

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Industrial networks – Single-drop digital communication interface –
Part 2: Functional safety extensions**

**Réseaux industriels – Interface de communication numérique point à point –
Partie 2: Extensions de sécurité fonctionnelle**

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022



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 Varembe
CH-1211 Geneva 20
Switzerland

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

About the IEC

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

About IEC publications

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

IEC publications search - webstore.iec.ch/advsearchform

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 It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

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

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

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

IEC Products & Services Portal - products.iec.ch

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

Electropedia - www.electropedia.org

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

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.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Industrial networks – Single-drop digital communication interface –
Part 2: Functional safety extensions**

**Réseaux industriels – Interface de communication numérique point à point –
Partie 2: Extensions de sécurité fonctionnelle**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040

ISBN 978-2-8322-3946-9

**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.**

CONTENTS

FOREWORD.....	12
INTRODUCTION.....	14
1 Scope.....	17
2 Normative references	17
3 Terms, definitions, symbols, abbreviated terms, and conventions	18
3.1 Terms and definitions.....	18
3.2 Common terms and definitions	18
3.3 Terms and definitions related to SDCI-FS	21
3.4 Symbols and abbreviated terms	23
3.5 Conventions.....	24
3.5.1 Behavioral descriptions.....	24
3.5.2 Memory and transmission octet order	25
4 Overview of SDCI-FS	25
4.1 Purpose of the technology and feature levels.....	25
4.1.1 Base SDCI-FS technology	25
4.1.2 From "analog" and "switching" to communication.....	26
4.1.3 Minimized paradigm shift from FS-DI to FS-Master	27
4.1.4 Following the SDCI paradigm (SIO vs. OSSDe)	28
4.1.5 Port class B	30
4.1.6 "USB-Master" with safety parameterization.....	31
4.1.7 Interoperability matrix of safety devices	31
4.2 Positioning within the automation hierarchy	32
4.3 Wiring, connectors, and power supply	33
4.4 Relationship to SDCI.....	33
4.5 Communication features and interfaces	34
4.6 Parameterization.....	34
4.7 Role of FS-Master and FS-Gateway.....	35
4.8 Mapping to upper-level systems.....	35
4.9 Structure of the document.....	35
5 Extensions to the Physical Layer (PL).....	35
5.1 Overview.....	35
5.2 Extensions to PL services	36
5.2.1 PL_SetMode.....	36
5.2.2 PL_Ready.....	36
5.3 Transmitter/receiver.....	37
5.3.1 Assumptions for the expansion to OSSDe.....	37
5.3.2 OSSDe specifics.....	37
5.3.3 Start-up of an FS-Device (Ready pulse).....	40
5.3.4 Electric characteristics of a receiver in FS-Device and FS-Master.....	41
5.4 Electric and dynamic characteristics of an FS-Device	41
5.5 Electric and dynamic characteristics of an FS-Master port (OSSDe)	44
5.6 FS-Master port FS-DI interface	45
5.7 Wake-up coordination	45
5.8 Fast start-up	46
5.9 Power supply	46
5.10 Medium.....	47

5.10.1	Constraints	47
5.10.2	Connectors	47
5.10.3	Cable characteristics	47
6	Extensions to SIO	47
7	Extensions to the data link layer (DL)	47
7.1	Overview	47
7.2	State machine of the FS-Master DL-mode handler	47
7.3	State machine of the FS-Device DL-mode handler	49
8	Extensions to the Master Configuration Manager (CM)	51
9	Extensions of the FS-Device	52
9.1	Principle architecture and models	52
9.1.1	FS-Device architecture	52
9.1.2	FS-Device model	53
9.2	Parameter Manager (PM)	54
9.3	Process Data Exchange (PDE)	54
9.4	Data Storage (DS)	54
9.4.1	General considerations and extensions including safety	54
9.4.2	Backup levels	55
10	Extensions of the FS-Master	55
10.1	Principle architecture	55
10.2	SMI service extensions	56
10.2.1	Overview	56
10.2.2	SMI_FSMasterAccess	58
10.2.3	SMI_SPDUIn	60
10.2.4	SMI_SPDUOut	60
10.2.5	SMI_FSPDInOut	60
10.3	ArgBlock extensions	62
10.3.1	Overview	62
10.3.2	FSMasterAccess	62
10.3.3	FSCPAAuthenticity	63
10.3.4	FSPortConfigList	63
10.3.5	FSPortStatusList	66
10.3.6	SPDUIn	67
10.3.7	SPDUOut	67
10.3.8	FSPDInOut	68
10.4	Safety Layer Manager (SLM)	69
10.4.1	Purpose	69
10.4.2	FS_PortModes	69
10.4.3	FSP parameter	69
10.5	Process Data Exchange (PDE)	73
10.6	Data Storage (DS)	74
11	Safety communication layer (SCL)	74
11.1	Functional requirements	74
11.2	Communication errors and safety measures	74
11.3	SCL services	75
11.3.1	Positioning of safety communication layers (SCL)	75
11.3.2	FS-Master SCL services	76
11.3.3	FS-Device SCL services	77

11.4	SCL protocol	79
11.4.1	Protocol phases to consider	79
11.4.2	FS-Device faults	80
11.4.3	Safety PDU (SPDU)	80
11.4.4	FS-Input and FS-Output data	81
11.4.5	Port number	81
11.4.6	Status and control	81
11.4.7	CRC signature	82
11.4.8	TADI safety considerations (informative)	83
11.4.9	Data types for SDCI-FS	84
11.5	SCL behavior	85
11.5.1	General	85
11.5.2	SCL state machine of the FS-Master	85
11.5.3	SCL state machine of the FS-Device	88
11.5.4	Sequence charts for several use cases	91
11.5.5	Monitoring of safety times	98
11.5.6	Reaction in the event of a malfunction	99
11.5.7	Start-up (communication)	101
11.6	SCL management	101
11.6.1	Parameter overview (FSP and FST)	101
11.6.2	Parameterization approaches	103
11.7	Integrity measures	104
11.7.1	IODD integrity	104
11.7.2	Tool integrity	104
11.7.3	Transmission integrity	104
11.7.4	Verification record	104
11.7.5	Authentication	105
11.7.6	Storage integrity	105
11.7.7	FS I/O data structure integrity	106
11.7.8	Technology parameter (FST) based on IODD	106
11.7.9	Technology parameter (FST) based on existing Dedicated Tool (IOPD)	107
11.8	Creation of FSP and FST parameters	108
11.9	Integration of Dedicated Tools (IOPD)	109
11.9.1	IOPD interface	109
11.9.2	Standard interfaces	109
11.9.3	Backward channel	110
11.10	Validation	111
11.11	Passivation	111
11.11.1	Motivation and means	111
11.11.2	Port selective (FS-Master)	111
11.11.3	Signal selective (FS-Terminal)	112
11.11.4	Qualifier settings in case of communication	112
11.11.5	Qualifier handling in case of OSSDe	112
11.12	SCL diagnosis	113
12	Functional safe processing (FS-P)	114
12.1	Recommendations for efficient I/O mappings	114
12.2	Embedded FS controller	114
Annex A	(normative) Extensions to parameters	115
A.1	Indices and parameters for SDCI-FS	115

A.2	Parameters in detail.....	116
A.2.1	FSP_Authenticity	116
A.2.2	FSP_Port.....	116
A.2.3	FSP_AuthentCRC	117
A.2.4	FSP_ProtVersion	117
A.2.5	FSP_ProtMode	117
A.2.6	FSP_Watchdog.....	117
A.2.7	FSP_IO_StructCRC	118
A.2.8	FSP_TechParCRC.....	119
A.2.9	FSP_ProtParCRC	119
A.2.10	FSP_VerifyRecord	119
A.2.11	FSP_TimeToReady.....	119
A.2.12	FSP_MinShutDownTime	120
A.2.13	FSP_WCDT	120
A.2.14	FSP_OFDT	120
A.2.15	FSP_ParamDescCRC	120
Annex B (normative)	Extensions to EventCodes	121
B.1	Additional FS-Device EventCodes.....	121
B.2	Additional Port EventCodes	121
Annex C (normative)	Extensions to Data Types	123
C.1	Data types for SDCI-FS	123
C.2	BooleanT (bit).....	123
C.3	IntegerT (16).....	124
C.4	IntegerT (32).....	124
C.5	Safety Code	125
Annex D (normative)	CRC generator polynomials	126
D.1	Overview of CRC generator polynomials	126
D.2	Residual error probabilities	126
D.3	Implementation considerations.....	128
D.3.1	Overview	128
D.3.2	Bit shift algorithm (16 bit).....	128
D.3.3	Lookup table (16 bit).....	128
D.3.4	Bit shift algorithm (32 bit).....	130
D.3.5	Lookup table (32 bit).....	130
D.3.6	Seed values.....	131
D.3.7	Octet order for CRC calculation	132
Annex E (normative)	IODD extensions	133
E.1	General.....	133
E.2	Schema	133
E.3	IODD constraints	133
E.3.1	General rules.....	133
E.3.2	Description of the IODD structure	133
E.3.3	Behavior of "Reset" SystemCommands in SDCI-FS	140
E.3.4	Profile Characteristic	141
E.3.5	ProcessDataInput and ProcessDataOutput	141
E.4	IODD conventions.....	141
E.4.1	Naming	141
E.4.2	Process Data (PD).....	141

E.4.3	IODD conventions for user interface	142
E.4.4	Master Tool features	142
E.5	Securing	142
E.5.1	General	142
E.5.2	DefaultValues for FSP	143
E.5.3	FSP_Authenticity	143
E.5.4	FSP_Protocol	143
E.5.5	FSP_IO_Description	144
E.5.6	Sample serialization for FSP_ParamDescCRC	144
E.5.7	FST and FSP parameters and Data Storage	145
E.5.8	Sample IODD of an FS-Device	145
Annex F (normative)	Device Tool Interface (DTI) for SDCI	156
F.1	Purpose of DTI	156
F.2	Base model	156
F.3	Invocation interface	157
F.3.1	Overview	157
F.3.2	Detection of Device Tool	158
F.3.3	Program Interface Description – PID	161
F.3.4	Temporary Parameter File – TPF	164
F.3.5	Temporary Backchannel File – TBF	169
F.3.6	Temporary Acknowledgment File – TAF	171
F.3.7	Invocation behavior	171
F.4	Device data objects (DDO)	172
F.4.1	General	172
F.4.2	Structure of DDO package	173
F.5	Communication Interface	173
F.5.1	General	173
F.5.2	Principle of DTI communications	174
F.5.3	Gateways	175
F.5.4	Configuration of the Communication Server	175
F.5.5	Definition of the Communication Interface	176
F.5.6	Sequence for establishing a communication relation	176
F.5.7	Usage of the Communication Server in stand-alone mode	177
F.5.8	SDCI specifics	178
F.5.9	Changing communication settings	178
F.6	Reaction on incorrect Tool behavior	179
F.7	Compatibility	179
F.7.1	Schema validation	179
F.7.2	Version policy	180
F.8	Scalability	180
F.8.1	Scalability of a Device Tool	180
F.8.2	Scalability of a Master Tool	181
F.8.3	Interactions at conformance class combinations	181
F.9	Schema definitions	181
F.9.1	General	181
F.9.2	Schema of the PID	181
F.9.3	Schema of the TPF	183
F.9.4	Schema of the TBF	185
F.9.5	Schema of the TAF	186

