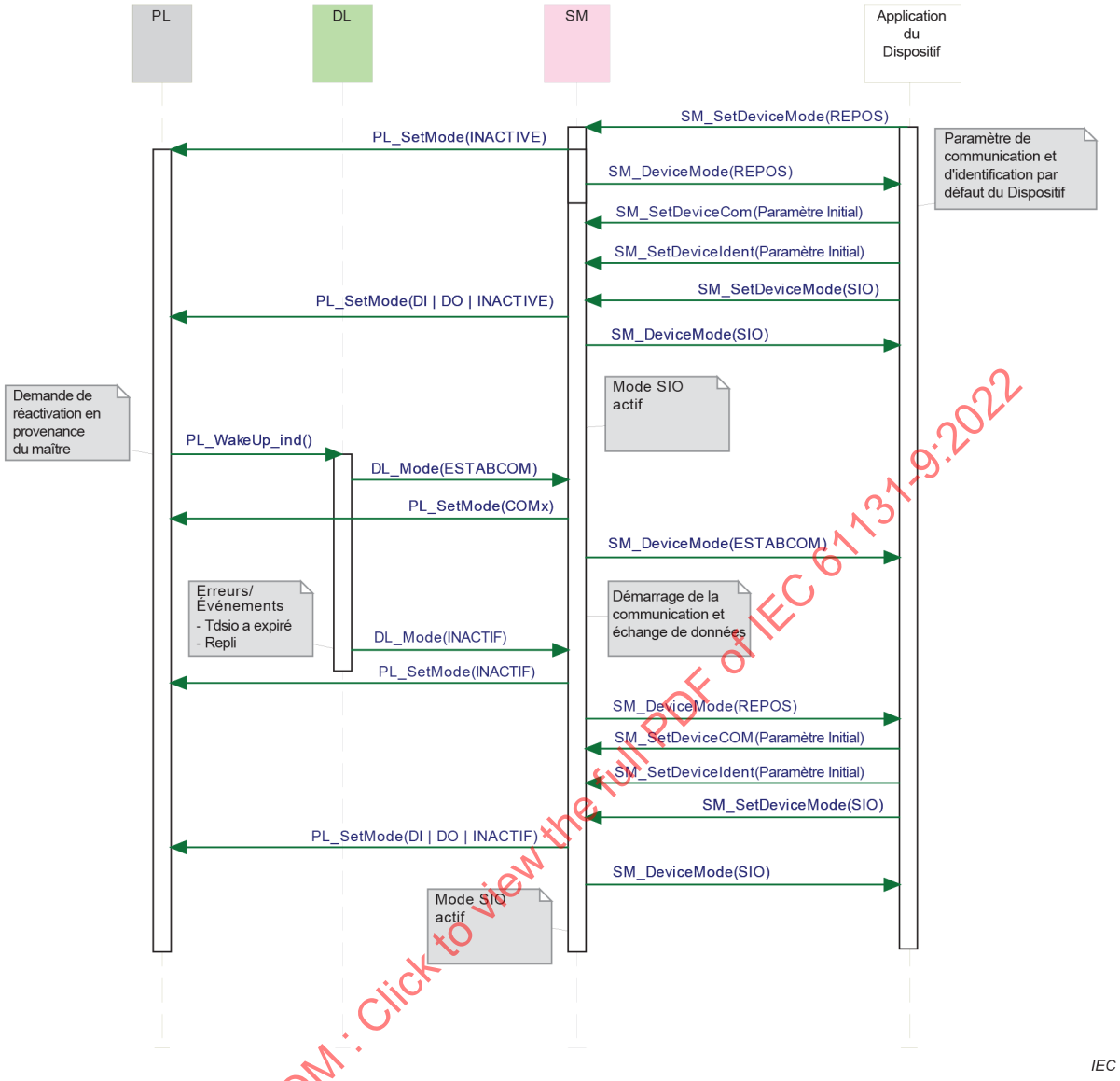


Figure 80 – Organigramme séquentiel du cas d'utilisation "INACTIVE – SIO – SDCI – SIO"

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Les services SM représentés à la Figure 80 sont définis au 9.3.2.

9.3.2 Services SM du Dispositif

9.3.2.1 Vue d'ensemble

Le paragraphe 9.3.2 décrit les services que la Gestion Système du Dispositif fournit à ses applications, comme cela est représenté à la Figure 79.

Le Tableau 88 énumère les attributions du Dispositif quant à son rôle d'initiateur ou de destinataire du service individuel de la Gestion Système.

Tableau 88 – Services SM au sein du Dispositif

Désignation du service	Dispositif
SM_SetDeviceCom	R
SM_GetDeviceCom	R
SM_SetDeviceIdent	R
SM_GetDeviceIdent	R
SM_SetDeviceMode	R
SM_DeviceMode	I
Légende (voir 3.3.4) I Initiateur du service R Destinataire (Répondeur) du service	

9.3.2.2 SM_SetDeviceCom

Le service SM_SetDeviceCom est utilisé pour configurer les propriétés de communication prises en charge par le Dispositif dans la Gestion Système. Les paramètres des primitives de service sont énumérés dans le Tableau 89.

Tableau 89 – SM_SetDeviceCom

Nom du paramètre	.req	.cnf
Argument ParameterList	M M	
Résultat (+)		S
Résultat (−) ErrorInfo		S M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

ParameterList

Ce paramètre contient les paramètres de communication configurés pour un Dispositif donné.

Type de paramètre: Enregistrement.

Enregistrement d'éléments:

SupportedSIOMode

Ce paramètre indique le mode SIO pris en charge par le Dispositif.

Valeurs admises:
INACTIVE (impédance haute de la ligne C/Q),
DI (ligne C/Q en mode d'entrée numérique),
DO (ligne C/Q en mode de sortie numérique).

SupportedTransmissionrate

Ce paramètre indique la vitesse de transmission prise en charge par le Dispositif.

Valeurs admises:
COM1 (vitesse de transmission de COM1),
COM2 (vitesse de transmission de COM2),
COM3 (vitesse de transmission de COM3).

MinCycleTime

Ce paramètre contient la durée de cycle minimale prise en charge par le Dispositif (voir B.1.3).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Capacité de séquence M

Ce paramètre indique les capacités prises en charge par le Dispositif (voir B.1.4):
– Prise en charge ISDU,
– Types de séquences M OPERATE,
– Types de séquences M PREOPERATE.

RevisionID (RID)

Ce paramètre contient la version du protocole (voir B.1.5) prise en charge par le Dispositif.

ProcessDataIn

Ce paramètre contient la longueur des PD à envoyer au Maître (voir B.1.6).

ProcessDataOut

Ce paramètre contient la longueur des PD à envoyer par le Maître (voir B.1.7).

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
PARAMETER_CONFLICT (cohérence du jeu de paramètres non respectée).

9.3.2.3 SM_GetDeviceCom

Le service SM_GetDeviceCom est utilisé pour lire les propriétés de communication courantes à partir de la Gestion Système. Les paramètres des primitives de service sont énumérés dans le Tableau 90.

Tableau 90 – SM_GetDeviceCom

Nom du paramètre	.req	.cnf
Argument	M	
Résultat (+)		S
ParameterList		M
Résultat (−)		S
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

ParameterList

Ce paramètre contient le paramètre de communication configuré pour un Dispositif.

Type de paramètre: Enregistrement.

Enregistrement d'éléments:

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

CurrentMode

Ce paramètre indique le mode SIO ou le mode de communication courant du Dispositif.

Valeurs admises:
INACTIVE (impédance haute de la ligne C/Q),
DI (ligne C/Q en mode d'entrée numérique),
DO (ligne C/Q en mode de sortie numérique),
COM1 (vitesse de transmission de COM1),
COM2 (vitesse de transmission de COM2),
COM3 (vitesse de transmission de COM3).

MasterCycleTime

Ce paramètre contient le MasterCycleTime à établir par la Gestion Système du Maître (voir B.1.3). Ce paramètre est applicable seulement à l'état SM_Operate.

Capacité de séquence M

Ce paramètre indique les capacités de la séquence M courante configurée dans la Gestion Système du Dispositif (voir B.1.4):
– prise en charge ISDU;
– types de séquences M OPERATE;
– types de séquences M PREOPERATE.

RevisionID (RID)

Ce paramètre contient la version courante du protocole (voir B.1.5) dans la Gestion Système du Dispositif.

ProcessDataIn

Ce paramètre contient la longueur courante des PD à envoyer au Maître (voir B.1.6).

ProcessDataOut

Ce paramètre contient la longueur courante des PD à envoyer par le Maître (voir B.1.7).

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (Service non disponible en l'état courant).

9.3.2.4 SM_SetDeviceIdent

Le service SM_SetDeviceIdent est utilisé pour configurer les données d'identification du Dispositif dans la Gestion Système. Les paramètres des primitives de service sont énumérés dans le Tableau 91.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 91 – SM_SetDeviceIdent

Nom du paramètre	.req	.cnf
Argument ParameterList	M M	
Résultat (+)		S
Résultat (−) ErrorInfo		S M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

ParameterList

Ce paramètre contient le paramètre d'identification configuré pour un Dispositif.

Type de paramètre: Enregistrement.

Enregistrement d'éléments:

VendorID (VID)

Ce paramètre contient le VendorID attribué à un Dispositif (voir B.1.8).

Longueur de données: 2 octets.

DeviceID (DID)

Ce paramètre contient un des DeviceID attribués (voir B.1.9).

Longueur de données: 3 octets.

FunctionID (FID)

Ce paramètre contient un des FunctionID attribués (voir B.1.10).

Longueur de données: 2 octets.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (service non disponible en l'état courant),
PARAMETER_CONFLICT (cohérence du jeu de paramètres non respectée).

9.3.2.5 SM_GetDeviceIdent

Le service SM_GetDeviceIdent est utilisé pour lire le paramètre d'identification du Dispositif à partir de la Gestion Système. Les paramètres des primitives de service sont énumérés dans le Tableau 92.

Tableau 92 – SM_GetDeviceIdent

Nom du paramètre	.req	.cnf
Argument	M	
Résultat (+)		S
ParameterList		M
Résultat (−)		S
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

ParameterList

Ce paramètre contient les paramètres de communication configurés du Dispositif.

Type de paramètre: Enregistrement.

Enregistrement d'éléments:

VendorID (VID)

Ce paramètre contient le VendorID réel du Dispositif (voir B.1.8).

Longueur de données: 2 octets.

DeviceID (DID)

Ce paramètre contient le DeviceID réel du Dispositif (voir B.1.9).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Longueur de données: 3 octets.

FunctionID (FID)

Ce paramètre contient le FunctionID réel du Dispositif (voir B.1.10).

Longueur de données: 2 octets.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (service non disponible en l'état courant).

9.3.2.6 SM_SetDeviceMode

Le service SM_SetDeviceMode est utilisé pour mettre le Dispositif dans un état opérationnel défini lors de l'initialisation. Les paramètres des primitives de service sont énumérés dans le Tableau 93.

Tableau 93 – SM_SetDeviceMode

Nom du paramètre	.req	.cnf
Argument	M	
Mode	M	
Résultat (+)		S
Résultat (−)		S
ErrorInfo		M

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Mode

Valeurs admises:
IDLE (Le Dispositif passe en attente de configuration),
SIO (Le Dispositif passe au mode défini dans le service "SM_SetDeviceCom").

Résultat (+):

Ce paramètre de sélection indique que le service a été exécuté avec succès.

Résultat (−):

Ce paramètre de sélection indique que le service a échoué.

ErrorInfo

Ce paramètre contient des informations d'erreur.

Valeurs admises:
STATE_CONFLICT (service non disponible en l'état courant).

9.3.2.7 SM_DeviceMode

Le service SM_DeviceMode est utilisé pour indiquer des changements d'états de communication à l'application du Dispositif. Les paramètres des primitives de service sont énumérés dans le Tableau 94.

Tableau 94 – SM_DeviceMode

Nom du paramètre	.ind
Argument	M
Mode	M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Les paramètres spécifiques au service sont transmis dans l'argument.

Mode

Valeurs admises:

IDLE	(Le Dispositif passe en attente de configuration),
SIO	(Le Dispositif passe au mode défini dans le service "SM_SetDeviceCom"),
ESTABCOM	(Le Dispositif passe en mode SM "SM_ComEstablish"),
COM1	(Le Dispositif passe en mode COM1),
COM2	(Le Dispositif passe en mode COM2),
COM3	(Le Dispositif passe en mode COM3),
STARTUP	(Le Dispositif passe en mode STARTUP),
IDENT_STARTUP	(Le Dispositif passe en mode SM "SM_IdentStartup"),
IDENT_CHANGE	(Le Dispositif passe en mode SM "SM_IdentCheck"),
PREOPERATE	(Le Dispositif passe en mode PREOPERATE),
OPERATE	(Le Dispositif passe en mode OPERATE).

9.3.3 Protocole SM du Dispositif

9.3.3.1 Vue d'ensemble

Le comportement du Dispositif est principalement piloté par les messages du Maître.

9.3.3.2 Diagramme d'états de la SM du Dispositif

La Figure 81 représente le diagramme d'états du module de traitement de ligne SM du Dispositif. Il est déclenché par le module de traitement de Mode DL et l'application du Dispositif. Il évalue les différentes phases de communication au cours du démarrage et contrôle l'état de la ligne du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

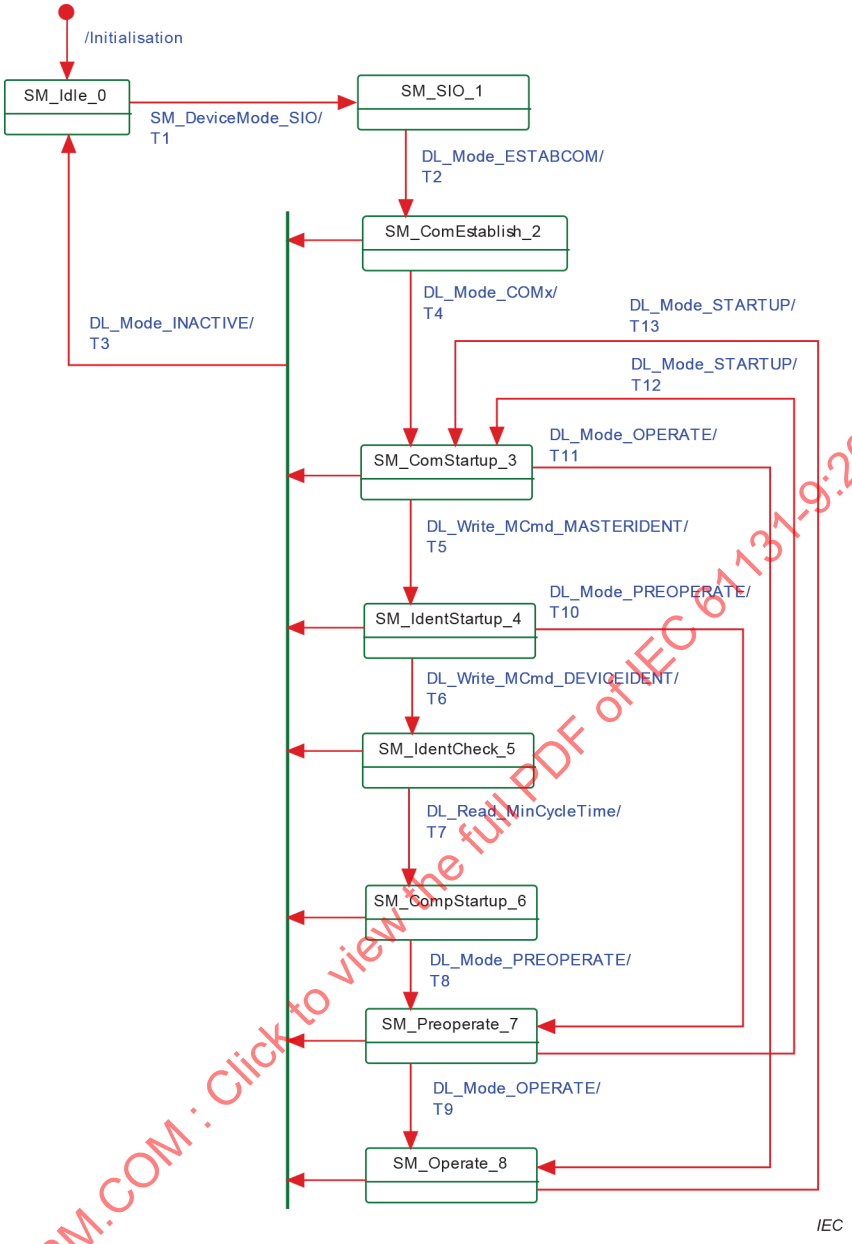


Figure 81 – Diagramme d'états de la Gestion Système du Dispositif

Le Tableau 95 définit les états individuels et les actions dans les transitions.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 95 – Table de transitions d'états de la Gestion Système du Dispositif