F.9.6	Schema of DTI primitives.....	187
Annex G (normative)	Main scenarios of SDCI-FS.....	189
G.1	Overview.....	189
G.2	Sequence chart of commissioning.....	190
G.3	Sequence chart of replacement.....	191
G.4	Sequence chart of misconnection.....	192
Annex H (normative)	System requirements	193
H.1	Indicators.....	193
H.1.1	General	193
H.1.2	OSSDe	193
H.1.3	Safety communication.....	193
H.1.4	Acknowledgment request.....	193
H.2	Installation guidelines, electrical safety, and security	193
H.3	Safety function response time	194
H.4	Duration of demands.....	194
H.5	Maintenance and repair	194
H.6	Safety manual.....	194
Annex I (informative)	Information for test and assessment of SDCI-FS components.....	195
Bibliography		196
Figure 1	Positioning of SDCI-FS in functional safety automation	14
Figure 2	Relationship of this document to standards	16
Figure 3	Memory and transmission octet order.....	25
Figure 4	SDCI-FS communication layer model	25
Figure 5	Port interface extensions for SDCI-FS.....	26
Figure 6	Migration to SDCI-FS	27
Figure 7	Minimized paradigm shift from FS-DI to FS-Master.....	28
Figure 8	FS-Master types and feature levels	28
Figure 9	Original pin layout of SDCI (port class A)	29
Figure 10	Optimized OSSDe commissioning with FS-Master	30
Figure 11	Level "d" of an FS-Master (Class B)	31
Figure 12	Off-site configuration and parameterization	31
Figure 13	SDCI-FS within the automation hierarchy	33
Figure 14	The SDCI physical layer of an FS-Master (class A)	36
Figure 15	The physical layer of an FS-Device (class A)	36
Figure 16	Cross compatibility OSSD and OSSDe	37
Figure 17	Principle OSSDe function	38
Figure 18	Test pulses to detect cross connection faults	39
Figure 19	OSSD timings.....	40
Figure 20	Typical start-up of an OSSD sensor	40
Figure 21	Start-up of an FS-Device	40
Figure 22	Switching thresholds for FS-Device and FS-Master receivers.....	41
Figure 23	Reference schematics (one OSSDe channel)	42
Figure 24	Voltage level definitions	42
Figure 25	Charge capability at power-up.....	45

Figure 26 – OSSDe input filter conflict resolution	45
Figure 27 – Start-up of an FS-Device	46
Figure 28 – Required fast start-up timings	46
Figure 29 – State machine of the FS-Master DL-mode handler	48
Figure 30 – State machine of the FS-Device DL-mode handler	50
Figure 31 – Extension to the Configuration Manager (VerifyRecord)	51
Figure 32 – Principle architecture of the FS-Device	53
Figure 33 – The FS-Device model	54
Figure 34 – Principle architecture of the FS-Master	56
Figure 35 – SMI service extensions	58
Figure 36 – FSP parameter use cases	70
Figure 37 – PDE Splitter	73
Figure 38 – PDE Composer	74
Figure 39 – Positioning of the SDCI-FS Safety Communication Layer (SCL)	76
Figure 40 – FS-Master Safety Communication Layer services	76
Figure 41 – FS-Device Safety Communication Layer services	78
Figure 42 – Protocol phases to consider	79
Figure 43 – Safety PDUs of FS-Master and FS-Device	80
Figure 44 – The 1 % share rule of IEC 61784-3:2021	83
Figure 45 – SCL state machine of the FS-Master	86
Figure 46 – SCL state machine of the FS-Device	89
Figure 47 – FS-Master and FS-Device both with power ON	92
Figure 48 – FS-Master power OFF → ON	93
Figure 49 – FS-Device with delayed SCL start	94
Figure 50 – FS-Device with power OFF and ON	95
Figure 51 – FS-Master detects CRC signature error	96
Figure 52 – FS-Device detects CRC signature error	97
Figure 53 – Monitoring of the SCL cycle time	98
Figure 54 – Parameter types and assignments	102
Figure 55 – FSCP-Host-centric system	103
Figure 56 – Structure of the FSP_VerifyRecord	105
Figure 57 – Start-up of SDCI-FS	106
Figure 58 – Securing of FST parameters via dedicated tool	107
Figure 59 – Modification of FST parameters via Device Tool	108
Figure 60 – Creation of FSP and FST parameters	109
Figure 61 – Example of a communication hierarchy	110
Figure 62 – Motivation for Port selective passivation	111
Figure 63 – Qualifier handler (communication)	112
Figure 64 – Qualifier handler (OSSDe)	112
Figure 65 – Qualifier behavior per FS-Master port	113
Figure 66 – Mapping efficiency issues	114
Figure A.1 – Instance of an FS I/O data description	118
Figure A.2 – Example FS I/O data structure with non-safety data	119

Figure A.3 – Securing of safety parameters	120
Figure C.1 – Example of a BooleanT data structure	124
Figure C.2 – Safety Code of an output message	125
Figure C.3 – Safety Code of an input message	125
Figure D.1 – CRC-16 generator polynomial.....	127
Figure D.2 – CRC-32 generator polynomial.....	127
Figure D.3 – Bit shift algorithm in "C" language (16 bit).....	128
Figure D.4 – CRC-16 signature calculation using a lookup table	128
Figure D.5 – Bit shift algorithm in "C" language (32 bit).....	130
Figure D.6 – CRC-32 signature calculation using a lookup table	130
Figure E.1 – Algorithm to build the FSP parameter CRC signatures	143
Figure F.1 – Principle of DTI invocation interface	157
Figure F.2 – Structure of the registry	158
Figure F.3 – Example of a DTI registry	159
Figure F.4 – Detection of a Device tool in registry	161
Figure F.5 – Menu for Device Tool invocation	162
Figure F.6 – Purpose of Device data objects (DDO).....	173
Figure F.7 – Communication routes between Device Tool and Device.....	174
Figure F.8 – Routing across networks and SDCI	174
Figure F.9 – Communication Server	175
Figure F.10 – Sequence chart for establishing communication	176
Figure F.11 – Create Communication Server Instance.....	177
Figure F.12 – Example of a Connect Request XML document for SDCI.....	178
Figure F.13 – XML schema of the PID file	181
Figure F.14 – XML schema of the TPF	183
Figure F.15 – XML schema of a TBF	185
Figure G.1 – Commissioning with test and armed operation	191
Figure G.2 – FS-Device replacement	192
Figure G.3 – FS-Device misconnection	192
Table 1 – Operational modes of feature level "a" to "c" (port class A).....	30
Table 2 – Interoperability matrix of safety devices.....	32
Table 3 – PL_Ready	36
Table 4 – OSSD states and conditions	38
Table 5 – Cross connection faults	39
Table 6 – Electric characteristics of a receiver	41
Table 7 – Electric and dynamic characteristics of the FS-Device (OSSDe).....	43
Table 8 – Electric and dynamic characteristics of the Port interface	44
Table 9 – Cable characteristics	47
Table 10 – State transition tables of the FS-Master DL-mode handler	49
Table 11 – State transition tables of the FS-Device DL-mode handler	50
Table 12 – State transition tables of the Configuration Manager	52
Table 13 – Extension to Data Storage (DS) state machine	55

Table 14 – Data Storage Backup Levels	55
Table 15 – SMI services used for FS-Master.....	57
Table 16 – SMI_FSMasterAccess	59
Table 17 – SMI_FSPDInOut.....	61
Table 18 – ArgBlock types and ArgBlockIDs	62
Table 19 – FSMasterAccess	63
Table 20 – FSCPAuthenticity	63
Table 21 – FSPortConfigList	64
Table 22 – FSPortStatusList	66
Table 23 – SPDUIIn	67
Table 24 – SPDUIOut	68
Table 25 – FSPDInOut.....	68
Table 26 – Use case reference table.....	70
Table 27 – Communication errors and safety measures	75
Table 28 – SCL services of FS-Master	77
Table 29 – SCL services of FS-Device	78
Table 30 – Protocol phases to consider	80
Table 31 – Control and counting (Control&MCnt)	81
Table 32 – Status and counting mirror (Status&DCnt).....	81
Table 33 – MCount and DCount_i values	82
Table 34 – FS process I/O data types	84
Table 35 – Rules for the layout of values and qualifiers	84
Table 36 – Order of values and qualifier	85
Table 37 – Definition of terms used in SCL state machine of the FS-Master.....	86
Table 38 – FS-Master SCL states and transitions	87
Table 39 – Definition of terms used in SCL state machine of the FS-Device.....	89
Table 40 – FS-Device SCL states and transitions	90
Table 41 – Timing constraints	98
Table 42 – Qualifier bits "GOOD/BAD"	112
Table 43 – State transition Table for the qualifier behavior.....	113
Table A.1 – Indices for SDCI-FS	115
Table A.2 – Coding of protocol version	117
Table A.3 – Coding of protocol mode	117
Table A.4 – Generic FS I/O data structure description.....	118
Table B.1 – FS-Device SCL specific EventCodes.....	121
Table B.2 – FS-Master SCL specific EventCodes.....	122
Table C.1 – Data types for SDCI-FS	123
Table C.2 – BooleanT for SDCI-FS	123
Table C.3 – Example of BooleanT within a RecordT.....	123
Table C.4 – IntegerT(16).....	124
Table C.5 – IntegerT(16) coding	124
Table C.6 – IntegerT(32).....	125
Table C.7 – IntegerT(32) coding	125

Table D.1 – CRC generator polynomials for SDCI-FS	126
Table D.2 – Definition of variables used in Figure D.3	128
Table D.3 – Definition of variables used in Figure D.4	129
Table D.4 – Lookup Table for CRC-16 signature calculation	129
Table D.5 – Definition of variables used in Figure D.5	130
Table D.6 – Definition of variables used in Figure D.4	130
Table D.7 – Lookup Table for CRC-32 signature calculation	131
Table E.1 – Specific behavior of FS-Device "Reset" SystemCommands	141
Table E.2 – User actions to replace DefaultValues	143
Table E.3 – RecordItems of FSP_Protocol where allowed values shall be serialized	144
Table E.4 – Sample serialization for FSP_ParamDescCRC	144
Table F.1 – Description of PID file elements	162
Table F.2 – Elements of a TPF	165
Table F.3 – Elements of the TBF	170
Table F.4 – Elements of the TAF	171
Table F.5 – Invocation cases and behaviors	172
Table F.6 – Communication Schema mapping	178
Table F.7 – Reaction on incorrect Tool behavior	179
Table F.8 – DTI conformance classes	180
Table F.9 – DTI feature levels of Device Tools	180
Table F.10 – Interactions at conformance class combinations	181
Table G.1 – Main scenarios of SDCI-FS	189

INTERNATIONAL ELECTROTECHNICAL COMMISSION

INDUSTRIAL NETWORKS – SINGLE-DROP DIGITAL COMMUNICATION INTERFACE –

Part 2: Functional safety extensions

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.

IEC 61139-2 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee TC65: Industrial-process measurement, control and automation. It is an International Standard.

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

Draft	Report on voting
65C/1168/FDIS	65C/1174/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 61139 series, published under the general title *Industrial networks – Single-drop digital communication interface*, 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 61139-2:2022

INTRODUCTION

The base technology of IO-Link™¹ is subject matter of the international standard IEC 61131-9 being part of a series of standards on programmable controllers and the associated peripherals such as remote I/O (RIO).

It specifies a single-drop digital communication interface technology – named SDCI, which extends the traditional switching input and output interfaces as defined in IEC 61131-2 towards a point-to-point communication link using coded switching. This technology enables the cyclic exchange of digital input and output process data between a Master and its associated Devices (sensors, actuators, I/O terminals, etc.). The Master can be part of a fieldbus communication system or any stand-alone processing unit. The technology also enables the acyclic transfer of parameters to Devices and the propagation of diagnosis information from the Devices to the upper-level automation system (controller, host) via the Master.

Physical topology is point-to-point from each Device to the Master using 3 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 and supports three transmission rates of 4,8 kbit/s, 38,4 kbit/s and 230,4 kbit/s are supported.

The main advantages of the SDCI technology are:

- dual use of either switching signals (DI/DO) or coded switching communication respectively,
- traditional switching sensors and actuators now providing alternatively single drop digital communication within the same Device,
- one thin, robust, very flexible cable without shielding for power supply and signalling,
- lowest-cost digital communication down to the lowest end sensors and actuators.

The functional safety variant of SDCI is called SDCI-FS. Figure 1 shows an example positioning of SDCI-FS in functional safety automation.

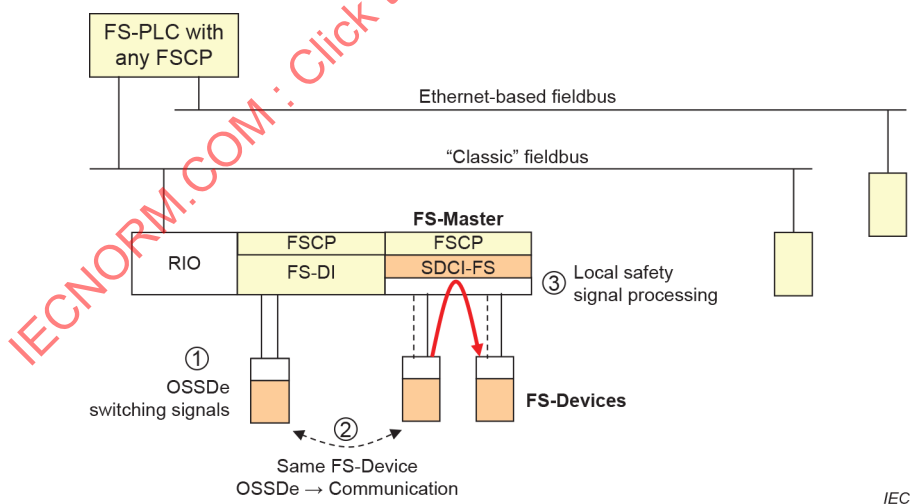


Figure 1 – Positioning of SDCI-FS in functional safety automation

¹ IO-Link™ is a trade name of the "IO-Link Community". This information is given for the convenience of users of this specification and does not constitute an endorsement by the IO-Link Community of the trade name holder or any of its products. Compliance to 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".

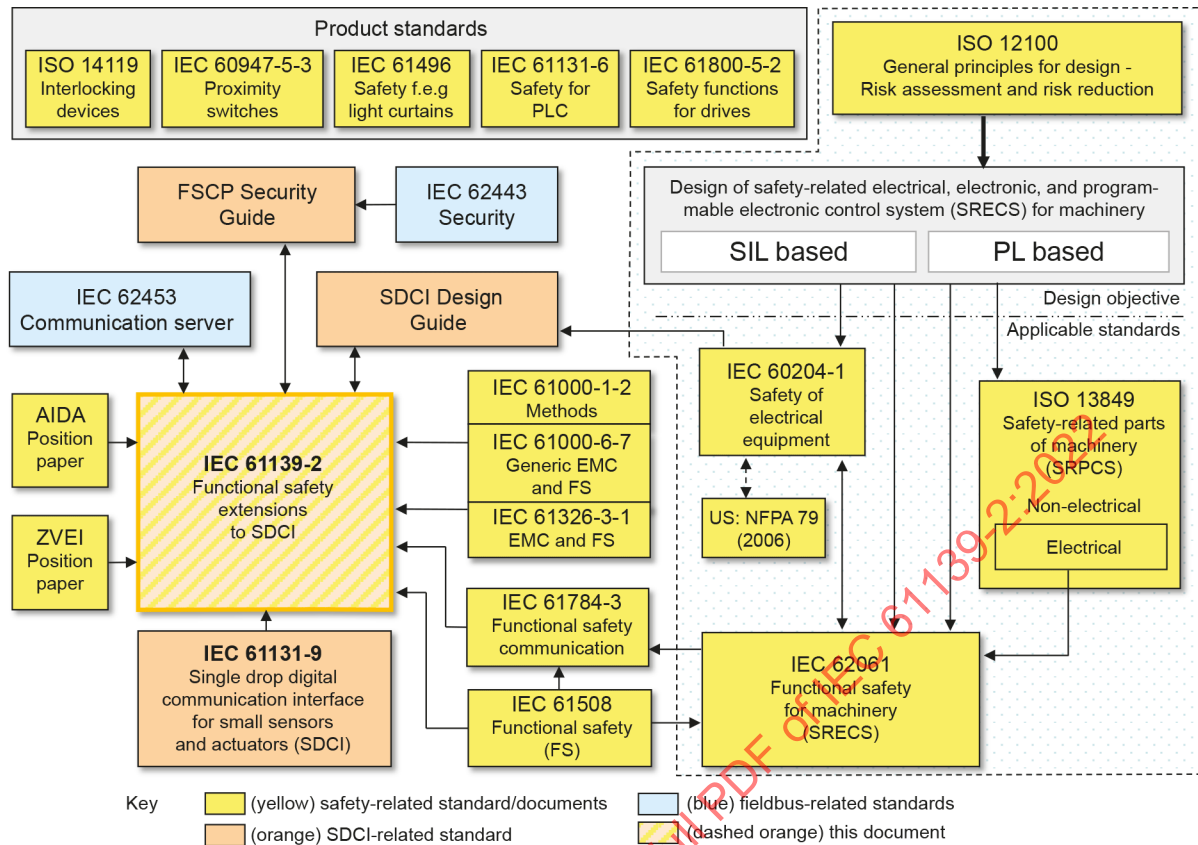
In this example, a remote I/O is connected to a functional safety programmable controller using one of the FSCPs of the IEC 61784-3 series [1]² to communicate with an FS-DI module and a gateway to an SDCI-FS FS-Master. FS-Devices with OSSDe can be connected to FS-DIs or FS-Masters. All FS-Devices can communicate with any FS-Master using the SDCI-FS protocol regardless of the upper-level FSCP-system. The same is true for safety actuators (FS-Devices) such as drives with integrated safety. This means the largest component commonality ① for sensors and actuators similar to the DI and DO interfaces standardized within IEC 61131-2.

Safety sensors with OSSDe interfaces – equipped with SDCI-FS communication – can be parameterized via auxiliary tools such as "USB-Masters", then connected to an FS-DI and operated in OSSDe mode. They also can be operated in OSSDe mode on an FS-Master that supports OSSDe. In case these safety sensors are equipped with SDCI-FS communication in addition, they can be operated in both modes ②, either OSSDe or SDCI-FS. This corresponds to the SDCI SIO paradigm.

The concept of SDCI-FS allows for local safety signal processing if the gateway/FS-Master provides a local safety controller ③.

This document provides the necessary extensions to IEC 61131-9 for functional safety communication including standardization of OSSDe and parameterization within the domain of safety for machinery. Figure 2 shows its relationships to international fieldbus and safety standards as well as to relevant specifications (see Clause 2 and bibliography). Any functional safety starts with risk assessment and risk reduction (ISO 12100). One possibility of risk reduction is the usage of electrical or electronic control systems. For the design of those, standards such as IEC 61508, IEC 62061, and ISO 13849 can be used. Environmental conditions such as EMC are covered by for example IEC 61000-6-7. Further aspects are installations and security issues. A number of product standards such as IEC 60947-5-3 and ISO 14119 complement the generic or sector standards.

² Numbers in square brackets refer to the Bibliography.



IEC

Figure 2 – Relationship of this document to standards

SDCI-FS can be used for functional safety applications according to IEC 62061 and IEC 61508 up to SIL3 and/or according to ISO 13849 up to PL_e.

INDUSTRIAL NETWORKS – SINGLE-DROP DIGITAL COMMUNICATION INTERFACE –

Part 2: Functional safety extensions

1 Scope

This part of IEC 61139 specifies the extensions to SDCI in IEC 61131-9 for functional safety. This comprises:

- a standardized OSSDe interface for redundant switching signals based on IEC 61131-2,
- minor modifications/extensions to state machines of SDCI to support the safety operations,
- a lean functional safety communication protocol on top of the standard SDCI communication which is a black channel according to IEC 61784-3:2021,
- protocol management functions for configuration, parameterization, and commissioning,
- IODD extensions for functional safety,
- a Device tool interface to support Dedicated Tools according to functional safety standards.

This document does not cover:

- communication interfaces or systems including multi-point or multi-drop linkages,
- communication interfaces or systems including multi-channel or encrypted linkages,
- wireless communication interfaces or systems,
- integration of SDCI-FS into upper-level systems such as fieldbuses/FSCPs.

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 60204-1, *Safety of machinery – Electrical equipment of machines – Part 1: General requirements*

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

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

IEC 61131-9:—³, *Programmable controllers – Part 9: Single-drop digital communication interface for small sensors and actuators (SDCI)*

IEC 61496-1, *Safety of machinery – Electro-sensitive protective equipment – Part 1: General requirements and tests*

³ Under preparation. Stage at the time of publication: IEC/CFDIS 61131-9:2022.

IEC 61508-3, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 3: Software requirements*

IEC 61784-3:2021, *Industrial communication networks – Profiles – Part 3: Functional safety fieldbuses – General rules and profile definitions*

IEC 62061, *Safety of machinery – Functional safety of safety-related control systems*

IEC 62443 (all parts), *Security for industrial automation and control systems*

ISO 639-2, *Codes for the representation of names of languages – Part 2: Alpha-3 code*

ISO 639-3, *Codes for the representation of names of languages – Part 3: Alpha-3 code for comprehensive coverage of languages*

ISO 13849-1, *Safety of machinery – Safety-related parts of control systems – Part 1: General principles for design*

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 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>

3.2 Common terms and definitions

3.2.1

address

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

3.2.2

application layer

AL

<SDCI>⁴ part of the protocol responsible for the transmission of Process Data objects and On-request Data objects

3.2.3

block parameter

consistent parameter access via multiple Indices or Subindices

3.2.4

checksum

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

⁴ Angle brackets indicate validity of the definition for the SDCI technology

3.2.5**coded switching**

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

3.2.6**COM1**

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

3.2.7**COM2**

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

3.2.8**COM3**

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

3.2.9**COMx**

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

3.2.10**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.

3.2.11**communication error**

unexpected disturbance of the SDCI transmission protocol

3.2.12**cycle time**

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

3.2.13**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.

3.2.14**Direct Parameter**

directly (page) addressed parameter transferred acyclically via the page communication channel without acknowledgement

3.2.15**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

3.2.16

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.

3.2.17

fallback

transition of a port from coded switching to switching signal mode

3.2.18

ISDU

indexed service data unit used for acyclic acknowledged transmission of parameters that can be segmented in a number of M-sequences

3.2.19

M-sequence

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

3.2.20

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

3.2.21

M-sequence type

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

3.2.22

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.

3.2.23

message

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

3.2.24

On-request Data

acyclically transmitted data upon request of the Master application consisting of parameters or Event data

3.2.25

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: Physical layer also provides means for wake-up and fallback procedures.

[SOURCE: ISO/IEC 7498-1, 7.7.2, modified – the descriptive text has been extracted from 7.7.2, the note has been added.]

3.2.26**Port**

<SDCI> communication medium interface of the Master to one Device

3.2.27**Port operating mode**

state of a Master's Port that can be either INACTIVE, DO, DI, FIXEDMODE, or SCANMODE

3.2.28**Process Data**

input or output values from or to a discrete or continuous automation process cyclically transferred with high priority and in a configured schedule automatically after start-up of a Master

3.2.29**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

3.2.30**switching signal**

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

3.2.31**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

3.2.32**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

3.2.33**wake-up**

procedure for causing a Device to change its mode from SIO to SDCI

3.2.34**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

3.3 Terms and definitions related to SDCI-FS**3.3.1****error**

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

Note 1 to entry: Errors may be due to design mistakes within hardware/software and/or corrupted information due to electromagnetic interference and/or other effects.

Note 2 to entry: Errors do not necessarily result in a *failure* or a *fault*.

SOURCE: [IEC 61508-4:2010, 3.6.11, modified – The notes have been added.]