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
SM_Idle_0	En l'état SM_Idle, la SM est en attente de configuration par l'application du Dispositif et d'établissement du mode SIO. Cet état est abandonné à la réception d'une demande SM_SetDeviceMode(SIO) en provenance de l'application du Dispositif La séquence suivante de services doit être exécutée entre l'application du Dispositif et la SM. Invocation de SM_SetDeviceCom(Liste de paramètres initiale) Invocation de SM_SetDeviceIdent(VID, DID initial, FID)
SM_SIO_1	En l'état SM_SIO, le module de traitement de ligne SM reste en mode SIO par défaut. La Couche Physique est réglée sur les caractéristiques du mode SIO définies par l'application du Dispositif au moyen du service SetDeviceMode. Cet état est abandonné à la réception d'une indication DL_Mode(ESTABCOM).
SM_ComEstablish_2	En l'état SM_ComEstablish, la SM est en attente d'établissement de la communication dans la Couche de Liaison de Données. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE) ou DL_Mode(COMx), où COMx peut être COM1, COM2 ou COM3.
SM_ComStartup_3	En l'état SM_ComStartup, les paramètres de communication (page 1 de Paramètres Directs, adresses 0x02 à 0x06) sont lus par la SM du Maître par l'intermédiaire de demandes DL_Read. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE), d'une indication DL_Mode(OPERATE) (Maître hérité uniquement), ou d'une demande DL_Write(MCmd_MASTERIDENT) (Maître conforme au présent document).
SM_IdentStartup_4	En l'état SM_IdentStartup, les données d'identification (VID, DID, FID) sont lues et vérifiées par le Maître. En cas d'incompatibilités, la SM du Maître écrit la version SDCI (RID) prise en charge et le DeviceID (DID) configuré vers le Dispositif. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE), d'une indication DL_Mode(PREOPERATE) (vérification de compatibilité réussie), ou une demande DL_Write(MCmd_DEVICEIDENT) (nouvelle compatibilité demandée).
SM_IdentCheck_5	En l'état SM_IdentCheck, la SM attend une nouvelle initialisation des paramètres de communication et d'identification. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE) ou une demande DL_Read(page 1 de Paramètres Directs, adresses 0x02 = "MinCycleTime"). Dans cet état, l'application du Dispositif doit vérifier les paramètres RID et DID à partir de la SM et régler ces données aux valeurs prises en charge. En conséquence, la séquence suivante de services doit être exécutée entre l'application du Dispositif et la SM. Invocation de SM_GetDeviceCom(RID configuré, liste de paramètres) Invocation de SM_GetDeviceIdent(DID configuré, liste de paramètres) Invocation de l'application du Dispositif qui vérifie et fournit la fonction de compatibilité et les paramètres Invocation de SM_SetDeviceCom(nouveau RID pris en charge, nouvelle liste de paramètres) Invocation de SM_SetDeviceIdent(nouveau DID pris en charge, liste de paramètres)
SM_CompStartup_6	En l'état SM_CompatStartup, les données de communication et d'identification sont relues et vérifiées par la SM du Maître. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE) ou DL_Mode(PREOPERATE).
SM_Preoperate_7	Au cours de l'état SM_Preoperate, le Numéro de série peut être lu et vérifié par la SM du Maître; de même que le Stockage des données et le paramétrage du Dispositif peuvent être exécutés. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE), DL_Mode(STARTUP) ou DL_Mode(OPERATE).
SM_Operate_8	Au cours de l'état SM_Operate, l'échange cyclique de Données de Processus et le transfert acyclique de Données à la Demande sont actifs. Cet état est abandonné à la réception d'une indication DL_Mode(INACTIVE) ou DL_Mode(STARTUP).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Le Dispositif passe en mode SIO configuré à la réception de la demande de déclenchement SM_SetDeviceMode.req(SIO). Invocation de PL_SetMode(DI|DO|INACTIVE) Invocation de SM_DeviceMode(SIO)
T2	1	2	Le Dispositif passe en mode de communication à la réception de l'indication de déclenchement DL_Mode.ind(ESTABCOM). Invocation de PL_SetMode(COMx) Invocation de SM_DeviceMode(ESTABCOM)
T3	2,3,4,5,6, 7,8	0	Le Dispositif passe en mode SM_Idle à la réception de l'indication de déclenchement DL_Mode.ind(INACTIVE). Invocation de PL_SetMode(INACTIVE) Invocation de SM_DeviceMode(IDLE)
T4	2	3	L'application du Dispositif reçoit une indication concernant le débit en bauds d'établissement de la communication dans la couche DL, déclenché par DL_Mode.ind(COMx). Invocation de SM_DeviceMode(COMx)
T5	3	4	La phase d'identification du Dispositif commence à la réception de l'indication de déclenchement DL_Write.ind(MCmd_MASTERIDENT). Invocation de SM_DeviceMode(IDENTSTARTUP)
T6	4	5	La phase de vérification de l'identité du Dispositif commence à la réception de l'indication de déclenchement DL_Write.ind(MCmd_DEVICEIDENT). Invocation de SM_DeviceMode(IDENTCHANGE)
T7	5	6	La phase de départ de la vérification de la compatibilité du Dispositif commence à la réception de l'indication de déclenchement DL_Read.ind (page 1 de Paramètres Directs, adresse 0x02 = "MinCycleTime").
T8	6	7	Le Dispositif entre dans la phase de préfonctionnement lorsqu'il reçoit le déclencheur DL_Mode.ind(PREOPERATE). Invocation de SM_DeviceMode(PREOPERATE)
T9	7	8	Le Dispositif entre dans la phase de fonctionnement lorsqu'il reçoit le déclencheur DL_Mode.ind(OPERATE). Invocation de SM_DeviceMode(OPERATE)
T10	4	7	Le Dispositif entre dans la phase de préfonctionnement lorsqu'il reçoit le déclencheur DL_Mode.ind(PREOPERATE). Invocation de SM_DeviceMode(PREOPERATE)
T11	3	8	Le Dispositif entre dans la phase de fonctionnement lorsqu'il reçoit le déclencheur DL_Mode.ind(OPERATE). Invocation de SM_DeviceMode(OPERATE)
T12	7	3	Le Dispositif entre dans la phase de démarrage de la communication lorsqu'il reçoit le déclencheur DL_Mode.ind(STARTUP). Invocation de SM_DeviceMode(STARTUP)
T13	8	3	Le Dispositif entre dans la phase de démarrage de la communication lorsqu'il reçoit le déclencheur DL_Mode.ind(STARTUP). Invocation de SM_DeviceMode(STARTUP)

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
COMx	Variable	N'importe quelle vitesse de transmission: COM1, COM2 ou COM3
DL_Write_MCmd_xxx	Service	Le service DL écrit des MasterCommands (xxx = valeurs tirées du Tableau B.2)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

La Figure 82 représente un organigramme séquentiel type pour le démarrage de la communication de la SM d'un Dispositif dont les réglages correspondent à la configuration de l'Accès du Maître (démarrage ordinaire).

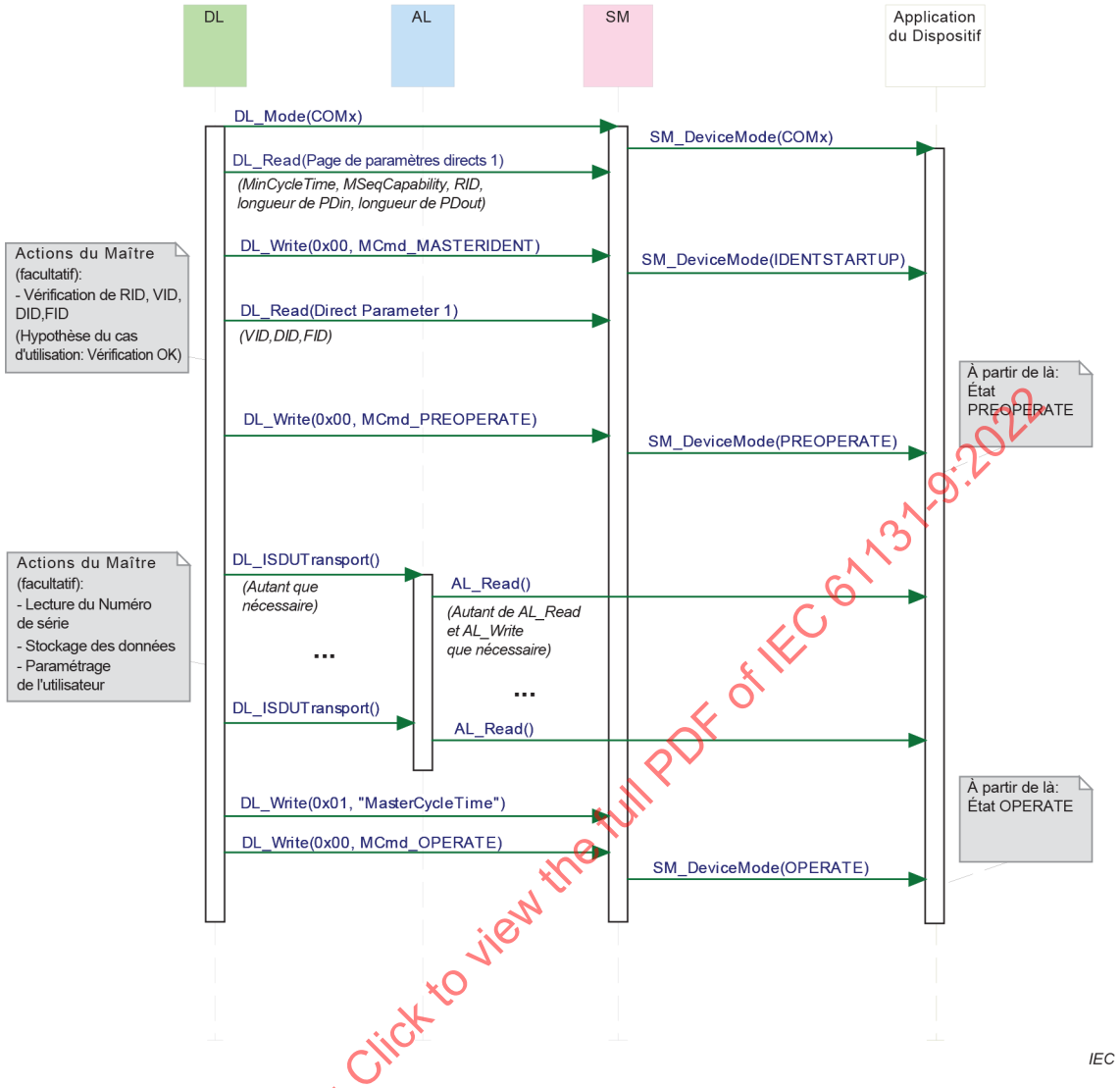


Figure 82 – Organigramme séquentiel de démarrage ordinaire du Dispositif

La Figure 83 représente un organigramme séquentiel type pour le démarrage de la communication de la SM d'un Dispositif dont les réglages ne correspondent pas à la configuration de l'Accès du Maître (mode compatibilité). Dans ce mode, le Maître tente d'écraser les paramètres d'identification du Dispositif pour parvenir à un mode compatible et réalisable.

L'organigramme séquentiel de la Figure 83 représente uniquement les actions jusqu'à l'état PREOPERATE. Les actions restantes jusqu'à l'état OPERATE peuvent être déduites de la Figure 82.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

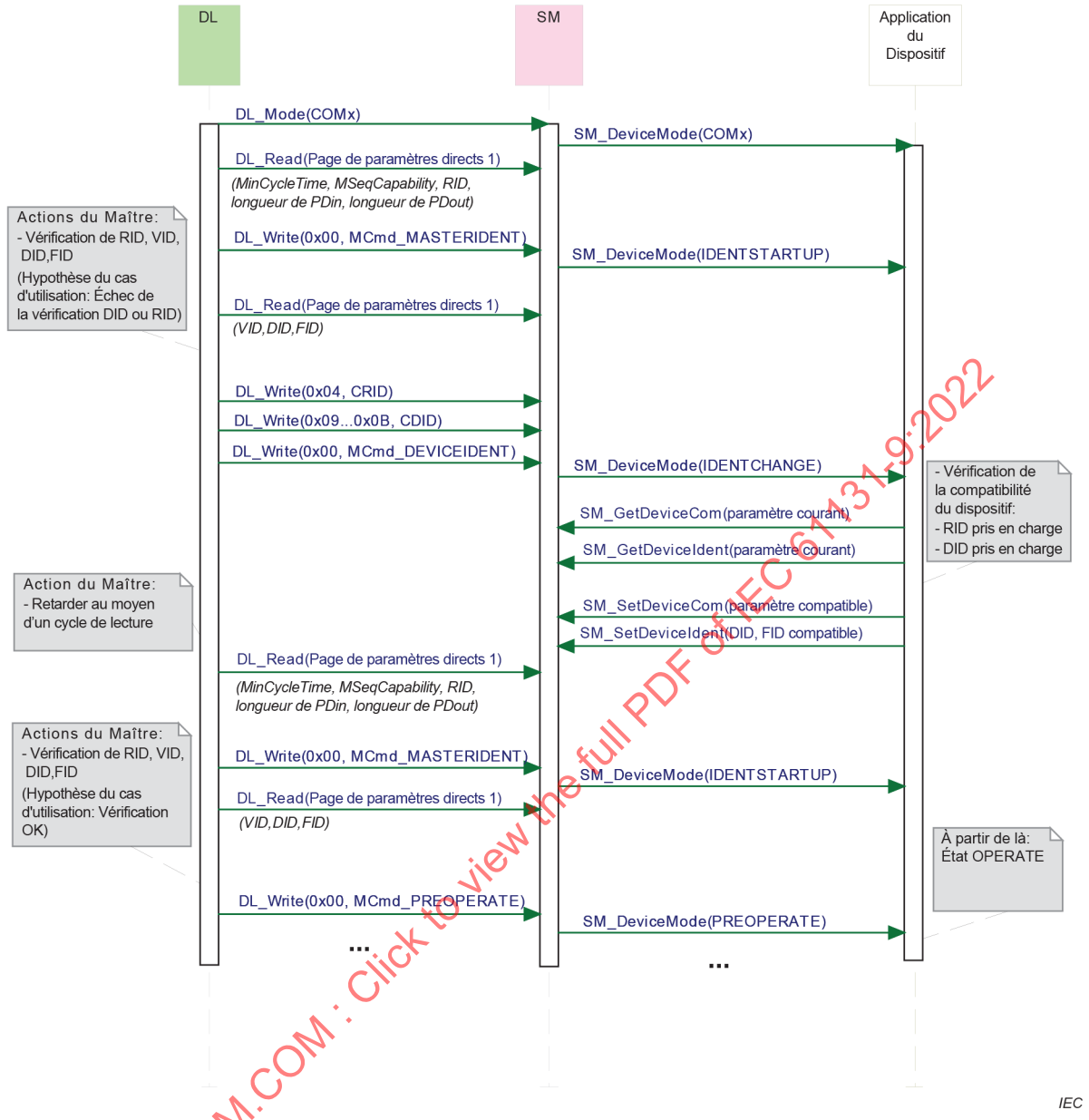


Figure 83 – Organigramme séquentiel du démarrage d'un Dispositif en mode compatibilité

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

La Figure 84 représente un organigramme séquentiel type pour le démarrage de la communication de la SM d'un Dispositif dont les réglages ne correspondent pas à la configuration de l'Accès du Maître. La Gestion Système du Maître tente de reconfigurer le Dispositif avec d'autres paramètres d'identification du Dispositif (mode compatibilité). Dans le présent cas d'utilisation, les paramètres de remplacement sont réputés incompatibles.

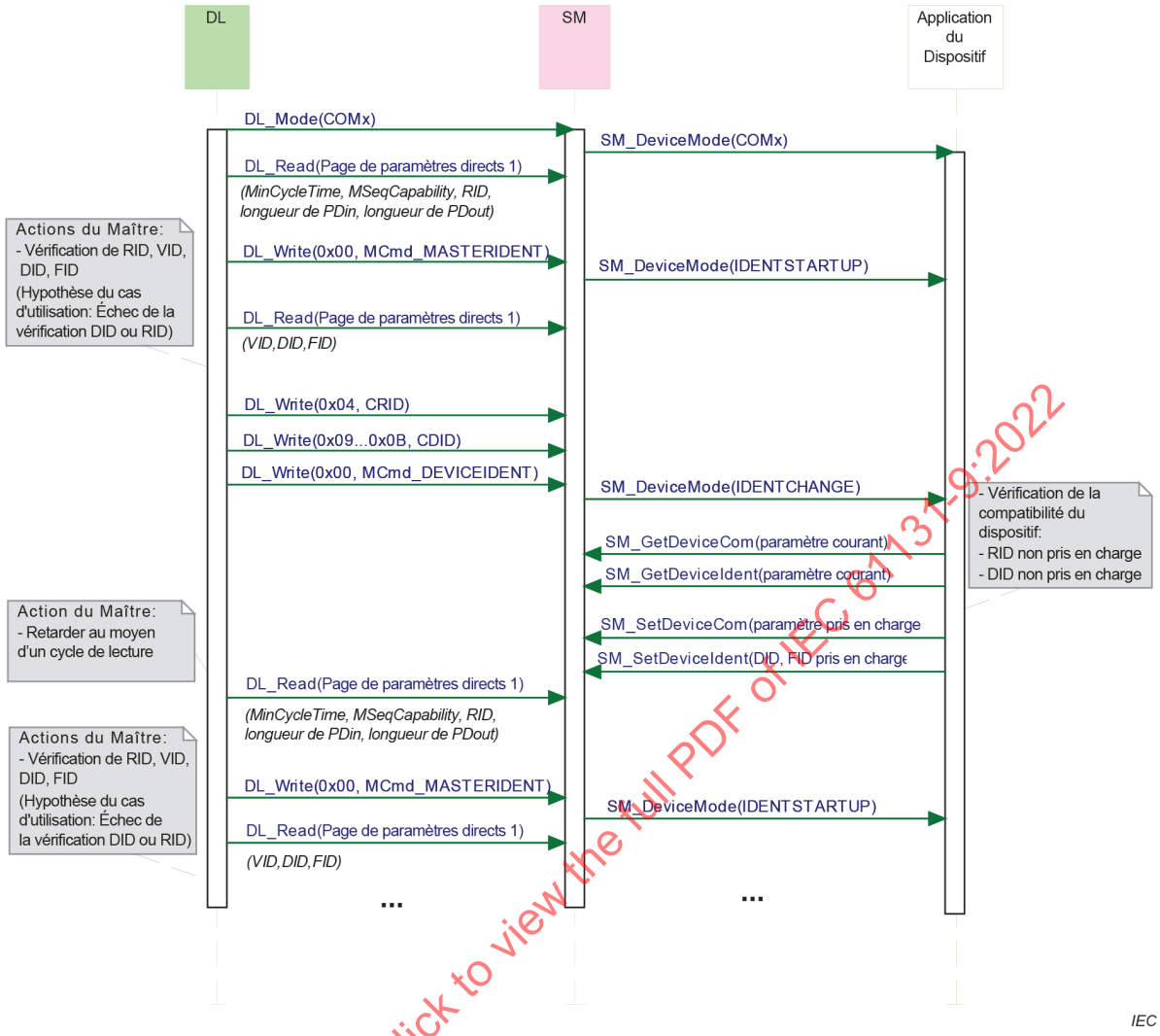


Figure 84 – Organigramme séquentiel du démarrage d'un Dispositif en cas d'échec de la compatibilité

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10 Dispositif

10.1 Aperçu

La Figure 85 donne une vue d'ensemble de la structure et des services d'un Dispositif.

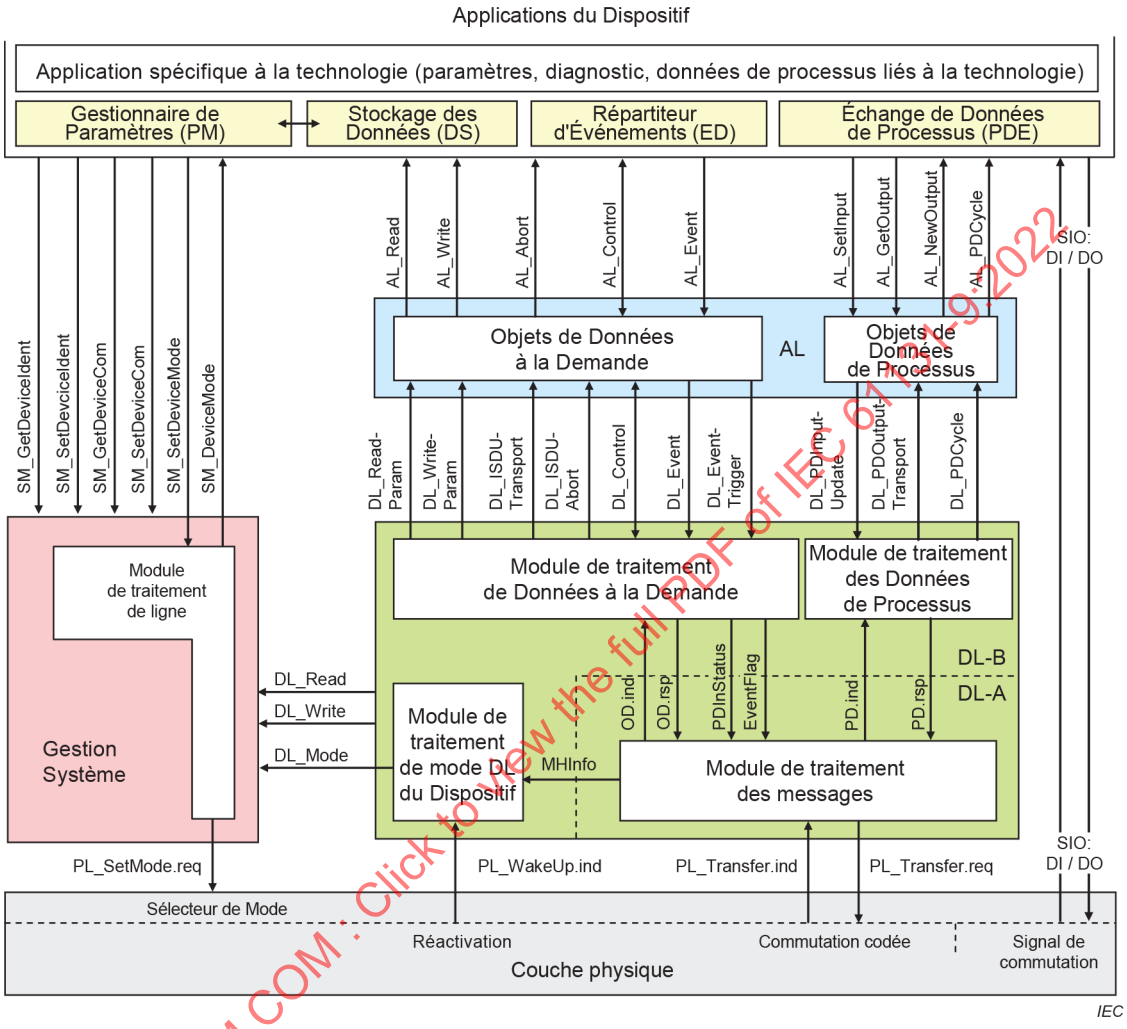


Figure 85 – Structure et services d'un Dispositif

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Les applications d'un Dispositif comprennent, en premier lieu, l'application spécifique à la technologie qui est constituée du transducteur ainsi que de ses paramètres technologiques, ses informations de diagnostic et ses Données de Processus. Les applications communes d'un Dispositif comprennent:

– un Gestionnaire de Paramètres (PM) chargé des vérifications de la compatibilité et du bien-fondé de l'ensemble des jeux de paramètres spécifiques à la technologie (fournisseur) et communs au système (voir 10.3);

– un mécanisme de Stockage des Données(DS) qui peut optionnellement télécharger des paramètres à partir du Maître ou vers le Maître (voir 10.4);

– un Répartiteur d'Événements (ED) qui supervise les états et achemine des informations de diagnostic telles que les notifications, les avertissements, les erreurs ainsi que les demandes du Dispositif telles les initiatives périphériques (voir 10.5);

– une unité d'Échange de Données de Processus (PDE) qui conditionne les structures de données pour la transmission dans le cas d'un capteur ou qui prépare les structures de données reçues pour la génération de signaux. Cette unité contrôle également les états opérationnels afin de garantir la validité des Données de Processus (voir 10.2).