3.3.2

failure

termination of the ability of a functional unit to perform a required function or operation of a functional unit in any way other than as required

Note 1 to entry: Failure may be due to an error (for example, problem with hardware/software design or message disruption)

SOURCE: [IEC 61508-4:2010, 3.6.4, modified – The notes have been removed and replaced by a new note and the figure has been deleted.]

3.3.3

fault

abnormal condition that may cause a reduction in, or loss of, the capability of a functional unit to perform a required function

Note 1 to entry: IEC 191-05-01 defines “fault” as a state characterized by the inability to perform a required function, excluding the inability during preventive maintenance or other planned actions, or due to lack of external resources.

SOURCE: [IEC 61508-4:2010, 3.6.1, modified – The reference to Figure 4 of IEC 61508-4:2010 in the note has been removed.]

3.3.4

FS-Device

single passive peer such as a functional safety sensor or actuator to a Master with functional safety capabilities

3.3.5

FS-Master

active peer with functional safety capabilities connected through Ports to one up to n Devices or FS-Devices and which provides an interface to the gateway to the upper-level communication systems (NSR or SR) or controllers with functional safety capabilities

3.3.6

FSP parameter

parameter set for the administration and operation of the SDCI-FS protocol

3.3.7

FST parameter

parameter set for the safety-related technology of an FS-Device, for example light curtain

3.3.8

Safety PDU

Safety Protocol Data Unit

SPDU

PDU transferred through the safety communication channel

[SOURCE: IEC 61784-3:2021, 3.1.47, modified – Notes have been removed and admitted term Safety Protocol Data Unit has been added.]

3.4 Symbols and abbreviated terms

AIDA	Automatisierungsinitiative Deutscher Automobilhersteller (Automation initiative of the German automotive manufacturers)	
AL	application layer	
BEP	bit error probability	
C/Q	connection for communication (C) or switching (Q) signal (SIO)	
CRC	cyclic redundancy check	
DDO	Device data object	
DI	digital input	
DIP	dual in-line package	
DL	data link layer	
DO	digital output	
DS	data storage	
DTI	Device Tool Interface	
DTM	Device Type Manager	[IEC 62453 (all parts)]
FDI	Field Device Integration	[IEC 62769 (all parts)]
FDT	Field Device Tool	[IEC 62453 (all parts)]
FS	functional safety	
FSCP	functional safety communication profile (for example IEC 61784-3 series [1])	
FS-AI/AO	functional safety analog input/output	
FS-DI/DO	functional safety digital input/output	
FSP	functional safety protocol (parameter)	
FST	functional safety technology (parameter)	
I/O	input / output	
IODD	IO Device Description	
IOPD	SDCI Parameterization & Diagnostic tool	
I/Q	connection with several options: DI, OSSD2e, or DO	
L-	power supply (-)	
L+	power supply (+)	
LSO, MSO	least significant octet, most significant octet	
M12	circular connector	[IEC 61076-2-113]
N24	24 V extra power supply (-); Port class B	
NSR	non-safety-related	
OD	On-request Data	
OK	"OK", values or state correct	
OSSD	output signal switching device (self-testing electronic device with built-in OSSD)	[IEC 61496-1]
OSSDe	output signal switching device (self-testing electronic device with built-in OSSD according to this document)	
OSSD1/2e	pin assignment of both OSSDe signals	
OSSDm	output signal switching device (relay and solid state outputs)	[IEC 60947-5-5]

P24	24 V extra power supply (+); Port class B	
PD	Process Data	
PDin	functional safety input process data (from an FS-Master's view)	
PDout	functional safety output process data (from an FS-Master's view)	
PDCT	port and Device configuration tool	
PDU	protocol data unit	
PFH	average frequency of a dangerous failure per hour	[IEC 61508-4]
PID	program interface description	
PL	physical layer	
PLC	programmable logic controller	
PS	power supply (measured in V)	
RIO	remote I/O	
rms	root mean square ("effective")	
SCL	safety communication layer	
SDCI	single-drop digital communication interface	[IEC 61131-9]
SDCI-FS	single-drop digital communication interface for functional safety	
SIO	standard input output (digital switching mode)	[IEC 61131-2]
SLM	safety layer manager	
SM	system management	
SPDU	safety protocol data unit	
SR	safety-related	
SSI	synchronous serial interface (usually for encoders)	
TAF	temporary acknowledgment file	
TBF	temporary backchannel file	
TPF	temporary parameter file	
UART	universal asynchronous receiver transmitter	
UML 2	unified modeling language, edition 2	[ISO/IEC 19505-2]
USB	universal serial bus	[www.usb.org]
WURQ	wake-up request pulse	
XML	extensible markup language	

3.5 Conventions

3.5.1 Behavioral descriptions

For the behavioral descriptions, the notations of UML 2 are used, mainly for state and sequence diagrams (see [2], [3], [4]).

Events to trigger a transition usually can be a signal, service call, or timeout. Logic conditions (true/false) shall be the result of a [guard]. To alleviate the readability and the maintenance of the state machines, the diagrams do not provide the actions associated with a transition. These actions are listed within a separate state-transition table according to [5].

The state diagrams shown in this document are entirely abstract descriptions. They do not represent a complete specification for implementation.

3.5.2 Memory and transmission octet order

Figure 3 demonstrates the order that shall be used when transferring WORD based data types from memory to transmission and vice versa.

NOTE Existing microcontrollers can differ in the way WORD based data types are stored in memory: "big endian" and "little endian". If designs are not considering this fact, octets can be erroneously permuted for transmission.

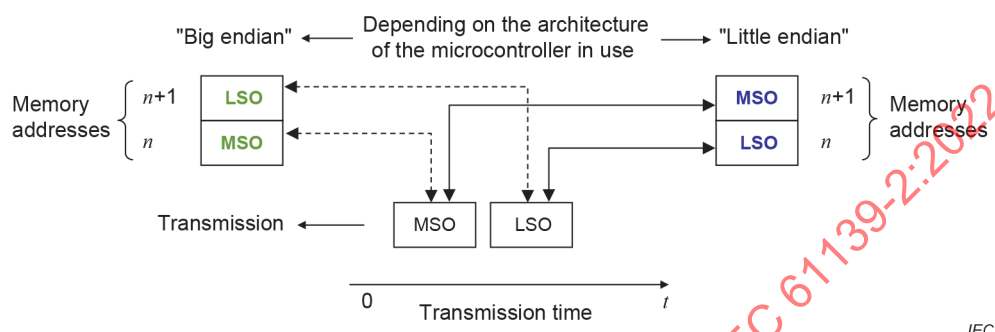


Figure 3 – Memory and transmission octet order

4 Overview of SDCI-FS

4.1 Purpose of the technology and feature levels

4.1.1 Base SDCI-FS technology

This document specifies a new lean functional safety communication protocol on top of the existing SDCI transmission system specified in IEC 61131-9. Figure 4 illustrates how the corresponding SDCI-FS communication layers are located within the architectural models of Master and Device such that they become FS-Master and FS-Device. Most of the original SDCI design remains unchanged for this document.

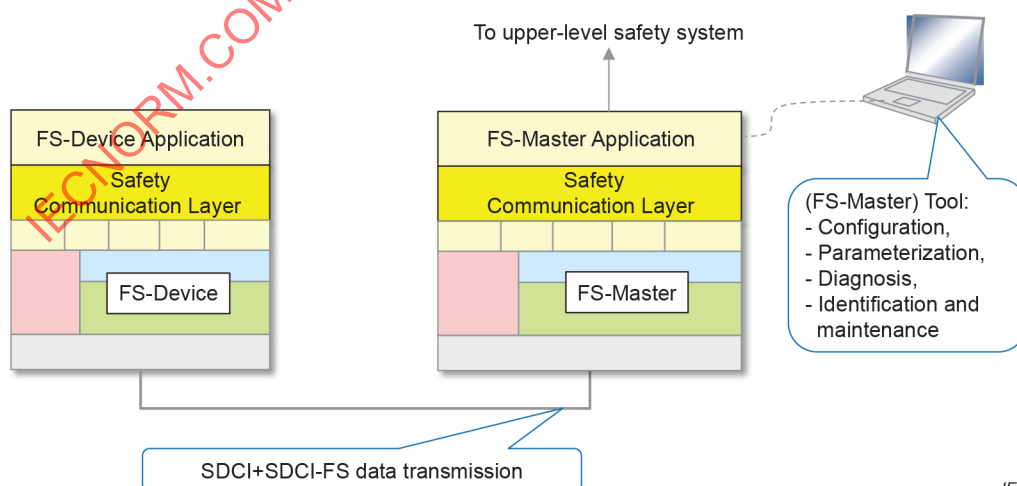


Figure 4 – SDCI-FS communication layer model

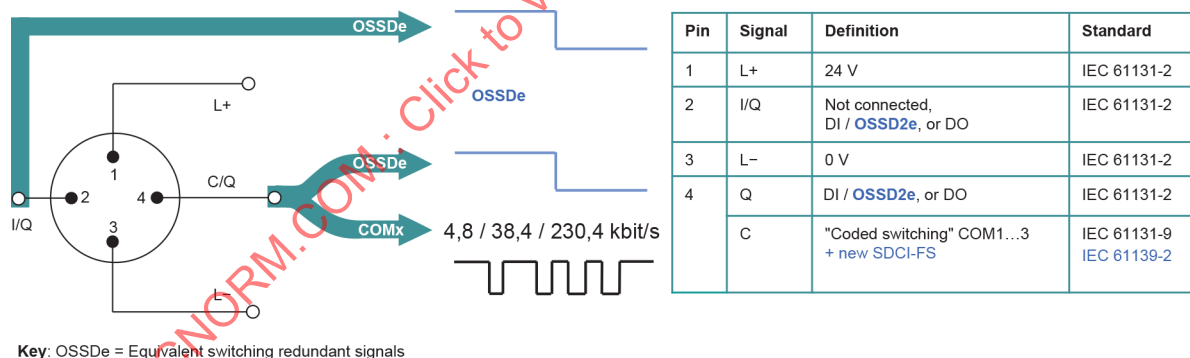
SDCI-FS increases the number of Port modes and thus requires changes to the Physical Layer and Configuration/System Management.

SDCI-FS also supports OSSDe as a migration strategy, like the SIO mode. It does not support

- wireless connections between FS-Master and FS-Device (see Clause H.2),
- cascaded FS-Master/FS-Device systems.

In "Safety-for-Machinery", usually the switch states (on/off) of relays or sensors are transmitted similar to standard SDCI (SIO) as a 24 V or 0 V signal to FS-DI-Modules within remote I/Os. In contrast to standard SDCI-FS, due to safety requirements, these signals are redundant, either equivalent (OSSDe = 11→00) or antivalent (OSSDm = 01→10) switching.

The electrical characteristics for the OSSDe interface are following IEC 61131-2, type 1 (see Figure 5).



Measurement of physical quantities such as temperature, pressure, position, or strain (FS-AI-Modules) has several interface solutions such as 4 mA to 20 mA, 0 V to 10 V, or SSI, but no common signal transmission technology (see Figure 6, left).

Actuators such as motors can be de-energized via FS-DO-Modules and connected relays as shown in Figure 6 (left).

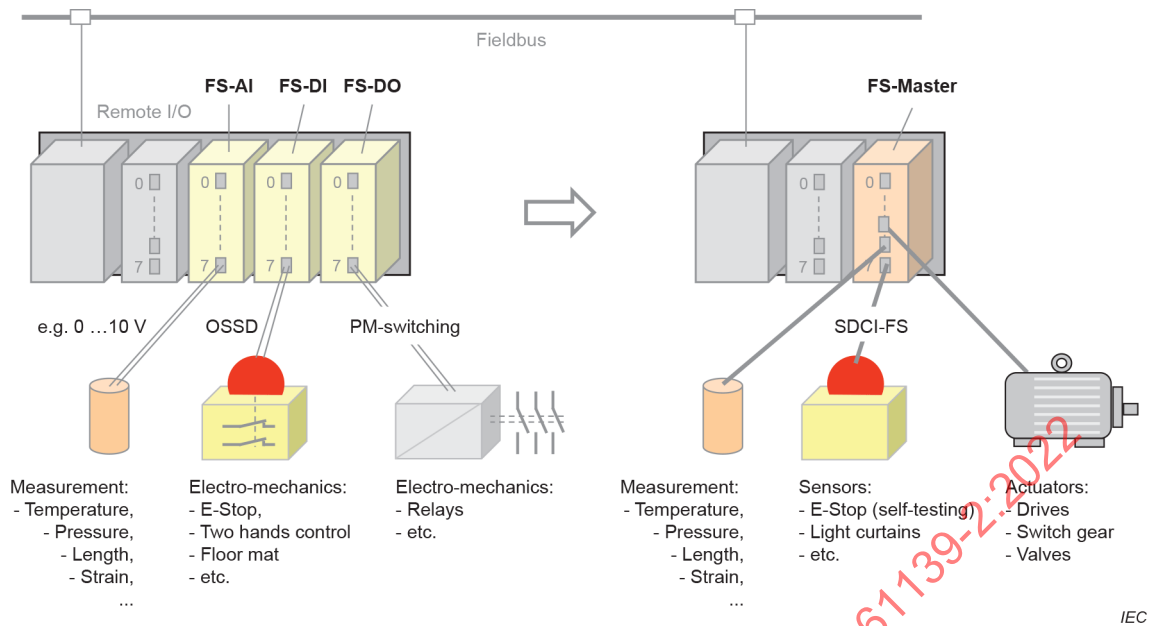


Figure 6 – Migration to SDCI-FS

Without additional interfaces, it was not possible in all cases to configure or parameterize the safety devices or to receive diagnosis information.

SDCI-FS can now provide a functional safe and reliable solution for process data exchange (signal states and measurement values) via single drop digital communication (SDCI), as well as parameterization and diagnosis (see Figure 6, right).

4.1.3 Minimized paradigm shift from FS-DI to FS-Master

Similar to nowadays safety devices for FS-DI modules (see Figure 7) and in contrast to FSCP-based safety devices, it is not necessary to

- setup an *authenticity code switch or adequate software solution*,
- assign a *watchdog time*,
- use any software tool in case of *FS-Device replacement*.

Authenticity is guaranteed through checking of the correct FS-Device to the assigned FS-Master Port during commissioning like FS-DI modules. However, SDCI-FS provides means to discover any incorrect plugging.

SDCI-FS uses a watchdog timer for the transmission of safety data in time (Timeliness). The system is able to calculate the required watchdog time automatically due to the point-to-point nature of the transmission.

FS-Device replacement without tools can be achieved using the original SDCI Data Storage mechanism.

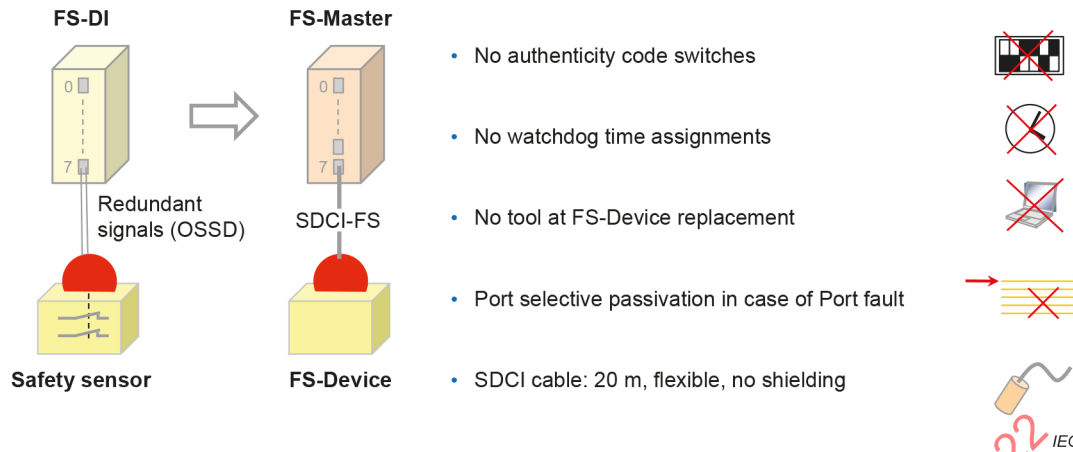


Figure 7 – Minimized paradigm shift from FS-DI to FS-Master

The FS-Master supports *port selective passivation* in case of a port fault and *signal granular passivation* in case of a channel fault within for example a remote I/O terminal ("Hub") connected to an FS-Master Port.

Cables are the same as with SDCI, i.e. unshielded with a maximum of 20 m. However, due to the higher permitted power supply current of 1000 mA per Port, the overall loop resistance RL_{eff} can only be 1,2 Ohm (see Table 9 and IEC 61131-9).

Compliance to AIDA rules requires cable color to be any except yellow. However, the connector color shall be yellow (RAL 1004).

4.1.4 Following the SDCI paradigm (SIO vs. OSSDe)

Standard SDCI-FS supports a port type A (4 pins) without extra power supply and a port type B (5 pins) with extra 24 V power supply (see IEC 61131-9). SDCI-FS takes care of several specification levels "a" to "d" (see Figure 8). The number of pins refers to the possible FS-Master pins.

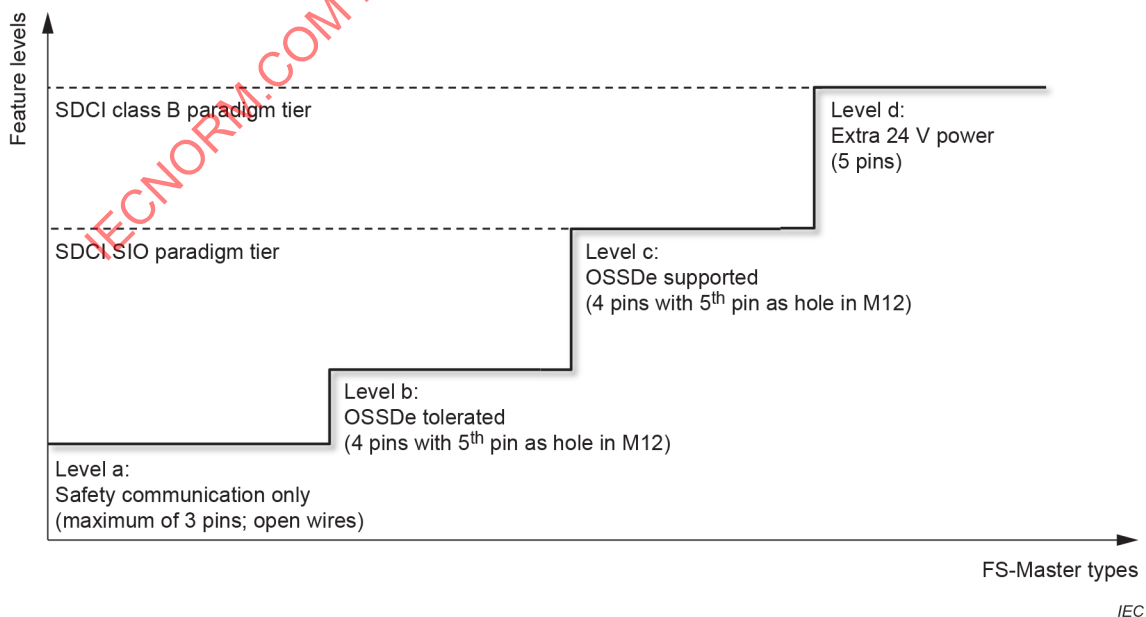


Figure 8 – FS-Master types and feature levels

The original pin layouts of SDCI for port class A are shown in Figure 9 together with the extensions for level "a" through "c". Table 1 shows the details of these levels.

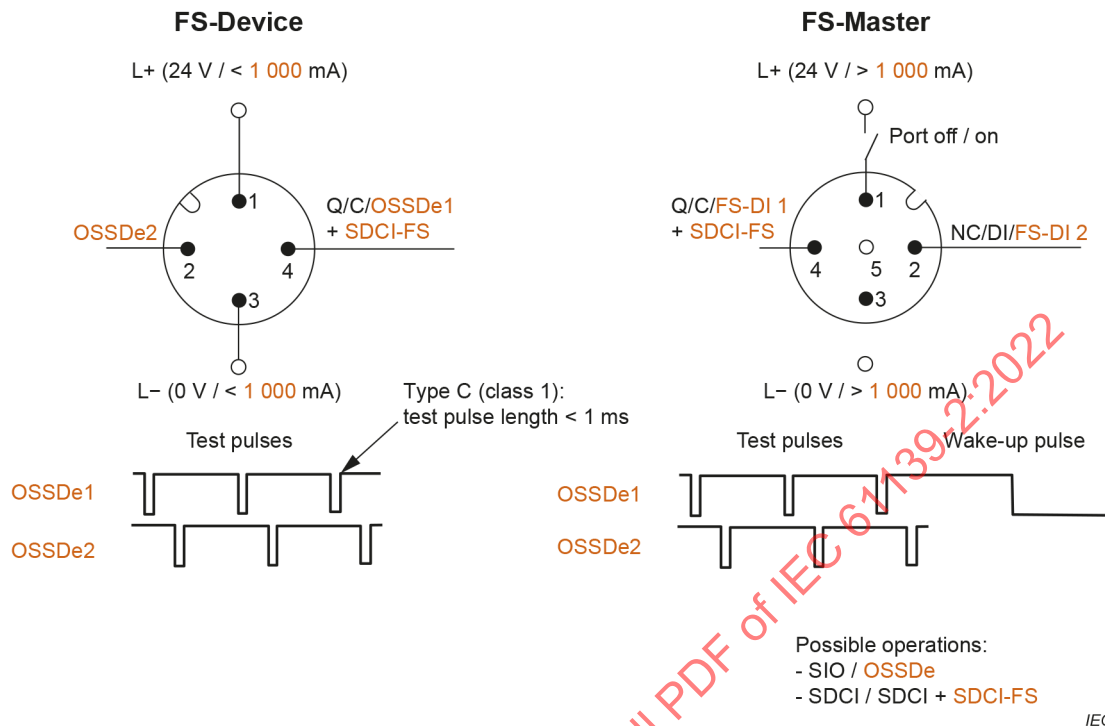


Figure 9 – Original pin layout of SDCI (port class A)

Level "a" provides communication only (Pin 1, 3, and 4). That means support for sensor-type FS-Devices and actuator-type FS-Devices.

Due to the redundant nature of most of the safety device interfaces, SDCI-FS considers pin 2 for the redundant signal path (e.g. OSSDe2) besides pin 4 for the primary signal path (e.g. OSSDe1)⁵. Thus, level "b" allows FS-Devices to provide OSSDe outputs besides the SDCI-FS communication capability. They can be parameterized with the help of a "USB-Master" and be connected to any FS-DI module in switching mode. When connected to an FS-Master, safety and standard non-safety communication is possible.

Level "c" corresponds to the SIO level of standard SDCI Master. In this case, the FS-Master supports an OSSDe mode besides communication (Pin 1, 3, 4 and 2).

Table 1 shows the pin layout and possible operational modes for the feature levels "a" to "c" of the port class A FS-Device and FS-Master.

⁵ FS-Devices are based on electronics and not on relays. Thus, the electronic version OSSDe is considered.