Ces applications de Dispositif fournissent des méthodes ou des fonctions et des paramètres normalisés communs à tous les Dispositifs, ainsi que des fonctions et des paramètres spécifiques au Dispositif, tous ces éléments étant définis dans l'Article 10.

10.2 Échange de Données de Processus (PDE)

L'unité d'Échange de Données de Processus transmet et reçoit de manière cyclique des Données de Processus sans interférence avec les Données à la Demande (paramètres, commandes et Événements).

Un actionneur (Données de Processus de sortie) doit observer l'émission cyclique et passer à un état par défaut approprié, pour garder par exemple la dernière valeur, s'arrêter ou se désactiver dès que l'émission de données est interrompue (voir 7.3.3.5 et 10.8.3). L'actionneur doit attendre la MasterCommand "ProcessDataOutputOperate" (voir le Tableau B.2, Données de Processus de sortie "valides") avant tout fonctionnement normal après redémarrage en cas d'interruption.

Au cours de l'échange des données cycliques, un actionneur (Données de Processus de sortie) reçoit une Commande du Maître "DeviceOperate" lorsque les Données de Processus de sortie ne sont pas valides et une Commande du Maître "ProcessDataOutputOperate" lorsqu'elles sont de nouveau valides (voir le Tableau B.2).

Il n'est pas nécessaire qu'un Dispositif capteur (Données de Processus d'entrée) surveille l'échange des données cycliques. Cependant, si le Dispositif est incapable de garantir des Données de Processus valides, l'état des PD "Données de Processus non valides" (voir A.1.5) doit être signalé à l'application du Maître.

10.3 Gestionnaire de Paramètres (PM)

10.3.1 Généralités

Un Dispositif peut être paramétré selon deux méthodes de base, en utilisant les Paramètres Directs ou l'espace mémoire des index accessible au moyen des ISDU (voir la Figure 6).

Les Paramètres dits Directs en page 1 sont obligatoires pour tous les Dispositifs. Cette page 1 contient des paramètres communs de communication et d'identification (voir l'Article B.1).

La page 2 de Paramètres Directs offre optionnellement un espace pour au maximum 16 octets de paramètres spécifiques à la technologie (fournisseur) pour les Dispositifs qui n'exigent pas plus de données que ce nombre limité et qui ont une empreinte système faible (communication des ISDU non prise en charge, éventuel traitement des bus de terrain possible, mais moins pratique). L'Accès à la page 2 de Paramètres Directs est réalisé par l'intermédiaire de AL_Read et AL_Write (voir 10.8.5).

La transmission des paramètres vers et en provenance de la mémoire des index spacieuse peut être effectuée de deux manières: paramètre unique par paramètre unique ou sous forme de bloc de paramètres. La transmission par paramètre unique telle qu'elle est spécifiée au 10.3.4 est assurée au moyen de plusieurs vérifications et par la confirmation du paramètre transmis. Un acquittement négatif contient une description appropriée de l'erreur et le paramètre n'est alors pas activé. La transmission sous forme de bloc de paramètres telle qu'elle est spécifiée au 10.3.5 reporte la vérification de la cohérence des paramètres, ainsi que l'activation jusqu'à l'accomplissement complet de la transmission. Le Dispositif effectue les vérifications sur réception d'une commande spéciale et renvoie une confirmation ou un acquittement négatif accompagné d'une description appropriée de l'erreur. Dans ce cas, les paramètres transmis doivent être rejetés et un retour au jeu de paramètres précédent doit être effectué pour assurer une fonctionnalité correcte du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.3.2 Diagramme d'états du Gestionnaire de Paramètres

Le Dispositif peut être paramétré au moyen de mécanismes d'ISDU lorsque le PM est actif. Les principales fonctions du PM sont la transmission de paramètres vers le Maître ("Téléchargement amont") et vers le Dispositif ("Téléchargement aval"), ainsi que la vérification de la cohérence et de la validité au sein du Dispositif ("Vérification de validité"), comme cela est représenté à la Figure 86.

Le PM est piloté par des messages de commande du Maître (voir le Tableau B.9). Par exemple, la protection [UploadStart] correspond à la réception de la SystemCommand (Commande Système) "ParamUploadStart" et [UploadEnd] correspond à la réception de la SystemCommand "ParamUploadEnd".

NOTE 1 Après une interruption de la communication, la Gestion Système du Maître utilise le service SM_DeviceMode avec la variable "INACTIVE" pour arrêter le processus de téléchargement et revenir à l'état "IDLE".

Toute nouvelle commande "ParamUploadStart" ou "ParamDownloadStart" alors qu'une autre séquence est en cours, due par exemple à l'arrêt imprévu de l'outil de paramétrage du fournisseur, entraîne l'abandon de la séquence en cours. Les modifications de paramètres correspondantes sont alors ignorées.

NOTE 2 Il peut y avoir conflit entre un programme utilisateur d'Automate programmable et un outil de paramétrage (Accès multiples), par exemple, si au cours de la mise en service, l'utilisateur n'a pas désactivé les Accès qui proviennent du programme de l'Automate programmable pendant la modification des paramètres au moyen de l'outil.

Le mécanisme de Gestionnaire de Paramètres dans un Dispositif est toujours actif et la demande DS_ParUpload.req au cours de la transition T4 est utilisée pour déclencher le mécanisme de Stockage des Données (DS) décrit en 10.4.2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

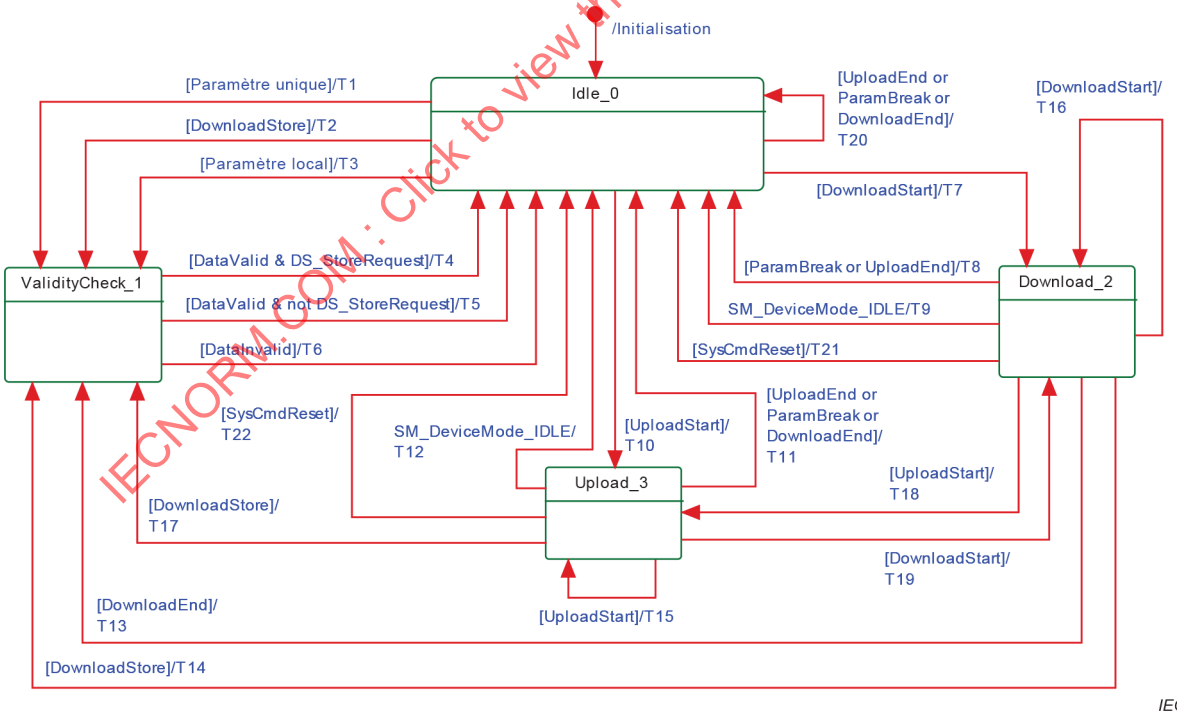


Figure 86 – Diagramme d'états du Gestionnaire de Paramètres (PM)

Le Tableau 96 présente la table de transitions d'états du diagramme d'états du Gestionnaire de Paramètres (PM) du Dispositif.

Tableau 96 – Table de transitions d'états du diagramme d'états du PM

DÉNOMINATION DE L'ÉTAT		DESCRIPTION DE L'ÉTAT	
Idle_0		Attente de transmission de paramètres	
ValidityCheck_1		Vérification de la cohérence et de la validité du jeu de paramètres courant.	
Download_2		Téléchargement aval de paramètres actif; paramétrage local verrouillé (par exemple, apprentissage en cours). Tous les services de Lecture vers des Indices autres que 3 (DataStorageIndex) doivent être refusés (ISDU ErrorType 0x8022 – "Service temporairement non disponible – contrôle du Dispositif")	
Upload_3		Téléchargement amont de paramètres actif; paramétrage globalement verrouillé. Tous les Accès en écriture pour modifier les paramètres non couverts par le diagramme d'états doivent être refusés (Code d'Erreur ISDU "Service temporairement non disponible – contrôle du Dispositif")	
TRANSITION	**ÉTAT DE LA SOURCE**	**ÉTAT DE LA CIBLE**	**ACTION**
T1	0	1	–
T2	0	1	Établissement de "StoreRequest" (= VRAI)
T3	0	1	Établissement de "StoreRequest" (= VRAI)
T4	1	0	Marquage du jeu de paramètres comme étant valide; Invocation de DS_ParUpload.req vers DS; activation de l'acquittement positif de la transmission; Réinitialisation de "StoreRequest" (= FAUX)
T5	1	0	Marquage du jeu de paramètres comme étant valide; activation de l'acquittement positif de la transmission
T6	1	0	Marquage du jeu de paramètres comme étant non valide; activation de l'acquittement négatif de la transmission; Réinitialisation de "StoreRequest" (= FAUX); Vidage du tampon de paramètres
T7	0	2	Verrouillage de l'Accès local aux paramètres
T8	2	0	Déverrouillage de l'Accès local aux paramètres; Vidage du tampon de paramètres
T9	2	0	Déverrouillage de l'Accès local aux paramètres; Vidage du tampon de paramètres
T10	0	3	Verrouillage de l'Accès local aux paramètres
T11	3	0	Déverrouillage de l'Accès local aux paramètres
T12	3	0	Déverrouillage de l'Accès local aux paramètres
T13	2	1	Déverrouillage de l'Accès local aux paramètres
T14	2	1	Déverrouillage de l'Accès local aux paramètres; Établissement de "StoreRequest" (= VRAI)
T15	3	3	Verrouillage de l'Accès local aux paramètres
T16	2	2	Vidage du tampon de paramètres, de façon à ne pas bloquer un éventuel second démarrage.
T17	3	1	Déverrouillage de l'Accès local au paramètre; établissement de "StoreRequest" (= VRAI)
T18	2	3	Vidage du tampon de paramètres, de façon à ne pas bloquer un éventuel second démarrage.
T19	3	2	–
T20	0	0	Renvoyer ErrorType 0x8036 – *Fonction temporairement non disponible* si le Paramétrage de bloc est pris en charge ou ErrorType 0x8035 – *Fonction non disponible* si le Paramétrage de bloc n'est pas pris en charge.
T21	2	0	Déverrouillage de l'Accès local au paramètre; vidage du tampon de paramètres
T22	3	0	Déverrouillage de l'Accès local au paramètre

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
DownloadStore	Bool	Commande Système "ParamDownloadStore" reçue, voir le Tableau B.9
DataValid	Bool	Résultat positif de la vérification de la conformité et de la validité
DataInvalid	Bool	Résultat négatif de la vérification de la conformité et de la validité
DownloadStart	Bool	Commande Système "ParamDownloadStart" reçue, voir le Tableau B.9
DownloadEnd	Bool	Commande Système "ParamDownloadEnd" reçue, voir le Tableau B.9
DS_StoreRequest	Bool	Drapeau pour une séquence de Stockage des Données demandées, c'est-à-dire Commande Système "ParamDownloadStore" reçue (= VRAI)
ParamBreak	Bool	SystemCommand "ParamBreak" reçue, voir le Tableau B.9
SysCmdReset	Bool	Une des SystemCommands de réinitialisation de parameter reçue, voir le Tableau 101
UploadStart	Bool	Commande Système "ParamUploadStart" reçue, voir le Tableau B.9
UploadEnd	Bool	Commande Système "ParamUploadEnd" reçue, voir le Tableau B.9
Single Parameter	Bool	En cas de "paramètre unique" comme cela est spécifié en 10.3.4
Local Parameter	Bool	En cas de "paramètre local" comme cela est spécifié en 10.3.3
"Verrouillage de l'Accès au paramètre" ne doit pas être confondu avec "Verrouillage de l'Accès au Dispositif" du Tableau B.12		

Le Gestionnaire de Paramètres (PM) prend en charge le traitement des transferts de "paramètre unique" (Index et Sous-index) ainsi que la transmission de "Paramètre de Bloc" (ensemble d'un jeu de paramètres).

10.3.3 Paramètre dynamique

Les paramètres accessibles par le biais des services de lecture ou d'écriture SDCI peuvent également être modifiés au moyen d'éléments de contrôle embarqués (par exemple, un bouton d'apprentissage) ou de l'interface homme-machine d'un Dispositif. Ces modifications doivent subir les mêmes vérifications de validité que l'Accès à un paramètre unique. Ainsi, en cas de résultat positif "DataValid" à la Figure 86, le drapeau "StoreRequest" doit être appliqué afin d'assurer la cohérence du Stockage des Données. En cas de résultat négatif "InvalidData", les valeurs précédentes des paramètres correspondants doivent être restaurées ("repli"). Par ailleurs, une indication spécifique au Dispositif sur l'interface homme-machine est recommandée en tant que retour d'information positif ou négatif vers l'utilisateur.

Il est recommandé d'éviter les Accès concurrents et simultanés à un paramètre au moyen des éléments de contrôle local et des services d'écriture SDCI.

10.3.4 Paramètre unique

La Figure 87 donne des exemples d'organigrammes séquentiels pour des modifications valides et non valides de paramètre unique.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

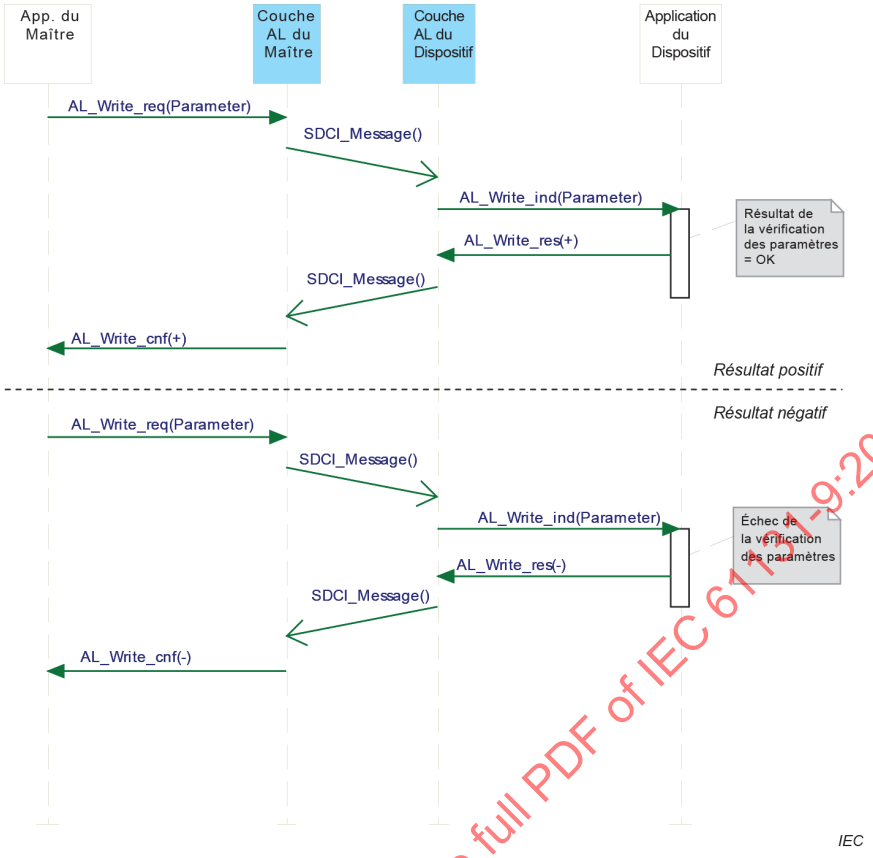


Figure 87 – Résultat positif et négatif d'une vérification de paramètre

Si un paramétrage unique est effectué par l'intermédiaire d'objets ISDU, le Dispositif doit vérifier l'Accès, la structure, la validité et la cohérence (voir le Tableau 97) des données transmises dans le contexte de l'ensemble du jeu de paramètres et retourner le résultat dans la confirmation. Avec une confirmation positive, le Dispositif indique que le contenu du paramètre

– a fait l'objet de toutes les vérifications du Tableau 97 dans l'ordre spécifié 1 à 4,
– est stocké dans la mémoire non volatile s'il s'agit de paramètres non volatiles, et
– est actif dans la technologie spécifique au Dispositif, le cas échéant.

La confirmation négative transporte l'un des ErrorTypes du Tableau C.2.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 97 – Définitions des vérifications de paramètre

Étape	Vérification de paramètre	Définition	Indication d'erreur
1	Access	Vérification des droits d'Accès valides à cet Index/Sous-index, indépendamment du contenu des données (Index/Sous-index indisponible en permanence ou temporairement; Accès en écriture/lecture à un Index en lecture/écriture seule)	Voir C.2.3 à C.2.8
2	Structure	Vérification de la structure valide des données, telle que la taille des données, seules les structures de données complètes peuvent être écrites, par exemple 2 octets pour un type de données UInteger16	Voir C.2.12 et C.2.13
3	Validity	Vérification du contenu valide des données de paramètres uniques, contrôle des limites de données	Voir C.2.9 à C.2.11, C.2.14, C.2.15
4	Cohérence	Vérification du contenu valide des données de l'ensemble du jeu de paramètres, contrôle des éventuelles interférences ou corrélations entre paramètres	Voir C.2.16 et C.2.17
NOTE Ces vérifications sont valides pour un seul paramètre et les Paramètres de Bloc (voir 10.3.5).			

10.3.5 Paramètre de bloc

Les applications utilisateur, telles que les blocs de fonctions dans les Automates programmables et les logiciels d'outils de paramétrage, peuvent utiliser les commandes de début et de fin pour indiquer le début et la fin de transmission d'un Paramètre de bloc. Pendant la durée de la transmission du Paramètre de bloc, l'application du Dispositif doit interdire toute modification de paramètre en provenance d'autres sources, par exemple paramétrage local ou apprentissage. Si l'Accès au paramètre est verrouillé, une application utilisateur doit déverrouiller "Accès (en écriture) au paramètre" (voir le Tableau B.12) avant de télécharger un jeu de paramètres.

Un exemple d'organigramme séquentiel pour des modifications valides de Paramètre de bloc, avec une demande facultative de Stockage des Données est représenté à la Figure 88.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

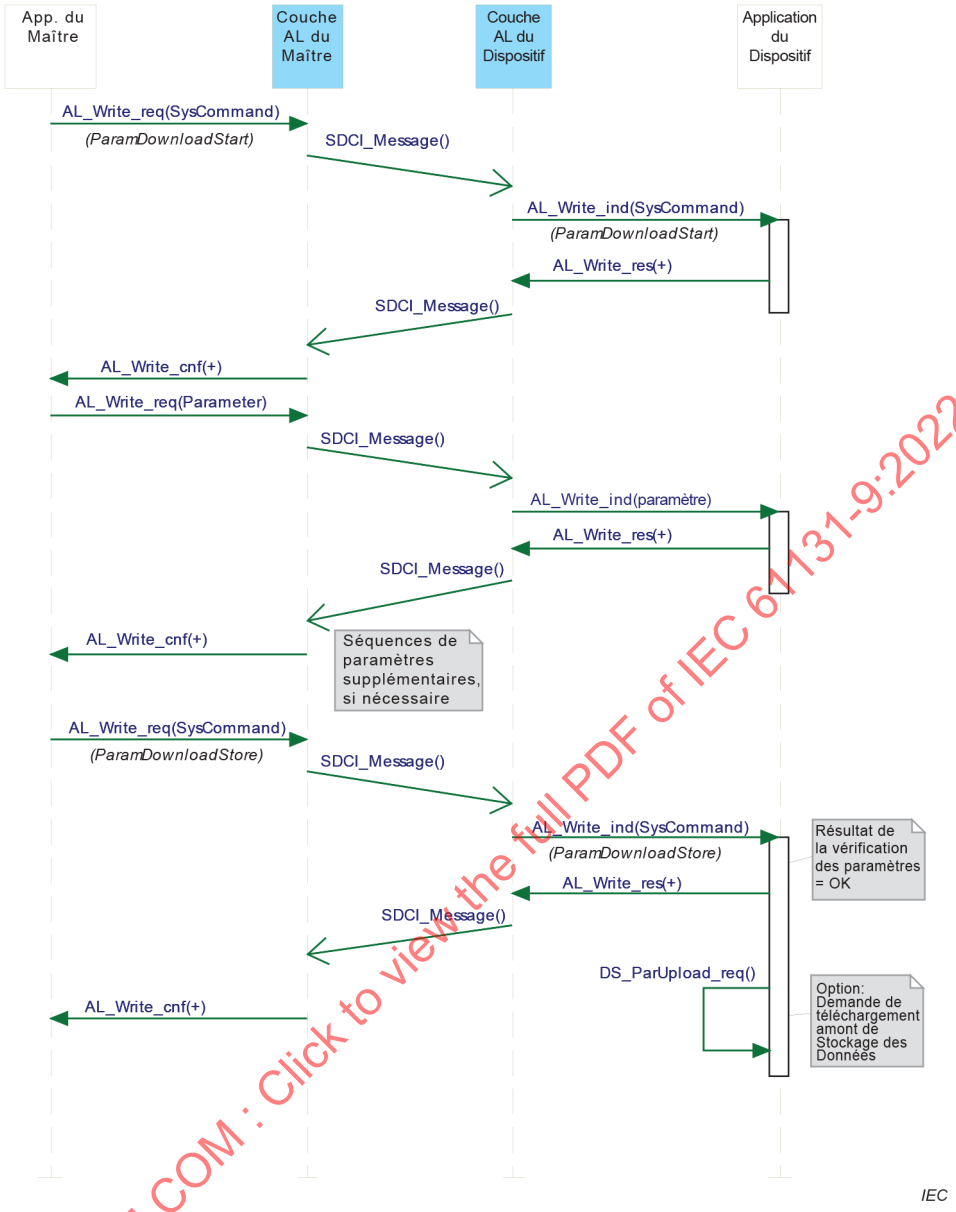


Figure 88 – Téléchargement aval positif de Paramètre de bloc avec demande de Stockage des Données

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Un exemple d'organigramme séquentiel pour des modifications non valides de Paramètre de bloc est représenté à la Figure 89.

La commande "ParamDownloadStart" (voir le Tableau B.9) indique le début de la transmission du Paramètre de bloc dans le sens de téléchargement aval (de l'application utilisateur vers le Dispositif). La SystemCommand "ParamDownloadEnd" ou "ParamDownloadStore" achève cette séquence. Les deux fonctions sont similaires. Cependant, la SystemCommand "ParamDownloadStore" force en outre le mécanisme de Stockage des Données (DS) à effectuer le téléchargement amont du jeu de paramètres par le biais de l'Événement DS_UPLOAD_REQ (voir 10.4.2).

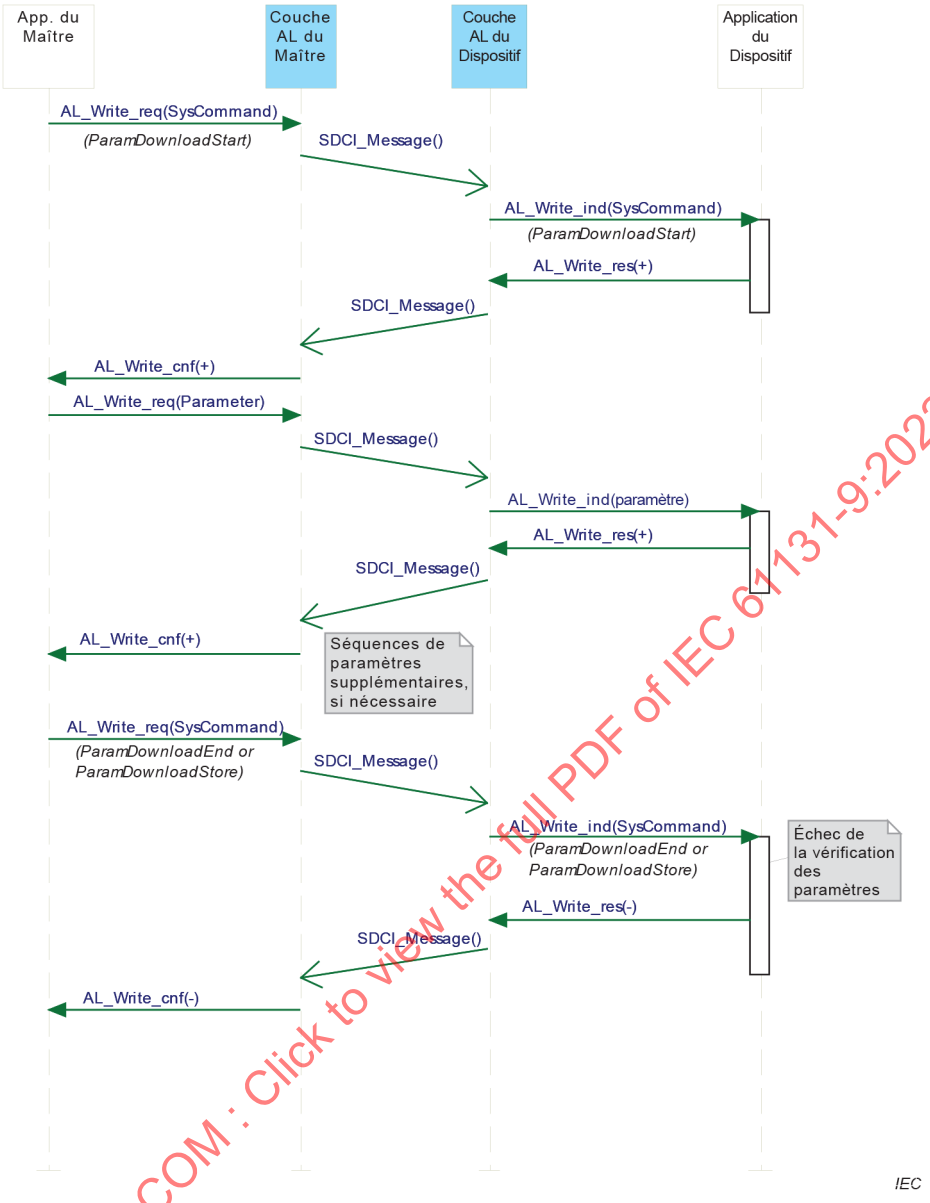


Figure 89 – Téléchargement aval négatif de Paramètre de bloc

Les étapes de vérification et les règles du Tableau 98 s'appliquent.

Tableau 98 – Étapes et règles pour la vérification de Paramètre de bloc

Règle	Action
1	En premier lieu, l'Accès et la structure doivent toujours être contrôlés pour chaque paramètre (voir le Tableau 97).
2	Ensuite, éventuellement, les contrôles de validité peuvent être réalisés pour chaque paramètre.
3	À cet instant, le contrôle de cohérence des paramètres transférés doit être désactivé et les paramètres uniques ne doivent pas être activés.
4	Le Gestionnaire de Paramètres ne doit pas quitter le mode de transfert de bloc en cas d'Accès non valide, de violations de la structure ou de défauts de validité. Le jeu de paramètres doit être traité comme non valide si l'un de ces contrôles échoue.
5	Avec la commande "ParamDownloadEnd" ou "ParamDownloadStore", le Dispositif contrôle la validité de chaque paramètre si cela n'a pas déjà été fait, ainsi que la cohérence de l'ensemble du jeu de paramètres. Le jeu de paramètres doit être traité comme non valide si l'un de ces contrôles échoue. Le résultat du contrôle est indiqué à l'émetteur de la transmission de Paramètre de bloc dans l'acquittement ISDU dans une commande de retour.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Règle	Action
6	Avec une confirmation positive, le Dispositif indique que les paramètres – ont fait l'objet de tous les contrôles du Tableau 97, – sont stockés dans la mémoire non volatile s'il s'agit de paramètres non volatils, – sont actifs dans la technologie spécifique au Dispositif, le cas échéant.
7	Avec une confirmation négative, le Dispositif indique que l'un des contrôles du Tableau 97 a échoué et que le jeu de paramètres n'est pas valide. Le précédent jeu de paramètres doit rester actif. Une demande de chargement de Stockage de données ne doit pas être déclenchée. La confirmation négative correspondante doit contenir l'ErrorType 0x8041 – Jeu de paramètres incohérent (voir C.2.17).

La commande "ParamUploadStart" (voir le Tableau B.9) indique le début de la transmission du Paramètre de bloc dans le sens du téléchargement amont (depuis le Dispositif vers l'application utilisateur). La SystemCommand "ParamUploadEnd" achève cette séquence et indique la fin de la transmission.

Une transmission de Paramètre de bloc est abandonnée si le Gestionnaire de Paramètres reçoit une SystemCommand "ParamBreak". Dans ce cas, la transmission de bloc est abandonnée sans aucune modification du paramétrage.

Dans tous les cas, la réponse à toutes les commandes "ParamXXX" (voir le Tableau B.9) doit être transmise après l'exécution de l'action demandée.

10.3.6 Accès concurrent au paramétrage

Il n'existe aucun mécanisme garantissant la cohérence des paramètres dans le Dispositif en cas d'Accès simultané à partir de différentes applications d'utilisateur au-dessus du niveau du Maître. Ceci doit être garanti ou bloqué au niveau de l'utilisateur (voir 13.2.2).

10.3.7 Traitement des commandes

Les commandes d'application sont acheminées sous la forme de paramètres. En tant que réponse de l'ISDU, le niveau de priorité approprié de la liste du Tableau 99 doit être utilisé.

Tableau 99 – Réponses hiérarchisées de l'ISDU sur les paramètres de commande

Priorité	Réponse de l'ISDU	Condition
1	"Index non disponible", voir C.2.13	Le paramètre de commande n'est pas pris en charge par le Dispositif
2	"Fonction non disponible", voir C.2.14	La commande n'est pas prise en charge par le Dispositif quel que soit l'état du Dispositif
3	"Fonction temporairement non disponible", voir C.2.15	La commande est prise en charge, mais l'état actuel du Dispositif ne permet pas la commande demandée.
4	Réponse (+) en écriture	La commande est prise en charge et acceptée dans l'état actuel du Dispositif, et l'action est terminée. Toutefois, dans le contexte de certaines commandes, l'action est simplement démarrée. Cette exception est définie au niveau d'une certaine commande.

Dans tous cas, l'expiration du délai imparti aux ISDU doit être respectée (voir le Tableau 102).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.4 Stockage des Données (DS)

10.4.1 Généralités

Le mécanisme de Stockage des Données (DS) permet une mise en mémoire tampon cohérente et actualisée des paramètres du Dispositif à des niveaux supérieurs, tels que les programmes d'Automate programmable ou le serveur de paramètre de bus de terrain. Le Stockage des Données entre Maîtres et Dispositifs est défini dans le présent document, tandis que les mécanismes adjacents de Stockage des Données de niveau supérieur dépendent du bus de terrain ou système spécifique. Le Dispositif détient un ensemble d'objets normalisés fournissant des informations sur les paramètres pour le Stockage des Données, telles que les exigences de taille de mémoire, ainsi que les informations de contrôle et d'état du mécanisme de Stockage des Données (voir le Tableau B.10). Les différentes versions des jeux de paramètres du Stockage des Données sont identifiées par une Somme de contrôle des paramètres.

Pendant le Stockage des Données, le Dispositif doit appliquer les mêmes règles de contrôle que celles spécifiées pour le transfert de Paramètre de bloc de 10.3.5.

Pour les Dispositifs, il est fortement recommandé de déployer le mécanisme DS défini dans le présent document. Si ce mécanisme n'est pas pris en charge, il incombe au fournisseur du Dispositif de décrire la manière d'assurer convenablement et sans outillage, dans un système donné, le paramétrage d'un Dispositif après remplacement.

10.4.2 Diagramme d'états du Stockage des Données

Toute modification d'un jeu de paramètres valides donne lieu à un nouveau téléchargement amont de Stockage des Données. Le téléchargement amont est lancé par le Dispositif lorsqu'il provoque un Événement "DS_UPLOAD_REQ" (voir le Tableau D.1. Le Dispositif doit stocker l'état interne "Téléchargement amont de Stockage des Données" dans une mémoire non volatile (voir le Tableau B.10, Propriété d'État), jusqu'à ce qu'il reçoive une commande de Stockage des Données "DS_UploadEnd" ou "DS_DownloadEnd".

Le Dispositif doit générer un Événement "DS_UPLOAD_REQ" (voir le Tableau D.1) uniquement si le jeu de paramètres est valide et

– si les paramètres destinés au Stockage des Données ont été modifiés localement sur le Dispositif (par exemple, apprentissage, interface homme-machine), ou
– si le Dispositif reçoit une SystemCommand "ParamDownloadStore".

Le mécanisme de Stockage des Données du Maître est déclenché par ces informations d'Événement et lance une séquence de téléchargement amont ou aval en fonction de la configuration de l'Accès. Le diagramme d'états de la Figure 90 définit le mécanisme de Stockage des Données du Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

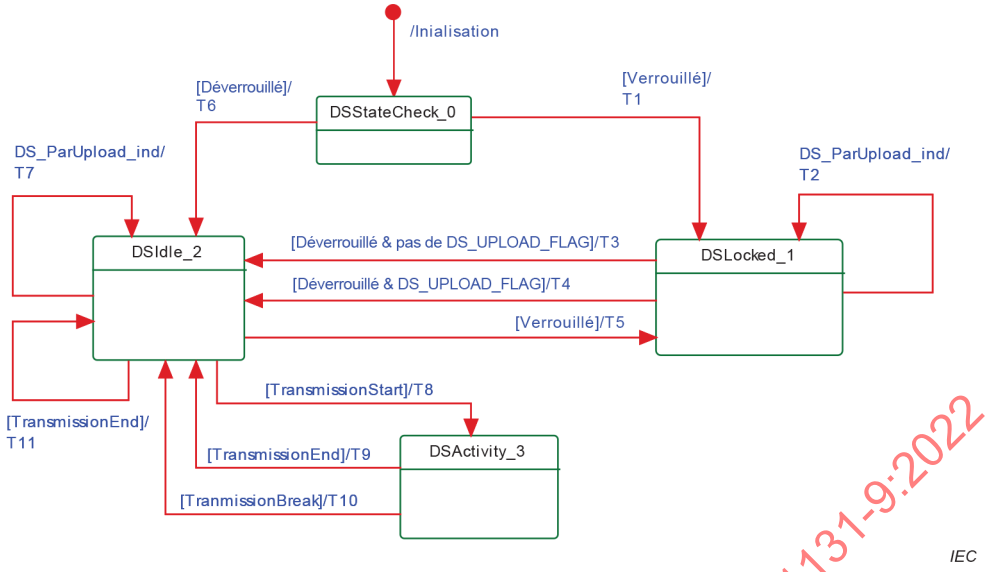


Figure 90 – Diagramme d'états du Stockage des Données (DS)

Le Tableau 100 présente la table de transitions d'états du diagramme d'états du Stockage des Données (DS) du Dispositif. Voir le Tableau B.10 pour plus de précisions concernant les attributions de DataStorageIndex.

Tableau 100 – Table de transitions d'états du diagramme d'états de Stockage des Données

DÉNOMINATION DE L'ÉTAT	DESCRIPTION DE L'ÉTAT
DSStateCheck_0	Vérification de l'état d'activation après initialisation.
DSLocked_1	Attente du déverrouillage du diagramme d'états de Stockage des Données. Cet état devient obsolète dans les versions ultérieures étant donné que le verrouillage d'Accès du Dispositif "Stockage des Données" ne doit plus être utilisé (voir le Tableau B.12).
DSIdle_2	Attente des activités de Stockage des données.
DSActivity_3	Fourniture du jeu de paramètres; paramétrage local verrouillé.