Table 1 – Operational modes of feature level "a" to "c" (port class A)

Feature level	FS-Device		FS-Master	
	Pin 2	Pin 4	Pin 2	Pin 4
"a"	- NC, DI, DO	- DI, DO - SDCI - SDCI + SDCI-FS	- NC, DI, DO	- DI, DO - SDCI - SDCI + SDCI-FS
"b"	- NC, DI, DO - OSSD2e	- DI, DO - OSSD1e - SDCI - SDCI + SDCI-FS	- NC, DI, DO	- DI, DO - SDCI - SDCI + SDCI-FS
"c"	- NC, DI, DO - OSSD2e	- DI, DO - OSSD1e - SDCI - SDCI + SDCI-FS	- NC, DI, DO - FS-DI2	- DI, DO - FS-DI1 - SDCI - SDCI + SDCI-FS

Figure 10 shows the optimized OSSDe commissioning with FS-Masters:

- No filter adjustments due to fixed maximum test pulse length of 1 ms according to type C and class 1 in [7], and
- No discrepancy time adjustments due to fixed maximum discrepancy.

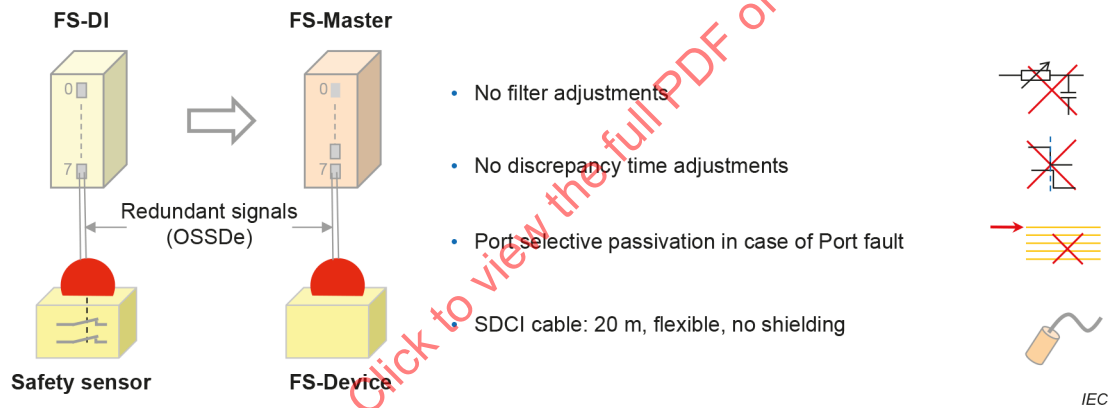


Figure 10 – Optimized OSSDe commissioning with FS-Master

4.1.5 Port class B

A Port class B provides for an extra 24 V power supply for actuators supplementing the main 24 V power supply of SDCI. See IEC 61131-9 for constraints on this extra power supply especially the requirement for electrical isolation of Power 2 from Power 1.

Figure 11 shows the pin layout, signal, and power supply assignment as well as the internal switch for L+.

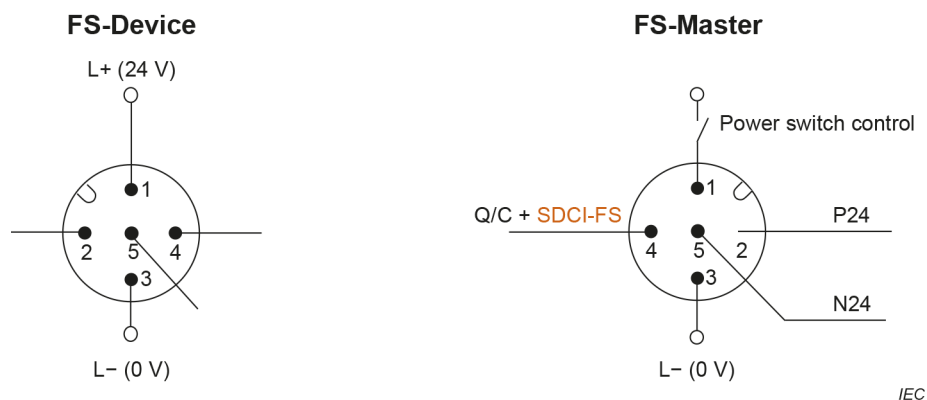


Figure 11 – Level "d" of an FS-Master (Class B)

4.1.6 "USB-Master" with safety parameterization

It is possible to use upgraded "USB-Masters" for off-site configuration, parameterization and test as shown in Figure 12.

Due to functional safety requirements, it will be necessary to extend the Master-Tool software for the functional safe configuration and parameterization of the FS-Device technology (FST-Parameters). See F.4.1 for use case.

Table 2 shows the device types that can be supported by such a "USB-Master".

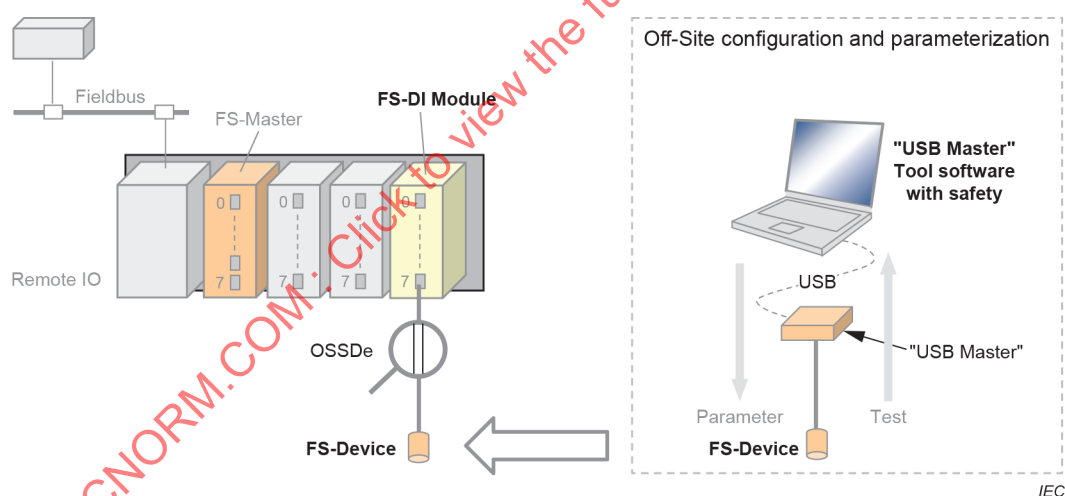


Figure 12 – Off-site configuration and parameterization

4.1.7 Interoperability matrix of safety devices

Table 2 provides an overview of typical safety sensors and actuators and their interoperability with FS-Masters of different feature levels, a "USB-Master" upgraded to safety parameterization, and conventional FS-DI modules connected to FSCPs.

Table 2 – Interoperability matrix of safety devices

Device type	FS-Master			"USB-Master" with safety parameterization	FS-DI module (FSCP)
	Communi-cation	OSSDe tolerated	OSSDe supported		
	"a"	"b"	"c"		
Sensor with OSSDe ^a	-	-	OSSDe	-	OSSDe
Sensor with OSSDe and SDCI	-	-	OSSDe	SDCI ^b	OSSDe
Sensor with OSSDe and SDCI-FS	SDCI-FS	SDCI-FS	OSSDe or SDCI-FS	SDCI	OSSDe
Sensor with SDCI-FS communication only, e.g. light curtain	SDCI-FS	SDCI-FS	SDCI-FS	SDCI	-
Sensor with OSSDm, e.g. E-Stop	-	-	-	-	OSSDm
Actuator with SDCI-FS, e.g. 400 V power drive, low voltage switch gear	SDCI-FS	SDCI-FS	SDCI-FS	SDCI	-
Key SDCI-FS = SR and NSR data exchange		USB = Universal Serial Bus, currently the most common interface amongst possible others for offsite parameterization tools due to fast communication combined with power supply			
^a Pin layout according to [8].					
^b Pin layout may differ					

4.2 Positioning within the automation hierarchy

Figure 13 shows the positioning of SDCI-FS within the automation hierarchy.

Classic safety is relay based and thus seemed to be straightforward, easily manageable, and reliable. However, the same criteria that led to the success of fieldbuses, led to the success of functional safety communication profiles (FSCP) on top of the fieldbuses also: reduced wiring, variable parameterization, detailed diagnosis, and more flexibility. SDCI is the perfect complement to the fieldbus communication and bridges the gap to the lowest cost sensors and actuators. It not only provides communication, but power supply on the same flexible and unshielded cable. One type of sensor can be used in the traditional switching mode or in the coded switching mode (communication).

SDCI-FS follows exactly this paradigm with its OSSDe.

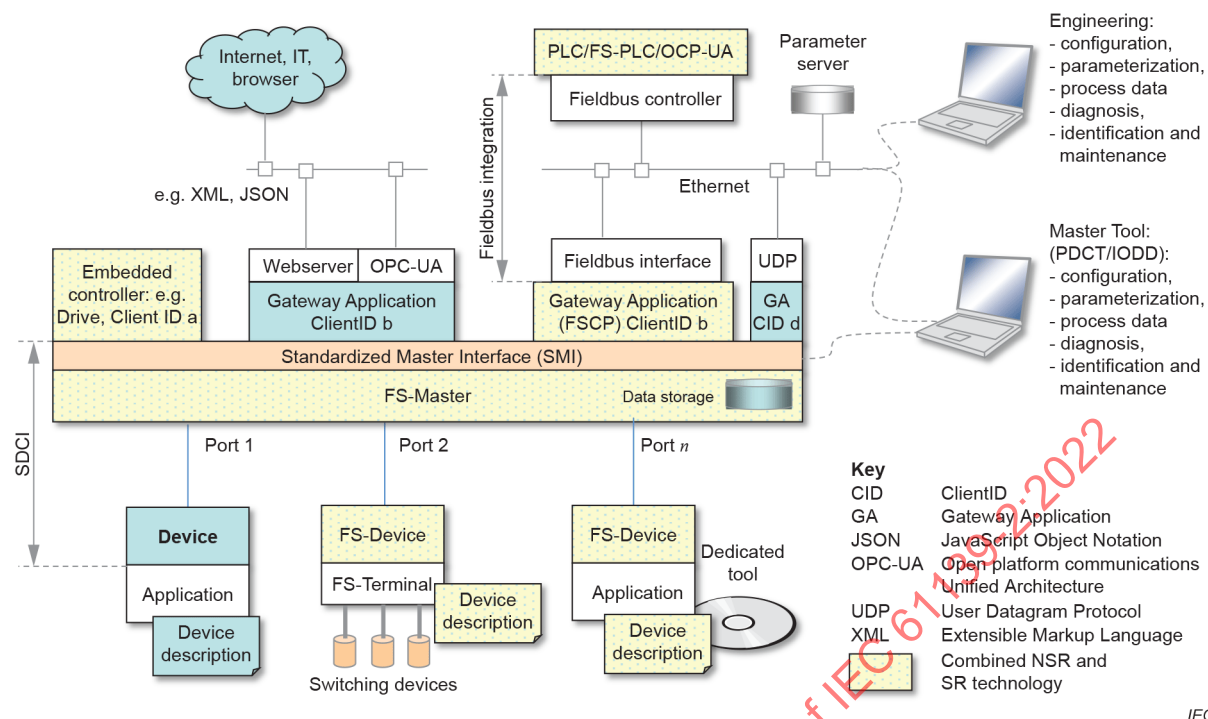


Figure 13 – SDCI-FS within the automation hierarchy

It aims for two main application areas. One is building up safety functions across the SDCI-FS communications and the functional safety communications across fieldbuses. The other builds up safety functions "locally" between an "embedded" safety controller and safety sensors/actuators using SDCI-FS communication.

SDCI-FS allows for building up power saving FS-Devices ("green-line"), for self-testing safety sensors in order to avoid yearly testing, for the reduction of interface types (e.g. 0 V to 10 V, 4 mA to 20 mA, etc.), and for robust and reliable transmission of safety information.

Finally, it is a precondition for new automation concepts such as Industry 4.0 or the Internet-of-Things (IoT).

4.3 Wiring, connectors, and power supply

Port class A types (3 to 4 wires): Cables and connectors as specified in IEC 61131-9 for Class A can be used for SDCI-FS also. However, due to the higher permitted power supply current of up to 1000 mA per Port, the overall loop resistance RL_{eff} can only be 1,2 Ohm. No shielding is required.