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T1	0	1	Établissement de State_Property = "Accès au Stockage des Données verrouillé"
T2	1	1	Établissement de DS_UPLOAD_FLAG = VRAI
T3	1	2	Établissement de State_Property = "Inactif"
T4	1	2	Invocation de AL_EVENT.req (EventCode: DS_UPLOAD_REQ), Établissement de State_Property = "Inactif"
T5	2	1	Établissement de State_Property = "Accès au Stockage des Données verrouillé"
T6	0	2	Établissement de State_Property = "Inactif"
T7	2	2	Établissement de DS_UPLOAD_FLAG = VRAI, invocation de AL_EVENT.req (EventCode: DS_UPLOAD_REQ)
T8	2	3	Verrouillage de l'Accès local aux paramètres, Établissement de State_Property = "Téléchargement amont" ou "Téléchargement aval"
T9	3	2	Établissement de DS_UPLOAD_FLAG = FAUX, déverrouillage de l'Accès local aux paramètres, Établissement de State_Property = "Inactif"

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

TRANSITION	ÉTAT DE LA SOURCE	ÉTAT DE LA CIBLE	ACTION
T10	3	2	Déverrouillage de l'Accès local aux paramètres, Établissement de State_Property = "Inactif"
T11	2	2	Établissement de DS_UPLOAD_FLAG = FAUX

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
Déverrouillé	Bool	Stockage des Données déverrouillé, voir B.2.4
Verrouillé	Bool	Stockage des Données verrouillé, voir B.2.4
DS_ParUpload.ind	Service	Service interne du Dispositif entre PM et DS (voir la Figure 86)
TransmissionStart	Bool	La DS_Command "DS_UploadStart" ou "DS_DownloadStart" a été invoquée
TransmissionEnd	Bool	La DS_Command "DS_UploadEnd" ou "DS_DownloadEnd" a été invoquée
TransmissionBreak	Bool	SM_MODE_INACTIVE ou DS_Command "DS_Break" reçu
"Verrouillage de l'Accès au paramètre" ne doit pas être confondu avec "Verrouillage de l'Accès au Dispositif" du Tableau B.12		

L'organigramme séquentiel tronqué de la Figure 91 représente les séquences de communication importantes après paramétrage.

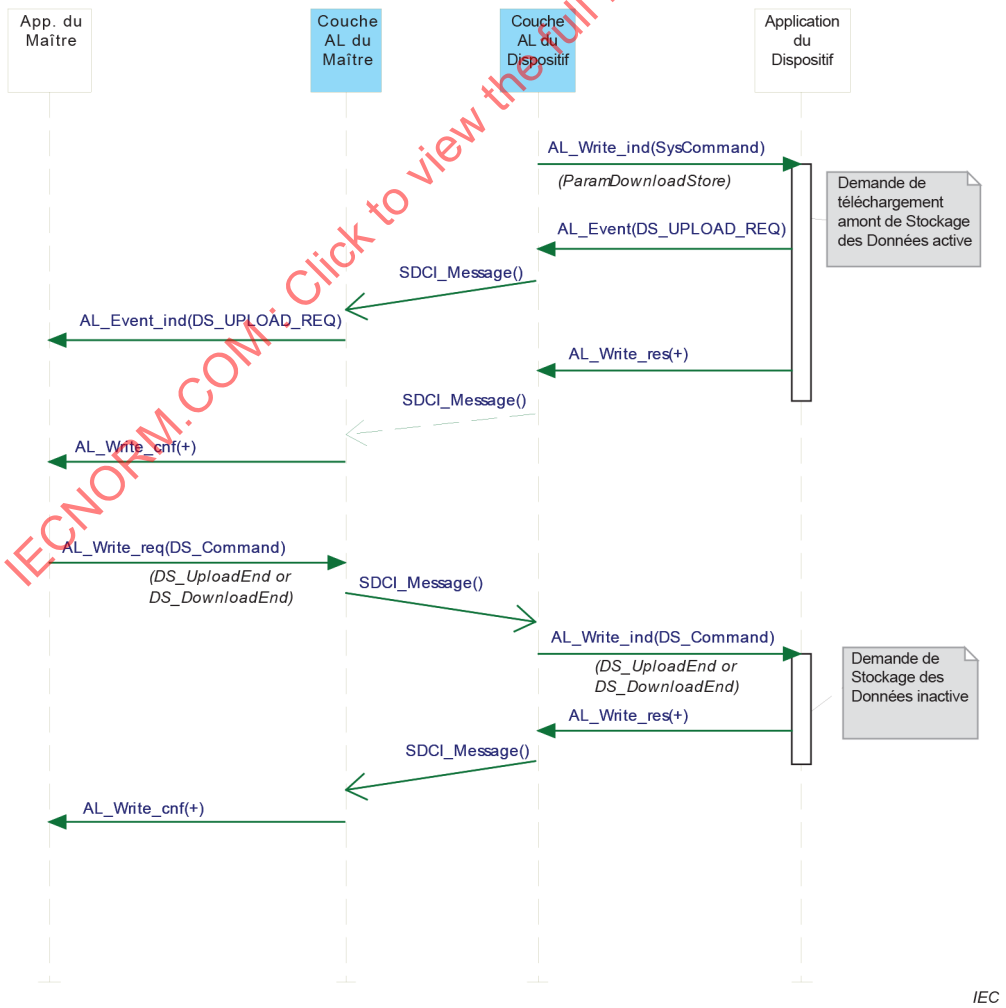


Figure 91 – Séquence du message de demande de Stockage des Données

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.4.3 Configuration du DS

Le mécanisme de Stockage des Données dans le Dispositif peut être désactivé par l'intermédiaire du Maître, par exemple au moyen d'un outil ou d'un programme d'automate programmable. Voir B.2.4 pour plus de précisions. Ceci est recommandé pendant une mise en service ou des essais du système pour éviter un flux de communication trop intensif.

NOTE Cette fonctionnalité est supprimée dans les versions ultérieures et le mécanisme de Stockage des Données est uniquement contrôlé par la configuration de l'Accès dans le Maître.

10.4.4 Espace mémoire du DS

Pour traiter en toutes circonstances la quantité de données à stocker, le nombre exigé d'Index à sauvegarder et l'espace mémoire total exigé sont fournis dans le paramètre Taille de Stockage des Données; voir le Tableau B.10. L'espace mémoire total requis (y compris les informations structurelles) ne doit pas dépasser 2 048 octets (voir l'Annexe G). Le mécanisme de Stockage des Données du Maître doit pouvoir prendre en charge cette quantité de mémoire pour chaque Accès.

10.4.5 Index_List de DS

Le Dispositif est le "propriétaire" de l'Index_List de DS (voir le Tableau B.10). Son but est de fournir les informations nécessaires au remplacement d'un Dispositif. L'Index_List de DS doit être fixe pour tout DeviceID spécifique. Dans le cas contraire, l'intégrité des données entre le Maître et le Dispositif ne peut pas être garantie. La Liste des Index doit contenir le Marqueur de Fin (Termination Marker) (voir le Tableau B.10), si le Dispositif ne prend pas en charge le Stockage des Données (voir 10.4.1). Dans ce cas, la taille de stockage exigée doit être 0.

10.4.6 Disponibilité des paramètres de DS

Tous les Index énumérés dans la liste doivent pouvoir être lus et écrits par les SystemCommands "DS_UploadStart" ou "DS_DownloadStart" et "DS_UploadEnd" ou "DS_DownloadEnd" (voir le Tableau B.10). Si l'un des Index est rejeté par le Dispositif, le Maître du Stockage des Données abandonne le téléchargement (amont ou aval) par une SystemCommand "DS_Break". Dans ce cas, il n'y a pas de nouvelle tentative de la séquence de Stockage des Données.

10.4.7 DS sans ISDU

La prise en charge de la transmission de ISDU dans un Dispositif est une condition préalable pour le Stockage des Données de paramètres. Les paramètres de la page 2 de Paramètres Directs ne peuvent pas être sauvegardés et restaurés par le mécanisme de Stockage des Données.

10.4.8 Indication de modification de paramètres de DS

La Somme de contrôle Parameter_Checksum spécifiée dans le Tableau B.10 est utilisée comme indicateur des modifications apportées à un jeu de paramètres donné. Le présent document n'exige pas un mécanisme spécifique de détection des modifications de paramètres. Un ensemble de méthodes recommandées est fourni dans l'Annexe K.

10.5 Répartiteur d'Événements (ED)

Toute application du Dispositif peut générer des informations prédéfinies d'état système lorsque les opérations de l'interface SDCI échouent ou lorsqu'il apparaît des informations spécifiques à la technologie (diagnostic) par suite de méthodes de diagnostic spécifiques à la technologie. Le Répartiteur d'Événements transforme ces informations en Événement conformément aux définitions données à l'Article A.6. L'Événement consiste en un EventQualifier (Qualificatif d'Événement) indiquant les propriétés d'un incident et en un EventCode ID (Identifiant de Code d'Événement) qui est un descriptif de l'incident, accompagné des éventuelles mesures correctives. Le Tableau D.1 donne une liste des identifiants et descriptifs prédéfinis pour des

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

incidents orientés application. Des plages d'identifiants sont réservées aux incidents spécifiques au profil et au fournisseur. Le Tableau D.2 donne une liste des identifiants prédéfinis pour des incidents spécifiques à l'interface SDCI.

Les Événements sont classés en "Erreurs", "Avertissements" et "Notifications". Le paragraphe 10.10.2 fournit des recommandations sur la manière d'affecter ces classifications; la méthode utilisée par le Maître pour contrôler et traiter ces Événements est donnée en 11.6.

Tous les Événements fournis à un moment donné dans le temps sont acquittés par une commande unique. Par conséquent, l'acquittement de l'Événement peut être retardé par l'acquittement le plus lent en provenance des niveaux système supérieurs.

10.6 Fonctionnalités de Dispositif

10.6.1 Généralités

Les fonctionnalités de Dispositif suivantes sont définies jusqu'à un certain point pour obtenir un comportement commun. Elles sont accessibles par les méthodes ou paramètres normalisés ou spécifiques au Dispositif. La disponibilité de ces fonctionnalités est définie dans l'IODD d'un Dispositif.

10.6.2 Rétrocompatibilité d'un Dispositif

Cette fonctionnalité permet à un Dispositif de jouer le rôle d'une version de Dispositif précédente. Au cours de la phase de démarrage, la Gestion Système du Maître écrase l'identifiant de Dispositif (DID) inhérent au Dispositif et le remplace par l'ancien DeviceID requis. L'application spécifique à la technologie du Dispositif doit basculer sur les anciens ensembles ou sous-ensembles fonctionnels attribués à ce DeviceID. La prise en charge de la rétrocompatibilité est facultative pour un Dispositif donné.

Étant donné qu'un Dispositif peut assurer une rétrocompatibilité avec des DeviceID (DID) précédents, ces Dispositifs compatibles doivent prendre en charge tous les paramètres et capacités de communication du DeviceID précédent. Ainsi, le Dispositif est admis dans ce cas à modifier tout paramètre de communication (sauf la vitesse de transmission) ou d'identification.

10.6.3 Compatibilité de versions du protocole

Cette fonctionnalité permet à un Dispositif d'ajuster ses couches protocolaires sur une version précédente du protocole SDCI, telle que par exemple, sur la version du protocole hérité d'un Maître hérité ou à l'avenir, de la version V(x) à la version V(x-n). Au cours de la phase de démarrage, la Gestion Système du Maître peut écraser le RevisionID (ID de version) (RID) inhérent au protocole du Dispositif en cas de divergence avec le RevisionID pris en charge par le Maître. Un Maître hérité n'écrit pas la MasterCommand "MasterIdent" (voir le Tableau B.2) et le Dispositif peut ainsi s'adapter au protocole hérité (V1.0). La prise en charge de la compatibilité entre versions est facultative pour un Dispositif donné.

Les dispositifs prenant en charge le mode V1.0 et V1.1 sont admis

– à utiliser les mêmes paramètres, Événements et ErrorTypes prédéfinis dans les deux modes;
– pour prendre en charge le Paramétrage de Bloc avec toutes les fonctionnalités, y compris l'Événement "DS_UP-LOAD_REQ". Un Maître hérité propage ce type d'événements sans actions supplémentaires.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.6.4 Indication visuelle d'état de l'interface SDCI

Cette fonctionnalité indique l'état opérationnel de l'interface SDCI du Dispositif. L'indication du mode SDCI est spécifiée en 10.10.3. L'indication du mode SIO est spécifique au fournisseur et n'est pas couverte par la présente définition. La fonction est déclenchée par une indication de la Gestion Système (dans tous les états sauf SM_Idle et SM_SIO à la Figure 81). L'indication de l'état SDCI est facultative pour un Dispositif donné.

10.6.5 Verrouillage de l'Accès aux paramètres

Cette fonctionnalité permet à un Dispositif de verrouiller ou de déverrouiller globalement l'Accès en écriture à tous les paramètres modifiables du Dispositif accessible par l'intermédiaire de l'interface SDCI (voir B.2.4). Le verrouillage est déclenché par la réception d'un paramètre système "Verrouillage d'Accès au Dispositif" (voir le Tableau B.8). La prise en charge de ces fonctions est facultative pour un Dispositif donné. Toutefois, il est fortement recommandé de ne pas mettre en œuvre cette fonctionnalité (héritée) étant donné qu'elle est ignorée dans les versions ultérieures.

10.6.6 Verrouillage du Stockage des Données

L'établissement de ce verrou fait passer "State_Property" dans le Tableau B.10 à l'état "Stockage des Données verrouillé" et dans ce cas, le Dispositif n'envoie pas un Événement DS_UPLOAD_REQ. La prise en charge de cette fonction est facultative pour un Dispositif donné si le mécanisme de Stockage des Données est utilisé. Toutefois, il est fortement recommandé de ne pas mettre en œuvre cette fonctionnalité (héritée) étant donné qu'elle est ignorée dans les versions ultérieures.

10.6.7 Verrouillage des entrées locales du paramètre

Ce verrouillage doit avoir l'effet d'une protection en lecture seule ou en écriture des entrées locales au niveau du Dispositif (Bit 2 du Tableau B.12). La prise en charge de cette fonction est facultative pour un Dispositif donné, voir B.2.4.

10.6.8 Verrouillage de l'interface utilisateur

Ce verrouillage doit désactiver complètement les commandes et affichages, par exemple l'arrêt de l'interface homme-machine embarquée, tels que le pavé numérique, sur un Dispositif donné (Bit 3 du Tableau B.12). La prise en charge de cette fonction est facultative pour un Dispositif donné.

10.6.9 Décalage temporel

Le Décalage temporel t_{offset} est un paramètre à configurer par l'utilisateur (voir B.2.24). Il détermine le début du traitement des données spécifiques à la technologie du Dispositif par rapport au début du cycle de séquence M, c'est-à-dire le début du message de (Accès) Maître. Le décalage permet:

– de synchroniser le traitement des Données d'un Dispositif avec le cycle du (Accès) Maître dans certaines limites;

– de synchroniser entre eux les traitements des Données de plusieurs Dispositifs sur différents Accès du Maître;

– d'exécuter avec un décalage défini le traitement des Données de plusieurs Dispositifs sur différents Accès du Maître.

La Figure 92 représente la synchronisation des messages par rapport au traitement des données dans les Dispositifs.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

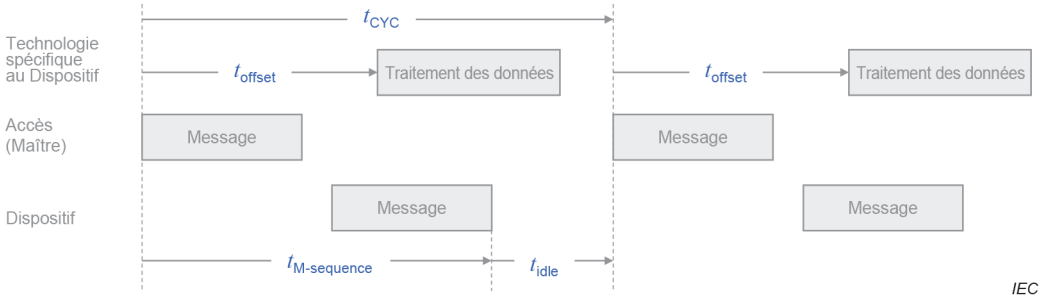


Figure 92 – Synchronisation du cycle

Le Décalage temporel définit un déclencheur par rapport au début d'un cycle de séquence M. La prise en charge de cette fonction est facultative pour un Dispositif donné.

10.6.10 Concept de Stockage des Données

Le mécanisme de Stockage des Données dans un Dispositif permet de sauvegarder automatiquement des paramètres dans le serveur de Stockage des Données du Maître et de les restaurer sur notification d'Événements. La cohérence des données est vérifiée dans chaque sens dans le Maître et dans le Dispositif. Le Stockage des Données concerne principalement les paramètres de configuration d'un Dispositif réglé lors de la mise en service (voir 10.4 et 11.4).

10.6.11 Paramètre de bloc

La fonctionnalité de transmission de paramètre de bloc dans un Dispositif permet le transfert de jeux de paramètres à partir d'un programme d'Automate programmable sans effectuer de vérification de la cohérence de chaque objet de données unique. La vérification de la validité et de la cohérence est réalisée à la fin de la transmission du paramètre de bloc pour l'ensemble du jeu de paramètres. Cette fonction concerne principalement l'échange de paramètres d'un Dispositif à configurer dans l'environnement d'exécution (voir 10.3). La prise en charge de cette fonction est facultative pour un Dispositif donné.

10.7 Options de réinitialisation du Dispositif

10.7.1 Vue d'ensemble

L'utilisateur a le choix entre cinq possibilités pour mettre un Dispositif dans un certain état défini

– Mise sous/hors tension (PowerCycle), ou
– SystemCommand "Réinitialisation du Dispositif" (128), ou
– SystemCommand "Réinitialisation de l'application" (129), ou
– SystemCommand "Restauration des réglages d'usine" (130), ou
– SystemCommand "Retour à la boîte" (131).

Le Tableau B.9 définit lesquelles de ces SystemCommands sont obligatoires, fortement recommandées ou facultatives.

Le Tableau 101 donne une vue d'ensemble des éléments impactés par l'exécution de l'une des options.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 101 – Vue d'ensemble des options de réinitialisation et de leur impact sur les Dispositifs

Élément impacté	Cycle de puissance	Réinitialisation du Dispositif	Réinitialisation de l'application	Rétablissement des paramètres d'usine	Retour à la boîte
Diagnostic et statut	"0"	"0"	Non	Supprimer	"0"
Enregistreur d'historique	Non	Non	Non	Non	Non
Paramètres spécifiques à la technologie (réglables, enseignables)	Non	Non	Par défaut	Par défaut	Par défaut
Identification/balises, par exemple, emplacement, fonction	Non	Non	Non	Par défaut	Par défaut
Comportement de Stockage des Données	Non	Non	Chargement en amont exigé DS_UPLOAD_REQ =1, DS Event	Suppression de la demande de chargement en amont DS_UPLOAD_REQ =0	Suppression de la demande de chargement en amont DS_UPLOAD_REQ =0
RevisionID	Par défaut	Par défaut	Non	Par défaut	Par défaut
DeviceID	Non	Non	Non	Par défaut	Par défaut
Comportement COM	Redémarrage par l'intermédiaire du Maître	Redémarrage déclenché par le Dispositif	Non	Redémarrage déclenché par le Dispositif si le paramètre COM actif est différent de la valeur par défaut	Le Dispositif s'arrête et désactive la communication jusqu'au prochain PowerCycle
Verrouillages des Accès	Non	Non	Par défaut	Par défaut	Par défaut
Transfert de Paramètre de bloc (voir la Figure 86)	–	Ignorer	Ignorer	Ignorer	Ignorer

Légende

"0"	Le paramètre numérique ou la liste des paramètres est réglé sur zéro
PowerCycle	Mise sous tension → hors tension → sous tension du Dispositif
Initial	Établissement des valeurs initiales en fonction de l'état de l'alimentation
COM	Communication
Non	Non affecté
Supprimer	Réglé sur "0" en cas de redémarrage COM. Tous les Événements actifs sont envoyés avec "Disparaître" pour supprimer DeviceStatus. Après l'exécution de "Restauration des réglages d'usine", les Événements en cours peuvent être renvoyés.
Par défaut	Rétablissement de la valeur initiale de l'état de livraison au client
Event	Chargement en amont du déclencheur au moyen du drapeau DS_UPLOAD_REQ
Ignorer	Paramètres transférés non activés
Enregistreur d'historique	Par exemple: Compteur horaire d'opérations

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.7.2 Réinitialisation du Dispositif

Cette fonction permet à un Dispositif de procéder à un "démarrage à chaud". Elle est particulièrement utile lorsqu'il est nécessaire de remettre le Dispositif à son état initial (mise sous tension, par exemple), ce qui signifie que la communication est interrompue.