Port class B types (5 wires): Cable, wire gauges, shielding, maximum switched currents, interference, signal levels, etc. are not specified within this document.

4.4 Relationship to SDCI

The SDCI communication and its SIO mode are used as the base vehicle ("black channel") for SDCI-FS. Besides SDCI-FS, any FS-Master Port can also be configured for standard SDCI operation.

The independent signal inputs of the SIO mode on Pin 2 and Pin 4 are scanned by an FS-Master simultaneously to achieve an OSSDe interface. The result is propagated to the upper-level safety system as one safety signal. A new Safety Layer Manager supports this feature.

Another new Port configuration mode enables the SDCI-FS communication. Standard state machines are slightly extended to support

- detection of a Ready pulse from the FS-Device on Pin 4;
- power supply (Pin 1) switching OFF/ON in case an FS-Device missed the Wake-up sequence and started its OSSDe operation;
- transmission of functional safety protocol parameters (FSP) during PREOPERATE from FS-Master to the FS-Device;
- activation of the SDCI-FS communication layer (SCL);
- activation of the FS Process Data Exchange within the Safety Layer Manager.

4.5 Communication features and interfaces

FS Process Data from and to an FS-Device are always packed into a safety code envelop consisting of the port number, a safety PDU counter, protocol Control/Status information, and a 16/32-bit CRC signature. The minimum safety PDU size is 4 octets in case of no FS Process Data. SDCI-FS uses M-Sequence TYPE_2_V.

Only a subset of the SDCI data types is permitted: Boolean (packed as record), IntegerT(16), and IntegerT(32).

Parameterization within the domain of safety for machinery requires a "Dedicated Tool" per FS-Device or FS-Device family. The Device Tool Interface (DTI) based on proven technology has been chosen for the links between FS-Master Tool, FS-Device, and its "Dedicated Tool". The FS-Master Tool shall provide communication means for a "Dedicated Tool" to allow for the transmission of safety technology parameters (FST parameters) to and from an FS-Device. The "Dedicated Tool" and the FS-Device are both responsible for sufficient means to secure the transmitted data, for example via CRC signature or read-back.

4.6 Parameterization

SDCI-FS comprises a three-tier concept. The first tier is IODD based and contains all basic non-safety parameters for a Device or FS-Device.

The second tier requires an extension of the IODD for the fixed set of protocol parameters (FSP). These parameters are safety-related and secured via CRC signature against unintended changes of the IODD file. The interpreter of the FS-Master Tool provides a safety-related extension for the handling of the FSP parameters. Usually, the FS-Master Tool is able to determine and suggest the FSP parameter assignments (instance values) automatically and thus relieves the user from assigning these values initially. He can check the plausibility of the values and modify them if required.

The third tier deals with technology specific safety parameters (FST) of an FS-Device. SDCI-FS classifies two types of FS-Devices. Type "basic" requires only a few orthogonal FST parameters, whereas type "complex" can have a number of FST parameters requiring business rules and verification or validation wizards. Usually, the latter comes already with existing PC software ("Dedicated Tool") used for several functional safety communication profiles for fieldbuses.

The FST parameters for type "basic" are coded as any non-safety parameter within the IODD. They can be modified and downloaded to the FS-Device as usual. However, a diverse second path allows for checking these assignments for correctness. At the end of a parameterization session, the user launches a safety-related "Dedicated Tool" (FS-IOPD) for the calculation of a CRC signature across all FST instance values provided by the FS-Master Tool.

For both types of FS-Devices, the "Dedicated Tool" presents a CRC signature, which the user can copy into one of the FSP parameters. Upon reception of the FSP parameters at start-up, the FS-Device calculates a CRC signature across the locally stored instance values and compares it with the received CRC signature.

This method is used also for the check after using the SDCI Data Storage mechanism.

4.7 Role of FS-Master and FS-Gateway

The role of the FS-Master is extended to safe monitoring of Process Data, transferred to and from FS-Devices with respect to timeliness, authenticity, and data integrity according to IEC 61784-3:2021. Concerning authenticity, it uses the authenticity code assigned to the FS-Master by the upper-level FSCP system and the port number. This prevents from local port related misconnections and misconnections whenever several FS-Masters are located side by side.

An FS-Master can be equipped by a safety controller, for example according to IEC 61131-6, or vice versa, and thus build-up a stand-alone safety system with its own complete safety functions.

With the help of an FS-Gateway in conjunction with the FS-Master, safety functions can be build-up across the upper-level FSCP system using the safety sensors and actuators connected to the FS-Master.

4.8 Mapping to upper-level systems

Specification of the mapping to an upper-level FSCP system is the responsibility of the fieldbus organization. SDCI-FS made provisions to meet the majority of FSCPs for example via reduced number of data types, descriptions of safety IO data, port selective passivation, and operator acknowledgment signals to prevent from automatic restart of machines.

4.9 Structure of the document

The structure of this document complies mostly with the structure of IEC 61131-9. Clause 5 specifies the extensions to the Physical Layer (PL), mainly the OSSDe issues, the wake-up behavior, and the additional Port modes. Extensions to SIO are specified in Clause 6, those to data link layer (DL) in Clause 7, those to system management (SM) in Clause 8, those to the FS-Device in Clause 9, and those to the FS-Master in Clause 10.

The core part of this document is the safety communication layer (SCL) in Clause 11. It comprises the SCL services, protocol, state machines, and management. In addition, it deals with integrity measures, with protocol (FSP) and technology (FST) parameters, with the integration of "Dedicated Tools" via Device Tool Interface (DTI) technology, with port selective passivation, and with SCL diagnosis. Clause 12 complements the core part by functional safety processing either through mapping to the upper-level system or local.

Extensions to parameters and commands are specified in Annex A, those to EventCodes in Annex B, and those to data types in Annex C. CRC polynomial issues are presented in Annex D, the IODD aspects in Annex E, the Device Tool Interface technology in Annex F, main scenarios in Annex G, and the system requirements in Annex H. Annex I provides information on test and assessment. Annexes A, C, D, and E are safety related.

5 Extensions to the Physical Layer (PL)

5.1 Overview

Figure 14 shows the adapted physical layer of an FS-Master (class A).

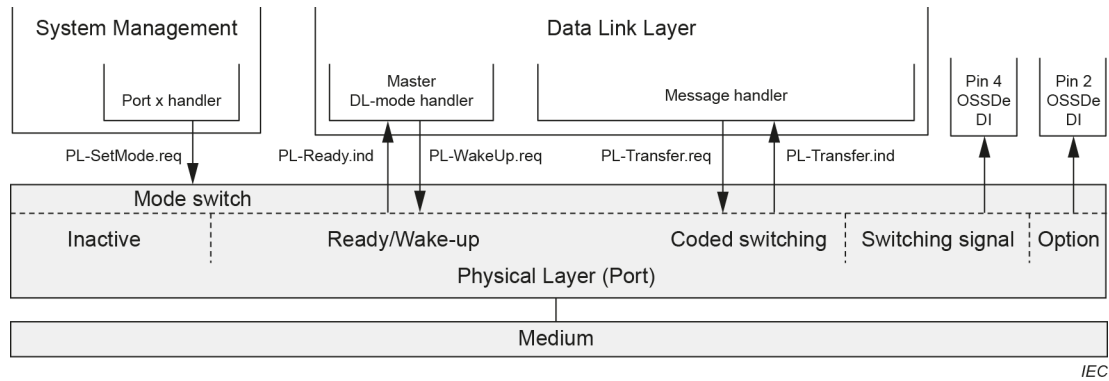


Figure 14 – The SDCI physical layer of an FS-Master (class A)

Pins 2 and 4 shall be scanned simultaneously to achieve OSSDe functionality. The FS-Master shall scan the C/Q line for the Ready signal of the FS-Device (see A.2.11).

Figure 15 shows the adapted physical layer of an FS-Device (class A).

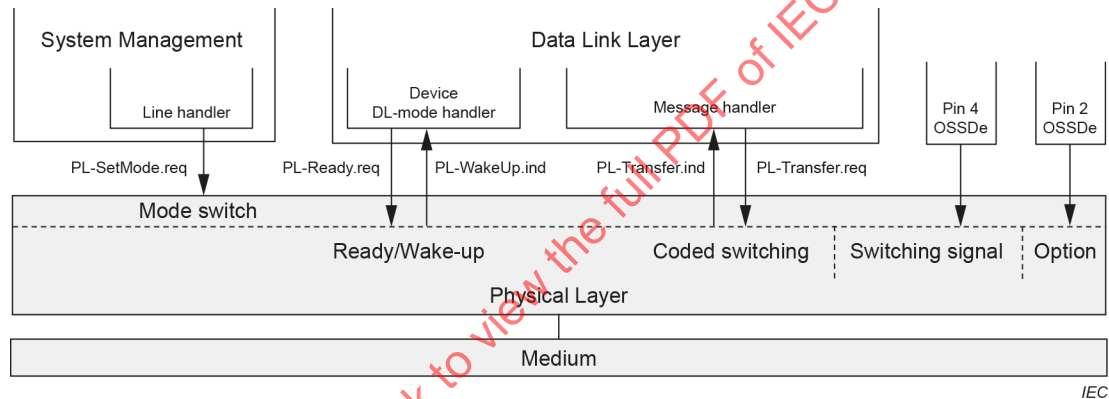


Figure 15 – The physical layer of an FS-Device (class A)

Pins 2 and 4 carry the OSSDe signals. FS-Device shall set the Ready pulse after internal testing.

5.2 Extensions to PL services

5.2.1 PL_SetMode

The PL-SetMode service is extended by the additional TargetMode "OSSDe" (C/Q line and I/Q line in digital input mode).

5.2.2 PL_Ready

The PL-Ready service initiates or indicates a Ready signal on the C/Q line. Whenever the FS-Device finished its internal safety-related hardware and software tests, it sets this signal. The FS-Master polls this signal and upon reception initiates the wake-up sequence. This unconfirmed service has no parameters. The service primitives are listed in Table 3.

Table 3 – PL_Ready

Parameter name	.req	.ind
<none>		

5.3 Transmitter/receiver

5.3.1 Assumptions for the expansion to OSSDe

Figure 16 shows the cross compatibility between OSSD based safety sensors and OSSDe based FS-Devices.

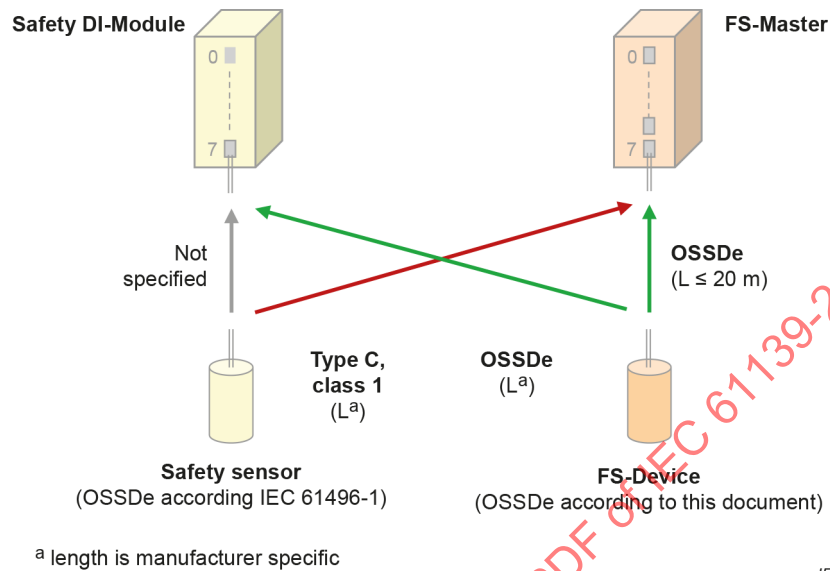


Figure 16 – Cross compatibility OSSD and OSSDe

The following assumptions are the basis for the design of the OSSDe expansion:

- The SIO paradigm of SDCI shall apply for SDCI-FS in order to allow manufacturers the combined function of OSSDe and SDCI-FS communication within one FS-Device.
- A Port on the FS-Master (with "FS-DI" according to Figure 9) shall have fixed configurations as either SDCI-FS or OSSDe interface with no or minor adjustments in respect to addressing, watchdog times, discrepancy times, or filter times.
- In order to allow OSSD based sensors on the market to be connected to the FS-Master, the FS-DI interface shall support the necessary adjustments for Type "C", class "1" devices according to [7].
- The OSSDe interface shall only be designed as input for the FS-Master port (safety sensors, Class A connectors). Most actuators are supplied by three-phase alternating current such as power drives, low voltage switch gears, motor starters, etc.
- Actuators such as valves with diversity and relays shall be supported by FS-Master with Ports "level d" (see Clause 6).