Cette fonction est déclenchée dès la réception d'une SystemCommand "Réinitialisation du Dispositif" (voir le Tableau B.9). Dans tous les cas, la réponse de l'ISDU à cette SystemCommand doit être transmise immédiatement au Maître, ce qui permet de procéder à de nouvelles tentatives en cas de défauts de communication.

Cette fonction est facultative pour un Dispositif.

10.7.3 Réinitialisation de l'application

Cette fonction permet à un Dispositif de réinitialiser l'application spécifique à la technologie. Elle est particulièrement utile lorsqu'il est nécessaire de définir l'état opérationnel prédéfini d'une application spécifique à la technologie sans interruption de communication ni cycle d'arrêt. À l'inverse de "Restauration des réglages d'usine", seuls les paramètres spécifiques à l'application sont réinitialisés à "Par défaut". Chaque paramètre d'identification reste inchangé.

Cette fonction est déclenchée dès la réception d'une SystemCommand "Réinitialisation de l'application" (voir le Tableau B.9). Dans tous les cas, la réponse de l'ISDU à cette SystemCommand doit être transmise au Maître après l'exécution réussie de l'action demandée.

Cette fonction est fortement recommandée pour un Dispositif.

10.7.4 Restauration des réglages d'usine

Cette fonction permet à un Dispositif de rétablir les paramètres au statut de livraison d'origine. Elle est déclenchée dès la réception de la SystemCommand "Restauration des réglages d'usine" (voir le Tableau B.9). DS_UPLOAD_FLAG (voir le Tableau B.10) et les autres paramètres dynamiques tels que "ErrorCount" (voir B.2.19), "DeviceStatus" (voir B.2.20) et "DetailedDeviceStatus" (voir B.2.21) doivent être réinitialisés lorsque cette fonction est appliquée. Cela ne concerne pas les paramètres spécifiques au fournisseur tels que les compteurs d'heures de fonctionnement.

NOTE Dans ce cas, un jeu de paramètres stockés existant dans le Maître est automatiquement téléchargé dans le Dispositif après le redémarrage de communication suivant. Cela peut être évité à l'aide de la SystemCommand "Retour à la boîte" (voir 10.7.5).

Il revient au fournisseur du Dispositif de garantir le bon fonctionnement dans toutes les circonstances. Si un paramètre de communication (voir Paramètre Direct, page 1 du Tableau B.1) est modifié pendant cette restauration, la communication doit être arrêtée par le Dispositif pour déclencher une nouvelle communication à l'aide des paramètres de communication mis à jour. Dans ce cas, la réponse à cette SystemCommand doit être transmise au Maître après l'exécution réussie de l'action demandée, ce qui permet également de procéder à de nouvelles tentatives en cas de défauts de communication.

Cette fonction est facultative pour un Dispositif.

10.7.5 Retour à la boîte

Cette fonction permet à un Dispositif de restaurer les paramètres aux valeurs de livraison d'origine sans aucune interaction avec des mécanismes de niveau supérieur tels que le Stockage des Données ou le paramétrage PLC. Elle est particulièrement utile lorsqu'un Dispositif est retiré d'une installation déjà paramétrée, puis réactivé par exemple en tant que pièce détachée. Si le Dispositif demeure dans une application d'automatisation après le PowerCycle suivant, tout le paramétrage est écrasé comme s'il s'agissait d'un remplacement.

Cette fonction est déclenchée dès la réception de la SystemCommand "Retour à la boîte" (voir le Tableau B.9), c'est-à-dire que le Dispositif doit arrêter et désactiver la communication jusqu'au PowerCycle suivant. Si le Dispositif fournit une interface utilisateur, l'état particulier "En attente de PowerCycle" doit être visible. Dans tous les cas, la réponse à une SystemCommand doit être transmise au Maître, ce qui permet également de procéder à de nouvelles tentatives en cas de défauts de communication.

Cette fonction est obligatoire pour un Dispositif.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.8 Règles et contraintes de conception d'un Dispositif

10.8.1 Généralités

Outre les définitions du protocole sous forme d'états, de séquences, d'activités et de chronogrammes, un certain nombre de règles et de contraintes supplémentaires sont nécessaires pour définir le comportement des Dispositifs. Une vue d'ensemble des principales variables du protocole, réparties dans l'ensemble du présent document, est regroupée dans le Tableau 102, qui fournit également les références correspondantes.

10.8.2 Données de Processus

Le canal de communication du processus transmet de manière cyclique les Données de Processus sans d'éventuelles interférences des canaux de communication des Données à la Demande. L'échange de Données de Processus commence automatiquement dès que le Dispositif passe à l'état OPERATE par l'intermédiaire d'un message en provenance du Maître.

Le format de données transmises est spécifique au Dispositif et varie de zéro à 32 octets de données dans chaque sens de communication.

Recommandations:

– il convient que les structures de données conviennent à une utilisation par des applications d'Automate programmable;

– il est fortement recommandé de satisfaire aux règles données en F.3.3 et en [6].

Pour plus de précisions concernant l'indication de Données de Processus valides ou non valides, au moyen d'un drapeau PDValid, au cours de l'échange cyclique de données, voir A.1.5.

10.8.3 Perte de communication

Il incombe au concepteur du Dispositif de définir le comportement convenable du Dispositif en cas de perte de communication avec le Maître (la transition T10 à la Figure 44 gère la détection de la perte de communication et les actions du Dispositif qui en résultent sont définies en 10.2).

NOTE Ceci est notamment important pour des actionneurs tels que les vannes ou les systèmes de gestion de moteurs.

10.8.4 Paramètre Direct

La communication avec la page de Paramètres Directs ne comporte pas de mécanismes d'échange d'authentification (poignée de main) pour assurer une réception correcte ou la validité des paramètres transmis. Il n'est possible d'accéder à la page de Paramètres Directs que par un octet unique à la fois (Sous-index) ou par ensemble (16 octets). La cohérence de paramètres d'une taille supérieure à 1 octet ne peut pas être garantie.

Les paramètres de la page de Paramètres Directs ne peuvent pas être sauvegardés et restaurés au moyen du mécanisme de Stockage des Données.

10.8.5 Canal de communication des ISDU

Le canal de communication des ISDU est un moyen puissant de transmission de paramètres et de commandes (voir l'Article B.2).

Les règles suivantes doivent être prises en considération lorsque ce canal est utilisé (voir la Figure 7).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

– L'Index 0 n'est pas accessible au moyen du canal de communication des ISDU. L'Accès est redirigé par le Maître vers la page 1 de Paramètres Directs en utilisant le canal de communication de page.

– L'Index 1 n'est pas accessible au moyen du canal de communication des ISDU. L'Accès est redirigé par le Maître vers la page 2 de Paramètres Directs en utilisant le canal de communication de page.

– Un programme d'application d'Automate programmable ne peut pas accéder à l'Index 3. L'Accès est limité à l'application du Maître uniquement (Stockage des Données).

– Après réception d'une demande de ISDU en provenance du Maître, le Dispositif doit répondre dans un délai de 5 000 ms (voir le Tableau 102). Toute infraction à cette règle entraîne l'abandon de la tâche en cours par le Maître.

– Les paramètres avec l'attribut en écriture seulement (W) doivent être traités comme une SystemCommand. Seuls les types de données de base sont admis.

10.8.6 Règles de DeviceID relatives aux variantes de Dispositifs

Les Dispositifs ayant un certain DeviceID et un certain VendorID ne doivent pas déroger au comportement de communication et de fonctionnement. Ceci s'applique aux capteurs et aux actionneurs. Les différences entre ces Dispositifs peuvent être par exemple

– les longueurs de câblage,

– les matériaux des enveloppes,

– les mécanismes de montage,

– d'autres fonctionnalités et conditions environnementales.

10.8.7 Constantes de protocole

Le Tableau 102 donne une vue d'ensemble des principales constantes de protocole pour les Dispositifs.

Tableau 102 – Vue d'ensemble des constantes de protocole pour les Dispositifs

Variable Système	Références	Valeurs	Définition
Délai d'acquittement des ISDU, par exemple après une Commande Système	B.2.2	5 000 ms	Délai à compter de la réception d'une ISDU, par exemple une Commande Système et le début du message de réponse du Dispositif (voir la Figure 63)
Nombre maximal d'entrées dans la Liste des Index	B.2.3	70	Chaque entrée comporte un Index et un Sous-index. 70 entrées donnent au total 210 octets.
Des valeurs de préréglage pour des paramètres non utilisés ou réservés, par exemple FunctionID	Annexe B	0 (s'il s'agit de chiffres) 0x00 (s'il s'agit de caractères)	L'ingénierie doit mettre tous les paramètres non utilisés aux valeurs de préréglage.
Procédure de réactivation	7.3.2.2	Voir le Tableau 42 et le Tableau 43	Temporisations minimales et maximales et nombre de nouvelles tentatives
MaxRetry	7.3.3.3	2, voir le Tableau 46	Nombre maximal de nouvelles tentatives après des erreurs de communication
MinCycleTime	A.3.7 et B.1.3	Voir le Tableau A.11 et le Tableau B.3	Le Dispositif définit sa durée de cycle minimale pour l'acquisition de données de processus d'entrée ou de sortie.
Plage d'Index utilisable	Article B.2	Voir le Tableau B.8	Le présent document réserve certaines zones dans la plage totale de 65 535 index.
Erreurs et avertissements	10.10.2	50 ms	Un Événement de MODE "Apparition d'Événement" doit au moins persister pendant cette durée.
EventCount	8.2.2.11	1	Contrainte pour AL_Event.req

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.9 Descriptif de Dispositif d'E/S (IODD)

Un IODD (Descriptif de Dispositif d'E/S) est un fichier qui fournit l'ensemble des propriétés nécessaires à l'établissement de la communication ainsi que les paramètres et limites correspondantes nécessaires à l'établissement de la fonction souhaitée d'un capteur ou d'un actionneur.

Un IODD (Descriptif de Dispositif d'E/S) est un fichier qui décrit formellement un Dispositif.

Un fichier IODD doit être fourni pour chaque Dispositif et inclure toutes les informations nécessaires à l'appui du présent document.

*L'IODD peut être utilisé par les outils techniques d'ingénierie d'Automates programmables et/ou de Maîtres pour l'identification, la configuration, la définition des structures de données d'Échange de Données de Processus, le paramétrage et le décodage de diagnostic d'un Dispositif particulier.

NOTE Une description détaillée utilisée par le langage IODD pour décrire un Dispositif est donnée en [6].

10.10 Diagnostic d'un Dispositif

10.10.1 Concepts

À l'Article D.2, le présent document fournit uniquement les EventCodes (codes d'événements) les plus communs. L'objectif de cette information de diagnostic commune est de permettre à un opérateur ou à un agent de maintenance de mettre rapidement en œuvre des mesures correctives sans avoir une connaissance approfondie de la technologie du Dispositif. Ainsi, le texte associé à un Code d'Événement particulier doit toujours contenir une instruction de maintenance corrective accompagnant les informations de diagnostic.

Les passerelles de Maître de bus de terrain ont tendance à n'établir de correspondance que pour quelques EventCodes jusqu'au niveau de système supérieur. Généralement, les EventCodes spécifiques au fournisseur définis par l'IODD peuvent uniquement être décodés en instructions lisibles au moyen d'un Outil de configuration des Accès et Dispositifs (PDCT) ou un outil spécifique au fournisseur utilisant l'IODD.

Des informations condensées relatives à l'"état de santé" du Dispositif peuvent être récupérées à partir du paramètre "DeviceStatus" (voir B.2.20). Le Tableau 103 donne une vue d'ensemble des différentes possibilités offertes aux Dispositifs et présente des exemples de consommateurs de ces informations.

Si cette fonction est mise en œuvre, il est également possible de lire le nombre de défauts depuis la mise sous tension ou la réinitialisation au moyen du paramètre "ErrorCount" (voir B.2.19) et des informations supplémentaires dans le cas de profils de Dispositifs au moyen du paramètre "DetailedDeviceStatus" (voir B.2.21).

NOTE Les valeurs spécifiques aux profils pour "DetailedDeviceStatus" sont données en [7].

Si nécessaire, il est fortement recommandé de fournir des informations supplémentaires de diagnostic "plus approfondies" et spécifiques à la technologie, sous la forme de paramètres spécifiques au Dispositif (voir le Tableau B.8) qui peuvent être récupérés au moyen d'outils de configuration des Accès et Dispositifs pour des Maîtres ou au moyen d'outils spécifiques au fournisseur. Généralement, seuls des spécialistes ou le personnel d'entretien du fournisseur sont en mesure de tirer des conclusions à partir de ces informations.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 103 – Classification d'incidents de diagnostics de Dispositifs

Incident diagnostiqué	Apparition/ disparition	Occurrence unique	Paramètre	Destination	Consommateur
Erreur (mesure corrective rapide; Codes d'événements normalisés)	oui	–	–	Automate programmable ou IHM (Mise en correspondance du bus de terrain)	Personnel chargé de la maintenance et des réparations
Erreur (IODD: Codes d'événements spécifiques au fournisseur; voir le Tableau D.1)	oui	–	–	PDCT ou outillage du fournisseur	Personnel d'entretien du fournisseur
Erreur (au moyen de paramètres spécifiques au Dispositif)	–	–	Voir le Tableau B.8	PDCT ou outillage du fournisseur	Personnel d'entretien du fournisseur
Avertissement (mesure corrective rapide; Codes d'événements normalisés)	oui	–	–	Automate programmable ou IHM	Personnel chargé de la maintenance et des réparations
Avertissement (IODD: Codes d'événements spécifiques au fournisseur; voir le Tableau D.1)	oui	–	–	PDCT ou outillage du fournisseur	Personnel d'entretien du fournisseur
Avertissement (au moyen de paramètres spécifiques au Dispositif)	–	-	Voir le Tableau B.8		
Notification (Codes d'événements normalisés)	–	oui		PDCT	Personnel chargé de la mise en service
État détaillé du Dispositif	–	–		PDCT ou outillage du fournisseur	Personnel chargé de la mise en service et personnel d'entretien du fournisseur
Nombre de défauts au moyen du paramètre "ErrorCount"	–	–	Voir B.2.19		
"État de santé" du Dispositif au moyen du paramètre "DeviceStatus"	–	–	Voir B.2.20, le Tableau B.13	IHM, des outils tels que "Gestion d'actifs"	Opérateur

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

10.10.2 Événements

Les valeurs de MODE doivent être attribuées comme suit (voir A.6.4):

– les Événements de TYPE "Erreur" doivent utiliser les MODES "Apparition/Disparition d'Événement";

– les Événements de TYPE "Avertissement" doivent utiliser les MODES "Apparition/Disparition d'Événement";

– les Événements de TYPE "Notification" doivent utiliser le MODE "occurrence unique d'Événement".

Les exigences suivantes sont applicables:

– tous les Événements déjà placés dans la file d'attente des Événements sont ignorés par le Répartiteur d'Événements lorsque la communication est interrompue ou annulée. Dès la

reprise de la communication, l'application spécifique à la technologie est chargée de rendre correctement compte des causes de l'Événement en cours;

– il incombe au Répartiteur d'Événements de contrôler le flux d'"'Apparition d'Événement" et de "Disparition d'Événement". Dès lors que le Répartiteur d'Événements a transmis un Événement en MODE "Apparition d'Événement" pour un EventCode donné, il ne doit pas le transmettre une nouvelle fois pour le même EventCode, avant d'avoir transmis un Événement en MODE "Disparition d'Événement" pour ce même EventCode;
– chaque Événement doit utiliser un mode statique, le type et les attributs d'instance;
– chaque EventCode spécifique au fournisseur doit être attribué de manière unique à l'un des TYPES (Erreur, Avertissement ou Notification).

Pour prévenir une surcharge du canal de communication de diagnostic (voir la Figure 7), les exigences suivantes s'appliquent:

– les mêmes informations de diagnostic ne doivent pas être signalées à des intervalles de moins de 1 s. Cela signifie que le Répartiteur d'Événements ne doit pas invoquer le service AL_Event avec les mêmes EventCode et EventQualifier plus d'une fois par seconde. Cette mesure permet d'éviter les répétitions fréquentes des Événements;
– le Répartiteur d'Événements ne doit pas produire de MODE "Disparition d'Événement" dans un intervalle de moins de 50 ms après l'occurrence du MODE "Apparition d'Événement" correspondant;
– les occurrences ultérieures d'erreurs ou d'avertissements ayant la même cause première doivent être ignorées; c'est-à-dire, une cause première doit donner lieu à une seule indication d'erreur ou d'avertissement;
– le Répartiteur d'Événements doit invoquer le service AL_Event avec un EventCount égal à un;
– les erreurs sont prioritaires par rapport aux Avertissements.

La Figure 93 représente la manière dont deux erreurs successives sont traitées et le flux correspondant d'"Apparition/Disparition d'Événement" pour chaque erreur.

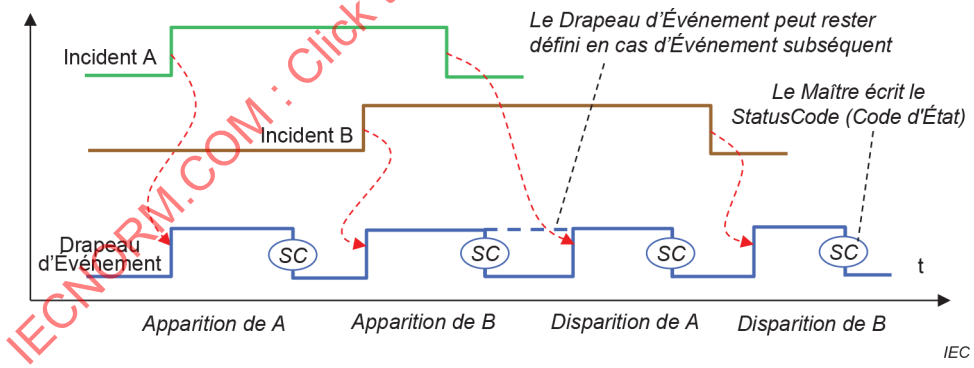


Figure 93 – Flux d'Événements en cas d'erreurs successives

10.10.3 Indicateurs visuels

L'indication de la communication SDCI sur le Dispositif est facultative. L'indication SDCI doit utiliser un indicateur de couleur verte. L'indication suit la temporisation et la spécification, comme cela est représenté à la Figure 94.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

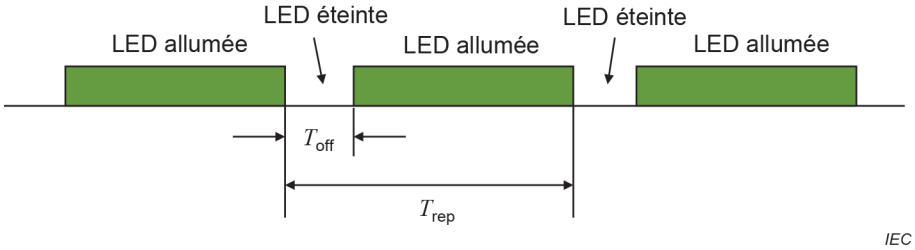


Figure 94 – Temporisation de l'indicateur à LED du Dispositif

Le Tableau 104 définit la temporisation de l'indicateur à LED des Dispositifs.

Tableau 104 – Temporisation pour des indicateurs à LED

Temporisation	Minimale	Type	Maximale	Unité
T_{rep}	750	1 000	1 250	ms
T_{off}	75	100	150	ms
T_{off}/T_{rep}	7,5	10	12,5	%
NOTE Les temporisations ci-dessus sont définies de sorte que la perception générale soit "dispositif sous tension".				

Une courte interruption périodique indique que le Dispositif est à l'état de communication COMx. Pour éviter le phénomène de clignotement, le cycle d'indication doit commencer par l'état "LED éteinte" et doit toujours s'achever (voir le Tableau 104).

10.11 Connectivité d'un Dispositif

Les différentes possibilités de connexion de Dispositifs aux Accès du Maître ainsi que les types de câbles et les codes de couleur correspondants, sont données en 5.5.

NOTE Pour des raisons de compatibilité, le présent document n'empêche pas l'utilisation de conducteurs supplémentaires sur des Dispositifs SDCI pour la connexion des fonctions non couvertes par le domaine d'application du présent document (par exemple, pour le transfert de signaux de sortie analogiques).

11 Maître

11.1 Vue d'ensemble

11.1.1 Positionnement d'un Maître et des Applications de Passerelle

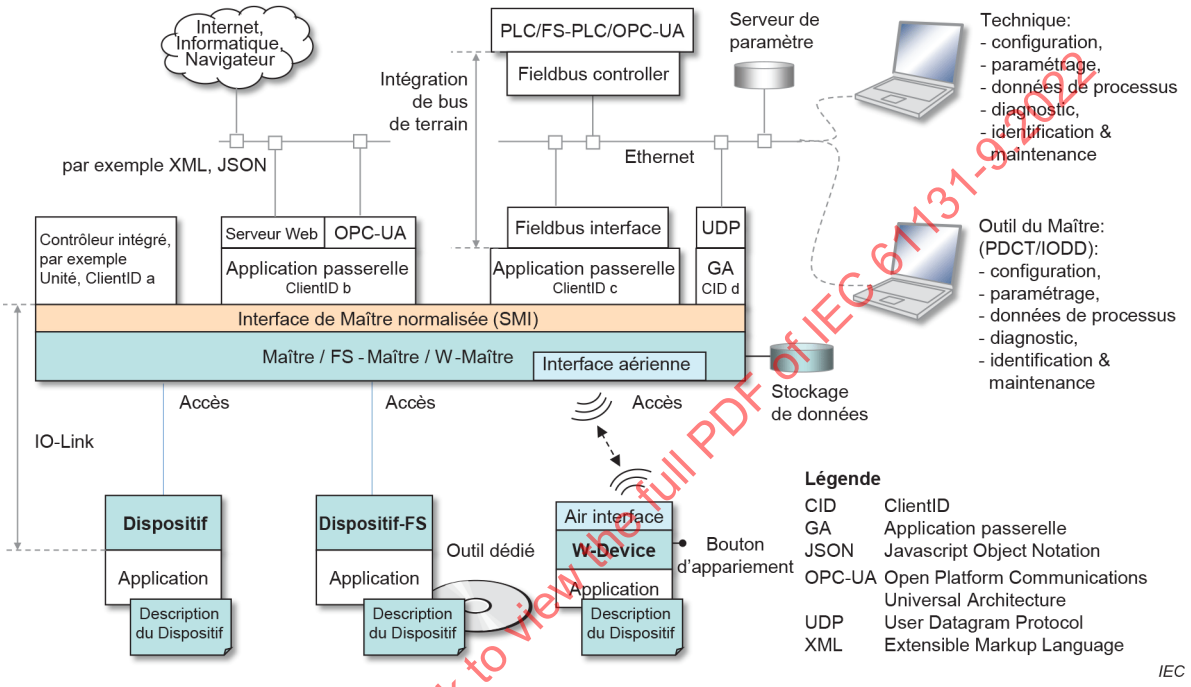
Le domaine de la technologie SDCI dans le cadre de la hiérarchie de l'automatisation a déjà été présenté en 4.2. La Figure 95 représente la relation recommandée entre technologie SDCI et technologie de bus de terrain. Même s'il peut s'agir du cas d'utilisation le plus important dans la pratique, ceci n'implique pas automatiquement une dépendance de la technologie SDCI de son intégration dans les systèmes de bus de terrain. L'interface SDCI peut également être directement intégrée dans les systèmes d'Automate programmable, dans les ordinateurs industriels ou dans d'autres systèmes d'automatisation qui ne sont pas reliés entre eux par une communication de bus de terrain.

Pour assurer le comportement de préférence uniforme des Maîtres, la Figure 95 représente une interface normalisée du Maître (SMI) en tant que couche entre le Maître et les Applications de Passerelle ou les systèmes intégrés en haut. Cette interface normalisée du Maître sert également les extensions du système de sécurité et les extensions du système sans fil. Dans les cas de FS-Masters, il ne faut pas oublier que cette SMI exige, à certains égards, la mise en œuvre selon les normes de sécurité.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

L'interface normalisée du Maître est spécifiée en 11.2 au moyen de services et d'objets de données similaires aux autres couches (PL, DL et AL) dans le présent document. Elle est conçue à l'aide de structures de base uniformes que le bus de terrain de la couche supérieure et les systèmes informatiques de la couche supérieure peuvent utiliser d'une manière différente: push ("écriture"), pull ("lecture"), push/pull ("écriture/lecture") et indication ("Événement").

La spécification des Applications de Passerelle ne relève pas du domaine d'application du présent document. Les concepteurs doivent respecter les exigences en temps réel des fonctions de commande et des fonctions de sécurité dans le cas des Applications de Passerelle (voir 13.2).



NOTE Les zones bleutées et orangées indiquent les fonctionnalités spécifiées dans le présent document, sauf celles relatives à la sécurité fonctionnelle (FS) et sans fil (W).

Figure 95 – Relation générique entre technologie SDCI et technologie d'automatisation

11.1.2 Structure, applications et services d'un Maître

La Figure 96 donne une vue d'ensemble de la structure et des services d'un Maître.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

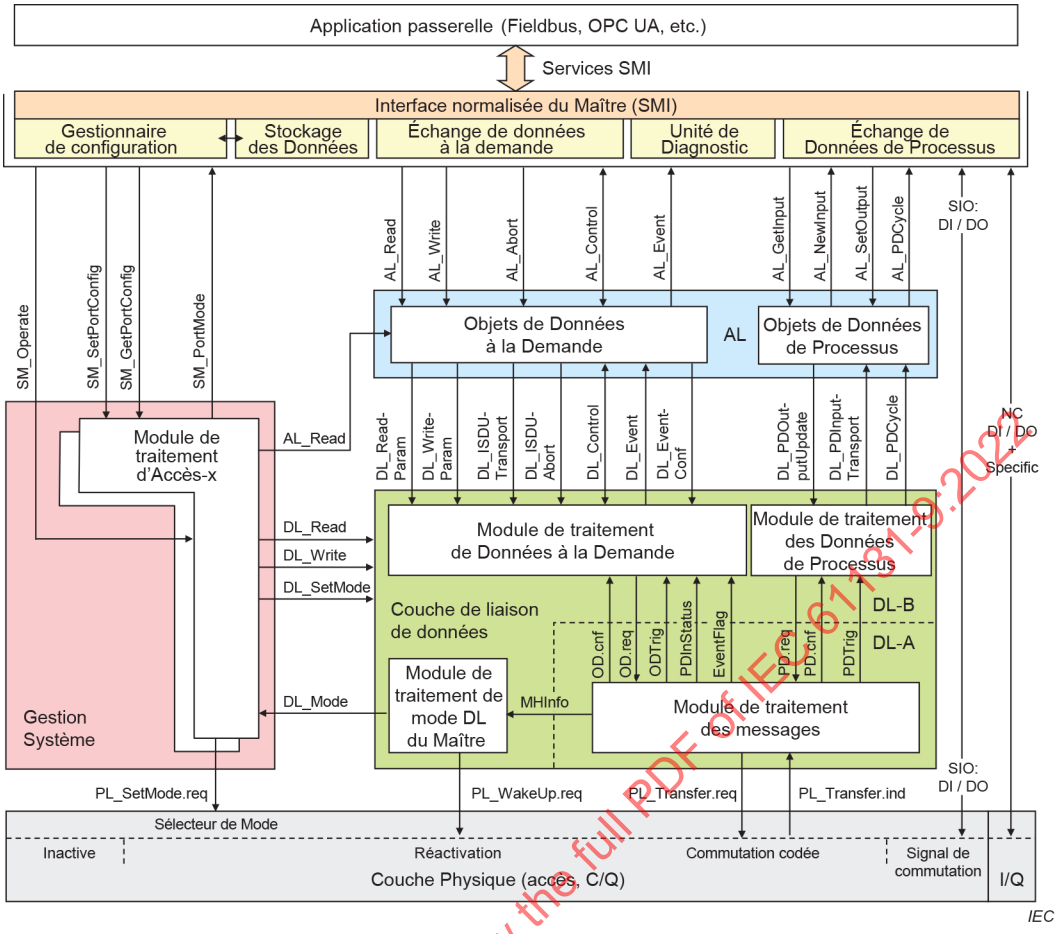


Figure 96 – Structure, applications et services d'un Maître

Les applications de Maître se trouvent au sommet de la structure de Maître et sont composées:

– du Gestionnaire de Configuration (CM), qui transforme les attributions de configuration de l'utilisateur en réglages d'Accès;

– de l'Échange de Données à la Demande (ODE), qui fournit par exemple l'Accès acyclique aux paramètres;

– du mécanisme de Stockage des Données (DS), qui peut être utilisé pour sauvegarder et restaurer les paramètres d'un Dispositif;

– de l'Unité de Diagnostic (DU), qui achemine les Événements de la couche AL à l'unité de Stockage des Données ou à l'application passerelle;

– de l'Échange de Données de Processus (PDE), qui constitue le pont de liaison avec les instruments d'automatisation de niveau supérieur.

Elles sont accessibles par les applications de passerelle (et d'autres) par l'intermédiaire de l'interface normalisée du Maître (SMI) et ses services et méthodes.

Ces services et les fonctions correspondantes sont spécifiés de manière abstraite de 11.2.2 à 11.2.22 et à l'Annexe E.

Les applications de Maître sont décrites en détail de 11.3 à 11.7. Le Gestionnaire de configuration (CM) et le mécanisme de Stockage de Données (DS) exigent une coordination particulière eu égard aux Données à la Demande.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.1.3 Vue d'objet d'un Maître et ses Accès

La Figure 97 représente le modèle d'objet de données du Maître et de ses Accès d'un point de vue d'une SMI

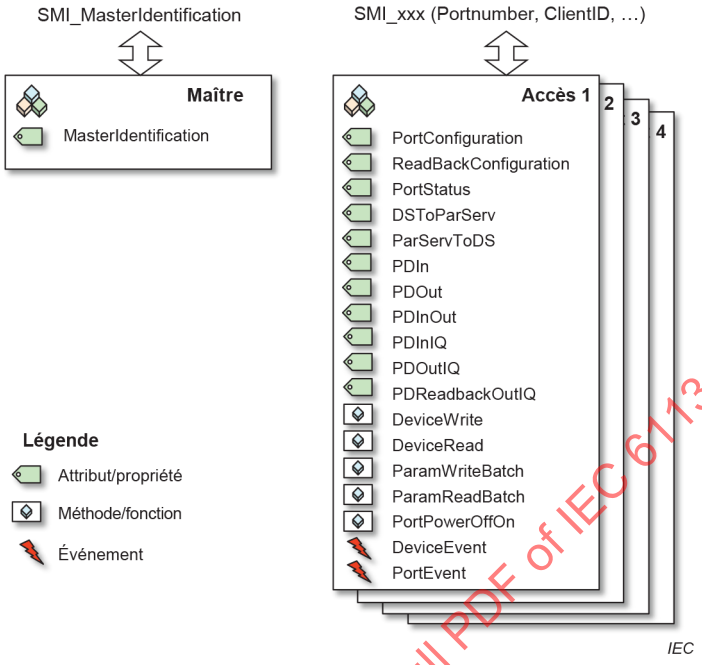


Figure 97 – Modèle d'objet du Maître et des Accès

Chaque objet est accompagné d'attributs et de méthodes auxquels les services SMI peuvent accéder. Tous sont spécifiés en 11.2.

11.2 Services de l'interface normalisée du Maître (SMI)

11.2.1 Vue d'ensemble

La Figure 98 représente les services SMI individuels disponibles, par exemple, pour la passerelle.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

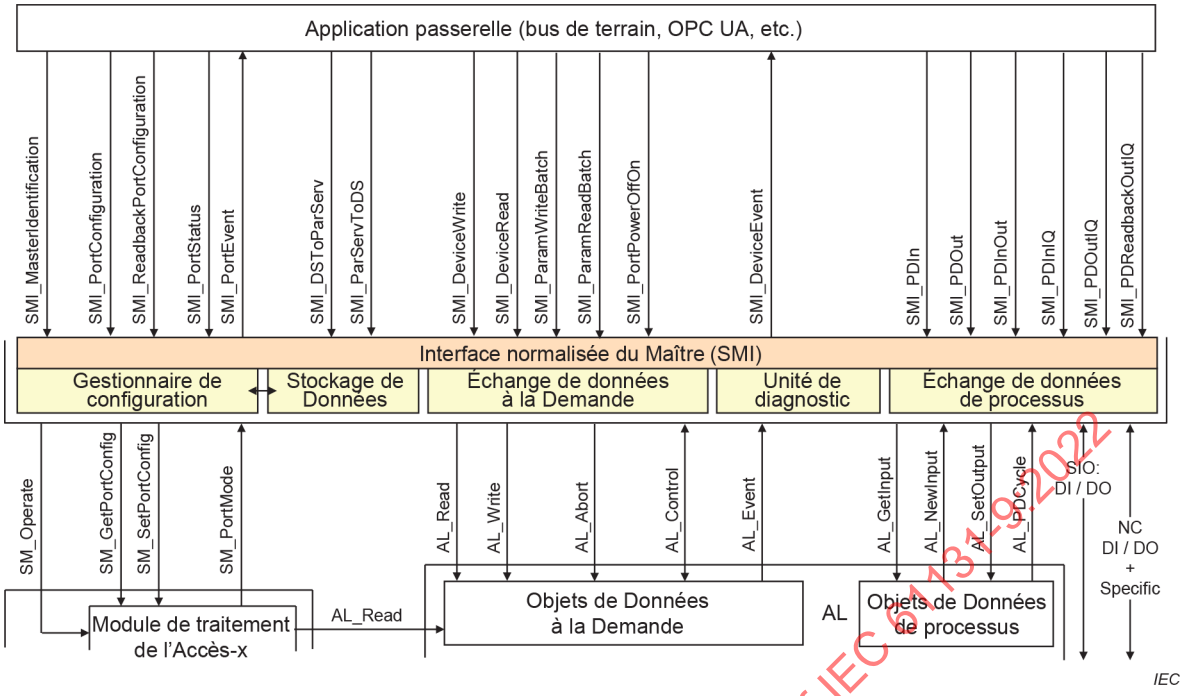


Figure 98 – Services SMI

Les interfaces de communication (Fieldbus, OPC UA, JSON, UDP ou analogues, par exemple) sont chargées de fournir l'Accès aux services SMI. Il est obligatoire que les systèmes de communication de niveau supérieur se réfèrent aux définitions SMI dans leurs adaptations. La fonctionnalité derrière la SMI est obligatoire, sauf si elle est spécifiquement déclarée comme étant facultative.

Le Tableau 105 répertorie les services SMI disponibles aux applications passerelles ou autres clients.

Tableau 105 – Services SMI

Désignation du service	Maître	O/F/C	Objet
SMI_MasterIdentification	R	O	Service universel d'identification d'un Maître
SMI_PortConfiguration	R	O	Configuration de l'Accès
SMI_ReadbackPortConfiguration	R	O	Extraction de la configuration d'Accès actuelle
SMI_PortStatus	R	O	Extraction de l'état de l'Accès
SMI_DSToParServ	R	O	Transfert du Stockage de Données vers le serveur de paramètres
SMI_ParServToDS	R	O	Transfert du serveur de paramètres vers le Stockage de Données
SMI_DeviceWrite	R	O	Transport de l'ISDU vers le Dispositif
SMI_DeviceRead	R	O	Transport de l'ISDU depuis le Dispositif
SMI_ParamWriteBatch	R	F	Transport de l'ISDU par lot des paramètres (écriture)
SMI_ParamReadBatch	R	F	Transport de l'ISDU par lot des paramètres (lecture)
SMI_PortPowerOffOn	R	F	PortPowerOffOn
SMI_DeviceEvent	I	O	Service "Push" universel pour les Événements de Dispositif
SMI_PortEvent	I	O	Service "Push" universel pour les Événements d'Accès
SMI_PDIn	R	O	Extraction de PD depuis InBuffer
SMI_PDOut	R	O	Établissement de PD dans OutBuffer

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Désignation du service	Maître	O/F/C	Objet
SMI_PDInOut	R	O	Extraction InBuffer et OutBuffer
SMI_PDInIQ	R	C	Données de processus en entrée au niveau de I/Q (Broche 2 sur M12)
SMI_PDOutIQ	R	C	Données de processus en sortie au niveau de I/Q (Broche 2 sur M12)
SMI_PDReadbackOutIQ	R	C	Extraction des données de processus en sortie au niveau de I/Q (Broche 2 sur M12)

Légende

I	Initiateur du service	R	Destinataire (Répondeur) du service		
O	Obligatoire	F	Facultatif	C	Conditionnel

11.2.2 Structure des Arguments de service SMI

Les arguments de service SMI contiennent une structure fixe d'éléments normalisés, qui sont caractérisés comme suit.

ClientID

Les Applications de Passerelle peuvent utiliser les services SMI en même temps que les clients de la SMI (voir 11.2.3). Par conséquent, les services SMI attribuent un ClientID unique à chaque client individuel. Il relève de la responsabilité des Applications de Passerelle de coordonner ces activités de service SMI et d'acheminer les réponses vers le client appelant. Le nombre maximal de clients simultanés est spécifique au Maître.

Type de Données: Unsigned8

Valeurs admises: 1 pour le nombre maximal de clients simultanés spécifique au fournisseur. "0" est uniquement utilisé pour les besoins de la diffusion en cas d'indications, voir 11.2.15 et 11.2.16.

PortNumber

Chaque service SMI contient le numéro d'Accès dans le cas d'un objet d'Accès adressé (job) ou d'objet d'Accès déclenché (événement).

Type de Données: Unsigned8

Valeurs admises: 1 à MaxNumberOfPorts. "0" est uniquement utilisé pour adresser l'ensemble du Maître (voir 11.2.4).

ExpArgBlockID

Cet élément spécifie l'ArgBlockID prévu pour transporter les données de réponse d'une demande de service. Les ID sont définis dans le Tableau E.1.

Type de Données: Unsigned16.

Valeurs admises: 1 à 65 535.

RefArgBlockID

Dans les résultats, cet élément spécifie l'ID de l'Argblock envoyé par la demande de service. Les ID sont définis dans le Tableau E.1.

Type de Données: Unsigned16.

Valeurs admises: 1 à 65 535.

ArgBlockLength

Cet élément spécifie la longueur totale de l'ArgBlock qui suit. Les extensions spécifiques au fournisseur ne sont pas admises.

Type de Données: Unsigned16.

Valeurs admises: 2 à 65 535.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlock

Tous les services SMI contiennent un ArgBlock caractérisé par un ArgBlockID et sa description. Les résultats du service fournissent l'ArgBlock associé à l'ExpArgBlockID, qui fait partie intégrante de cet ArgBlock. La longueur variable possible de l'ArgBlock est prédéfinie par la définition dans le présent document.

Les paires d'ExpArgBlock/RefArgBlock et d'ArgBlockID à l'intérieur d'une structure SMI doivent être uniques. Le codage détaillé des ArgBlocks est spécifié à l'Annexe E. Les types d'ArgBlock et leurs ArgBlockID sont définis dans le Tableau E.1. Les erreurs de service figurent dans chaque service individuel et à l'Article C.4.

11.2.3 Simultanéité et hiérarchisation des services SMI

Les règles suivantes s'appliquent pour la simultanéité des services SMI lors de l'Accès aux attributs:

– tous les services SMI avec un PortNumber différent accèdent à différents objets d'Accès (opérations disjointes);
– différents services SMI utilisant le même PortNumber accèdent à des attributs/méthodes différent(e)s d'un objet d'Accès (opérations simultanées);
– des services SMI identiques utilisant le même PortNumber et différents ClientID accèdent simultanément à des attributs identiques (cohérence);

Les règles suivantes s'appliquent pour les services SMI lors de l'Accès aux méthodes:

– les services SMI pour des méthodes utilisant des PortNumbers différents accèdent à différents objets d'Accès (opérations disjointes);
– les services SMI pour des méthodes utilisant le même PortNumber et différents ClientID créent des instances "job" et sont traités dans l'ordre de leur arrivée (n Clients simultanément);
– SMI_ParamWriteBatch (ArgBlock "DeviceBatch") doit être traité comme une instance "job" qui ne doit pas être interrompue par un service SMI_DeviceWrite ou SMI_DeviceRead service.

Les services SMI à l'intérieur de l'interface normalisée du Maître ne sont pas hiérarchisés. Tous les services qui accèdent aux méthodes sont traités dans l'ordre de leur arrivée (premier entré, premier service).

11.2.4 SMI_MasterIdentification

Jusqu'à présent, une identification explicite d'un Maître n'avait pas la priorité dans la SDCI étant donné que les applications de passerelle utilisaient généralement des informations d'identification et de maintenance figées dans le code, comme cela était exigé dans le système de bus de terrain. En raison de l'exigence "un Outil Maître (PCDT) s'adapte à différentes marques de Maître", les nouveaux Outils Maîtres correspondants doivent être en mesure de se connecter aux Maîtres en fournissant une SMI. À cet effet, le service SMI_MasterIdentification a été créé. Il permet aux Outils Maîtres de s'adapter aux marques et types de Maître individuel, si une passerelle de bus de terrain particulière fournit les services SMI dans un codage accessible uniforme (voir l'Article 13). Une classe de Maîtres avec un certain MasterID et un certain VendorID ne doivent pas déroger au comportement de communication et de fonctionnement (identification du type de Maître). Le Tableau 106 présente le service SMI_MasterIdentification.

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 106 – SMI_MasterIdentification

Nom du paramètre		.req	.cnf
Argument		M	
ClientID		M	
PortNumber	(0x00)	M	
ExpArgBlockID	(par exemple, 0x0001)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Résultat (+)			S
ClientID			M
PortNumber	(0x00)		M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber	(0x00)		M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

Ce paramètre contient un Accès virtuel adressant l'ensemble de l'unité Maître (0x00).

ExpArgBlockID

Ce paramètre contient un ArgBlockID de la famille MasterIdent, par exemple 0x0001 (voir le Tableau E.1).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "VoidBlock".

ArgBlock

Ce paramètre contient l'ArgBlock "VoidBlock" (0xFFF0, voir l'Article E.17).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir le Tableau E.2).

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé (voir le Tableau C.3):
ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID (longueur d'ArgBlock incorrecte).

11.2.5 SMI_PortConfiguration

À l'aide de ce service, un client SMI (une application de passerelle, par exemple) lance l'Accès de Maître indiqué et le Dispositif connecté en utilisant les éléments du paramètre PortConfigList. Le service doit être accepté immédiatement et exécuté sans délai. Le contenu du Stockage de Données de cet Accès est supprimé à chaque nouvelle configuration d'Accès "DS_Delete" (voir la Figure 99). Le Tableau 107 présente la structure du service. En règle générale, ArgBlock est différent dans les extensions SDCI (sécurité et sans fil, par exemple) et est spécifié ici (voir [10] et [11]).

Tableau 107 – SMI_PortConfiguration

Nom du paramètre		.req	.cnf
Argument		M	
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(par exemple, 0x8000)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x8000)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x8000)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID "VoidBlock" (0xFFF0, voir l'Article E.17).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant à "pousser".

ArgBlock

Ce paramètre contient un ArgBlock de la famille PortConfigList, par exemple 0x8000 (voir le Tableau E.1).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x8000).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (0xFFF0) .

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x8000).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:
PORT_NUM_INVALID (numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID (longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT (type de contenu ArgBlock incorrect).

11.2.6 SMI_ReadbackPortConfiguration

Ce service permet d'extraire la configuration effective de l'Accès Maître indiqué. Le Tableau 108 présente la structure du service. En règle générale, ce service est différent dans les extensions SDCI (sécurité et sans fil, par exemple) (voir [10] et [11]).

Tableau 108 – SMI_ReadbackPortConfiguration

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(par exemple, 0x8000)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient un ArgBlockID de la famille PortConfigList, par exemple 0x8000 (voir le Tableau E.1).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "VoidBlock".

ArgBlock

Ce paramètre contient l'ArgBlock "VoidBlock" (0xFFF0, voir l'Article E.17).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir l'Article E.3).

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:

PORT_NUM_INVALID	(numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID	(longueur ArgBlock incorrecte).

11.2.7 SMI_PortStatus

Ce service permet d'extraire l'état effectif de l'Accès Maître indiqué. Le Tableau 109 présente la structure du service. En règle générale, ce service est différent dans les extensions SDCI (sécurité et sans fil, par exemple) (voir [10] et [11]).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 109 – SMI_PortStatus

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(par exemple, 0x9000)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient un ArgBlockID de la famille PortStatusList, par exemple 0x9000 (voir le Tableau E.1).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "VoidBlock".

ArgBlock

Ce paramètre contient l'ArgBlock "VoidBlock" (0xFFF0, voir l'Article E.17).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir l'Article E.4).

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:
PORT_NUM_INVALID (numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID (longueur ArgBlock incorrecte).

11.2.8 SMI_DSToParServ

À l'aide de ce service, un client SMI (une application de passerelle, par exemple) est en mesure d'extraire le jeu de paramètres de technologie d'un Dispositif du Stockage de Données, puis de le renvoyer dans un serveur de paramètres de niveau supérieur (voir la Figure 95, 11.4 et 13.4.2). Le Tableau 110 présente la structure du service.

Dans le cas de DI ou DO sur cet Accès, le contenu du Stockage de Données est supprimé. Cela s'applique également si le Stockage de Données n'est pas activé sur cet Accès.

Tableau 110 – SMI_DSToParServ

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(0x7000)	M	
ArgBlockLength		M	
ArgBlock	(VoidBlock: 0xFFF0)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xFFF0)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID 0x7000 (voir le Tableau E.1).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "VoidBlock".

ArgBlock

Ce paramètre contient l'ArgBlock "VoidBlock" (0xFFF0, voir l'Article E.17).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir l'Article E.6).

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xFFF0).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:

PORT_NUM_INVALID	(numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID	(longueur ArgBlock incorrecte).

11.2.9 SMI_ParServToDS

À l'aide de ce service, un client SMI (une application de passerelle, par exemple) est en mesure de restaurer le jeu de paramètres de technologie d'un Dispositif à l'intérieur du Stockage de Données à partir du serveur de paramètres de niveau supérieur (voir la Figure 95, 11.4 et 13.4.2). Le Tableau 111 présente la structure du service.

Si le Stockage de Données n'est pas pris en charge ou n'est pas activé sur cet Accès, le service est appliqué avec Résultat(-) INCONSISTENT_DS_DATA. Cela s'applique également si le Stockage de Données n'est pas cohérent avec la configuration de l'Accès (VendorID ne correspond pas, par exemple).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 111 – SMI_ParServToDS

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(0x7000)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x7000)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x7000)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID "VoidBlock" (0xFFF0, voir l'Article E.17).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant à "pousser".

ArgBlock

Ce paramètre contient l'ArgBlock DS_Data((0x7000, voir le Tableau E.1).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7000).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID.

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7000).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:
PORT_NUM_INVALID (numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID (longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT (type de contenu ArgBlock incorrect),
INCONSISTENT_DS_DATA (données du Stockage de Données incohérentes).

11.2.10 SMI_DeviceWrite

Ce service permet d'écrire les Données à la Demande (OD) pour la propagation au Dispositif. Le Tableau 112 présente la structure du service.

Tableau 112 – SMI_DeviceWrite

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(0x3000)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x3000)		M
ArgBlockLength			M
ArgBlock	(associé à l'ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x3000)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID "VoidBlock" (0xFFF0, voir l'Article E.17).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant à "pousser".

ArgBlock

Ce paramètre contient l'ArgBlock "On-requestData" (0x3000, voir le Tableau E.1).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x3000).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID.

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x3000).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:

PORT_NUM_INVALID	(numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID	(longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT	(type de contenu ArgBlock incorrect),
SERVICE_TEMP_UNAVAILABLE	(Maître occupé),
DEVICE_NOT_ACCESSIBLE	(Dispositif ne communique pas) ,
Device ErrorType	(Voir les Articles C.2 et C.3).

11.2.11 SMI_DeviceRead

Ce service permet de lire les Données à la Demande (OD) depuis le Dispositif par l'intermédiaire du Maître. Le Tableau 113 présente la structure du service.

Tableau 113 – SMI_DeviceRead

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(0x3000)	M	
ArgBlockLength		M	
ArgBlock	("On-requestData/Index": 0x3001)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x3001)		M
ArgBlockLength			M
ArgBlock	(associé à ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x3001)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ExpArgBlockID

Ce paramètre contient l'ArgBlockID de "On-requestData" (0x3000, voir le Tableau E.1).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock "On-requestData/Index" (0x3001, voir l'Article E.5).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x3001).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir le Tableau E.5).

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x3001).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:

PORT_NUM_INVALID	(numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID	(longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT	(type de contenu ArgBlock incorrect),
SERVICE_TEMP_UNAVAILABLE	(Maître occupé),
DEVICE_NOT_ACCESSIBLE	(Dispositif ne communique pas) ,
DeviceErrorType	(Voir les Articles C.2 et C.3).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

11.2.12 SMI_ParamWriteBatch

Ce service permet le transfert "push" d'un grand nombre d'objets de Dispositif cohérents par l'intermédiaire de plusieurs ISDU. Le Tableau 114 présente la structure du service. Les règles suivantes sont applicables:

– le service transfère l'ArgBlock "DeviceParBatch" au Maître qui achemine le contenu objet par objet au Dispositif par l'intermédiaire de AL_Write (ISDU);

– la même structure ArgBlock est renvoyée en tant que Résultat (+). Toutefois, une valeur "0x0000" indique le succès d'un AL_Write particulier ou un ISDU ErrorType d'un AL_Write qui n'a pas abouti au lieu d'un enregistrement de paramètre;

– résultat (−) est uniquement renvoyé en cas de défaillance d'un service par l'intermédiaire de "JobError".

NOTE 1 Ce service prend pour hypothèse l'utilisation d'un Paramétrage de Bloc et de ressources tampons suffisantes.

NOTE 2 Ce service peut avoir une durée plus longue imprévue.

Ce service est facultatif. La disponibilité est indiquée au moyen de l'identification du Maître (voir le Tableau E.2).

Tableau 114 – SMI_ParamWriteBatch

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	("DeviceParBatch": 0x7001)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x7001)		M
ArgBlockLength			M
ArgBlock	(associé à l'ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x7001)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID "VoidBlock" (0xFFF0, voir l'Article E.17).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant à "pousser".

ArgBlock

Ce paramètre contient l'ArgBlock "DeviceParBatch" (0x7001, voir le Tableau E.1).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7001).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir le Tableau E.7).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7001).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:

SERVICE_NOT_SUPPORTED	(Service inconnu),
PORT_NUM_INVALID	(numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED	(ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID	(longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT	(type de contenu ArgBlock incorrect),
MEMORY_OVERRUN	(mémoire insuffisante),
SERVICE_TEMP_UNAVAILABLE	(Maître occupé),
DEVICE_NOT_ACCESSIBLE	(Dispositif ne communique pas).

11.2.13 SMI_ParamReadBatch

Ce service permet le transfert "pull" d'un grand nombre de paramètres de Dispositif cohérents par l'intermédiaire de plusieurs ISDU. Le Tableau 114 présente la structure du service. Les règles suivantes sont applicables:

– le service transfère l'ArgBlock "IndexList" vers le Maître qui transforme le contenu entrée par entrée dans AL_Read (ISDU) vers le Dispositif;

– l'ArgBlock correspondant "DeviceParBatch" est renvoyé en tant que Résultat (+). En cas d'AL_Read réussi d'un objet, l'enregistrement de paramètre correspondant ou un ISDU ErrorType d'un AL_Read qui n'a pas abouti au lieu d'un enregistrement de paramètre est renvoyé;

– résultat (−) est uniquement renvoyé en cas de défaillance d'un service par l'intermédiaire de "JobError".

NOTE 1 Ce service prend pour hypothèse l'utilisation d'un Paramétrage de Bloc et de ressources tampons suffisantes.

NOTE 2 Ce service peut avoir une durée plus longue imprévue.

Ce service est facultatif. La disponibilité est indiquée au moyen de l'identification du Maître (voir le Tableau E.2)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Tableau 115 – SMI_ParamReadBatch

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	("DeviceParBatch": 0x7001)	M	
ArgBlockLength		M	
ArgBlock	("IndexList": 0x7002)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x7002)		M
ArgBlockLength			M
ArgBlock	(associé à l'ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0x7002)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID de "DeviceParBatch" (0x7001, voir le Tableau E.1).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "IndexList".

ArgBlock

Ce paramètre contient l'ArgBlock "IndexList" (0x7002, voir le Tableau E.1).

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7002).

ArgBlockLength

Ce paramètre contient la longueur conditionnelle de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir le Tableau E.7).

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7002)

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError"

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18)

Valeurs admises dans l'ordre hiérarchisé:
SERVICE_NOT_SUPPORTED (Service inconnu),
PORT_NUM_INVALID (numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID (longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT (type de contenu ArgBlock incorrect),
MEMORY_OVERRUN (mémoire insuffisante),
SERVICE_TEMP_UNAVAILABLE (Maître occupé),
DEVICE_NOT_ACCESSIBLE (Dispositif ne communique pas) .

11.2.14 SMI_PortPowerOffOn

Ce service permet d'activer ou de désactiver Power 1 d'un Accès particulier (voir 5.4.1). Il renvoie dans le délai écoulé indiqué dans l'ArgBlock. Le Tableau 116 présente la structure du service.

Tableau 116 – SMI_PortPowerOffOn

Nom du paramètre		.req	.cnf
Argument			
ClientID		M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	("PortPowerOffOn": 0x7003)	M	
Résultat (+)			S
ClientID			M
PortNumber			M
ExpArgBlockID	(ID de demande ArgBlock 0x7003)		M
ArgBlockLength			M
ArgBlock	(associé à l'ExpArgBlockID)		M
Résultat (−)			S
ClientID			M
PortNumber			M
ExpArgBlockID	(ID de demande ArgBlock 0x7003)		M
ArgBlockLength			M
ArgBlock	(JobError: 0xFFFF)		M

Argument

Pour la demande de service, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID "VoidBlock" (0xFFF0, voir l'Article E.17).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant à "pousser".

ArgBlock

Ce paramètre contient l'ArgBlock "PortPowerOffOn" (0x7003, voir le Tableau E.1).

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Résultat (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7003).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (0xFFF0) .

Résultat (−):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0x7003).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir l'Article E.18).

Valeurs admises dans l'ordre hiérarchisé:
PORT_NUM_INVALID (numéro d'Accès incorrect),
ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu),
ARGBLOCK_LENGTH_INVALID (longueur ArgBlock incorrecte),
ARGBLOCK_INCONSISTENT (type de contenu ArgBlock incorrect),
SERVICE_TEMP_UNAVAILABLE (Maître occupé).

11.2.15 SMI_DeviceEvent

Ce service permet de signaler un Événement de Maître créé par le Dispositif. Le Tableau 117 présente la structure du service.

Tableau 117 – SMI_DeviceEvent

Nom du paramètre		.ind	.rsp
Argument			
ClientID	(= "0" → Broadcast)	M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	("DeviceEvent": 0xA000)	M	
Acquittement			S
ClientID	(= "0")		M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xA000)		M
ArgBlockLength			M
ArgBlock	(VoidBlock: 0xFFF0)		M

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022

Argument

Pour cette indication, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

Pour cette indication, le ClientID doit être "0" ("broadcast" au système de niveau supérieur).

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID "VoidBlock" (0xFFF0, voir l'Article E.17).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock reporté 0xA000.

ArgBlock

Ce paramètre contient l'ArgBlock "DeviceEvent" (0xA000, voir le Tableau E.1).

Acquittement

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

Le ClientID doit être "0".

PortNumber

RefArgBlockID

Ce paramètre contient en référence l'ID de l'ArgBlock envoyé par la demande (0xA000).

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant.

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (0xFFF0) .

11.2.16 SMI_PortEvent

Ce service permet de signaler un Événement de Maître créé par l'Accès. Le Tableau 118 présente la structure du service.

Tableau 118 – SMI_PortEvent

Nom du paramètre		.ind	.rsp
Argument			
ClientID	(= "0" → Broadcast)	M	
PortNumber		M	
ExpArgBlockID	(VoidBlock: 0xFFF0)	M	
ArgBlockLength		M	
ArgBlock	(PortEvent: 0xA001)	M	
Acquittement			S
ClientID	(= "0")		M
PortNumber			M
RefArgBlockID	(ID de demande ArgBlock 0xA001)		M
ArgBlockLength			M
ArgBlock	(VoidBlock: 0xFFF0)		M

Argument

Pour cette indication, les paramètres spécifiques sont transmis dans l'argument (voir 11.2.2).

ClientID

Pour cette indication, le ClientID doit être "0" ("broadcast" au système de niveau supérieur).

PortNumber

IECNORM.COM : Click to view the full PDF of IEC 61131-9:2022