5.3.2 OSSDe specifics

5.3.2.1 General

Similar to the SIO approach, FS-Master according to level "c" support connectivity to existing functional safety devices with OSSDe. OSSDe in this document is defined as two outputs with signals that are both switching in equivalent manner as opposed to antivalent manner, where one signal is normally off and the other normally on (OSSDm).

The FS-Master port is designed to achieve a maximum of possible compatibility to existing OSSD devices using interface type C, class 1 defined in [7].

Figure 17 shows a corresponding reference model from [7], adapted to SDCI-FS. The information-"source" on the left corresponds for example to a sensor device, whereas the information-"sink" on the right side represents an input of the FS-Master Port class A. Power is supplied by the sink.

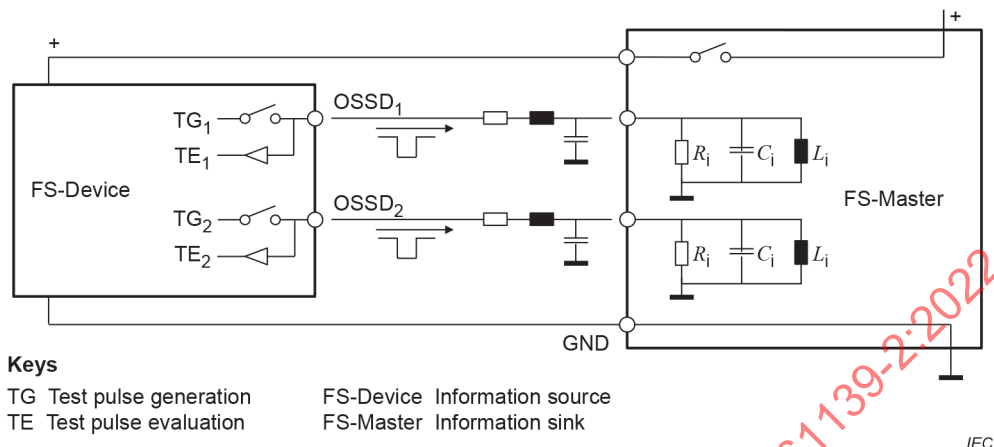


Figure 17 – Principle OSSDe function

The worst-case values for the line resistance and capacitance are defined in Table 9. In case of SDCI-FS, line inductance is negligible at a length of 20 m. The design of the FS-Master Port shall ensure values for R_i , C_i , and L_i guaranteeing proper signal behavior according to Table 8.

Table 4 shows the OSSD states and conditions defined in IEC 61496-1:2020.

Table 4 – OSSD states and conditions

State	Cause	Voltage range	Current
OFF	Demand	- 3 V to + 2 Vrms (+ 5 V peak)	< 2 mA (leakage) ^a
ON	No demand	+ 11 V to + 30 V	> 6 mA
^a IEC 61131-9 permits 5 mA for the voltage range of 5 V to 15 V			

OFF state:

For this interface, the OFF state is defined as the "powerless" state, where voltage and current of at least one OSSDe shall be within (voltage) and below (current) defined limits (see Table 4). If the safety function is demanded, or the source (the device) detects a fault, the OSSDe signals shall go to the OFF state. Antivalent voltage levels, so-called discrepancy, on both OSSDe outputs of the device shall be treated as OFF state. The duration of this state shall be within a specified discrepancy tolerance time. If the tolerance time is exceeded, the port is considered to be faulty.

ON state:

For this interface, the ON state is defined as the "powered" state, where voltage and current on both OSSDe outputs shall be within the voltage range and above defined current limits, when sinked by IEC 61131-2 inputs (see Table 4). Test pulses within specified ranges in voltage levels, durations and intervals are permitted. Antivalent voltage levels, so-called discrepancy, on both OSSDe outputs of the device shall be treated as OFF state.

5.3.2.2 Detection of cross connection faults

Tests are required for the detection of the cross-connection faults specified in IEC 61496-1 and shown in Table 5.

Table 5 – Cross connection faults

Fault	Diagnostics
Short circuit between OSSD 1 and OSSD 2	Test pulses (runtime diagnosis)
Short circuit between OSSD 1 or OSSD 2 and V+	Test pulses (runtime diagnosis)
Short circuit between OSSD 1 or OSSD 2 and V-	Test pulses (runtime diagnosis)
Open circuit of the power supply return cable (V-)	Type test, maximum leakage current
Open circuit of the functional earth (bonding) conductor	Type test, no functional earth
Open circuit of the screen of screened cable	Not required due to no shielding
Incorrect wiring	Discrete wiring only, organizational issue (test during commissioning)

The means for detecting short circuits are test pulses at runtime. The means for testing the behavior in case of open circuits is the type test during the assessment. Figure 18 shows the test pulses approach for the detection of cross connection faults. A cross fault occurs if one channel detects both signal patterns.

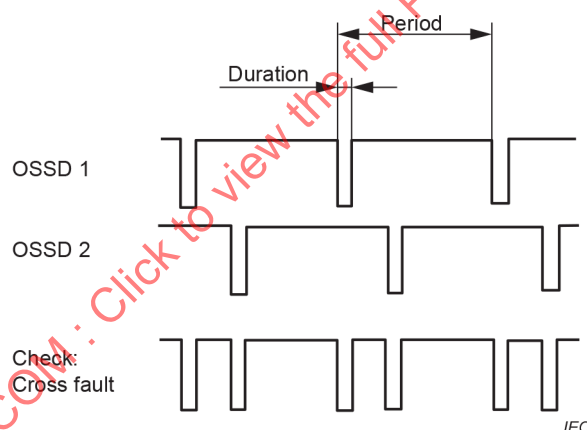


Figure 18 – Test pulses to detect cross connection faults

Three methods of testing (intervals) are commonly used:

- Test pulses at each program cycle of the safety device (dependency on configuration);
- Test pulses at fixed times;
- Test pulses after any commutation OFF → ON.

5.3.2.3 FS-Device OSSDe output testing

The test pulses of this interface type for testing the transmission line are created and evaluated on the safety device side. This way the source can diagnose the correct functioning of the output stage. In case of any detected error both OSSDe outputs shall be switched to the safe state (Lock-out condition = OFF).

The test pulses are created in a periodic manner on both OSSD lines. In order to detect short circuits between the lines or between the lines and power-supply, the test pulses of both lines can be time-shifted to each other (see Figure 19).

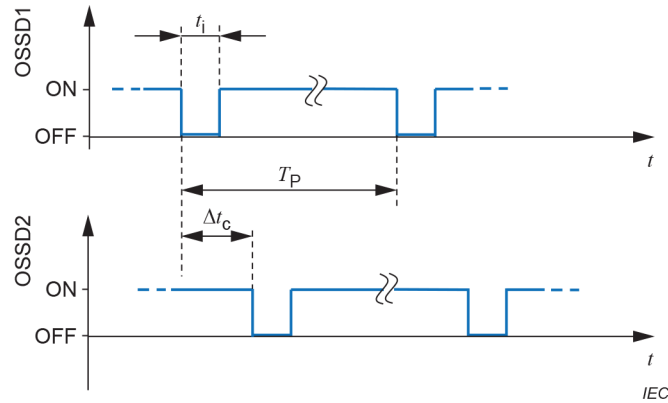


Figure 19 – OSSD timings

The following parameters specify the characteristics of the test pulses on the OSSD interface:

- Period of test pulses (T_P);
- Duration of test pulses (t_i);
- Time-shift between test pulses of both channels (Δt_c).

The characteristics of test pulses are classified in [7]. FS-Devices shall meet type C and class 1 requirements with a test pulse length $t_i \leq 1\,000\,\mu\text{s}$ (see Table 7).

5.3.3 Start-up of an FS-Device (Ready pulse)

Figure 20 shows the typical start-up sequence of an OSSD sensor without SDCI-FS capability. During self-test for functional safety, both OSSD signals shall be OFF. When finished, the sensor switches to ON and starts test pulses. A demand causes the sensor to switch OFF. A fault causes the sensor to switch to lock-out condition (OFF) and to remain in this state until repair.

NOTE For simplicity, Figure 20 shows only one OSSD channel.

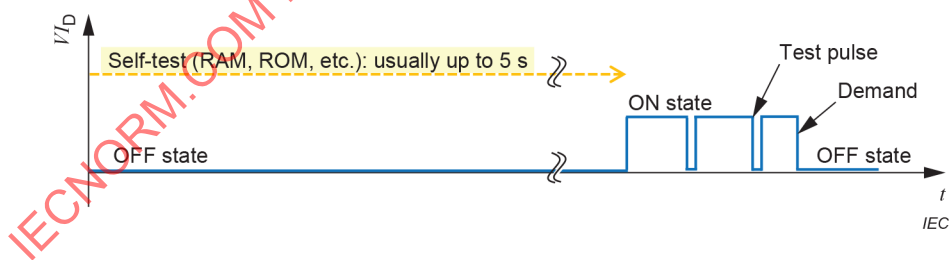


Figure 20 – Typical start-up of an OSSD sensor

Figure 21 shows the start-up of an FS-Device with OSSDe capability connected to a classic FS-DI module.

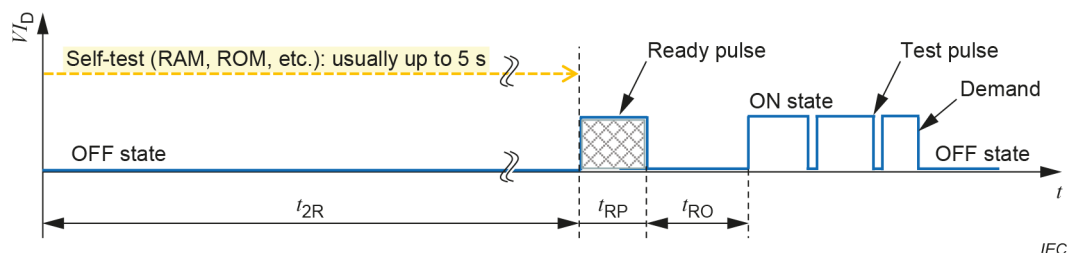


Figure 21 – Start-up of an FS-Device

In contrast to a classic sensor, the FS-Device provides only on pin 4 (see Figure 9) a so-called Ready-pulse of a certain length to indicate the FS-Master its readiness after self-testing. After a certain recovery time, the FS-Device switches to ON and starts test pulses like a classic safety sensor.

Timings and Wake-up behavior of the FS-Device are specified in 5.7.

5.3.4 Electric characteristics of a receiver in FS-Device and FS-Master

The voltage range and switching threshold definitions are the same for FS-Master and FS-Device since FS-Master ports shall be able to operate with non-safety SDCI Devices. The definitions in Table 6 in accordance with IEC 61131-9 apply.

Table 6 – Electric characteristics of a receiver

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
$V_{THH_{D,M}}$	Input threshold 'ON'	10,5	n/a	13	V	See NOTE 1
$V_{THL_{D,M}}$	Input threshold 'OFF'	8	n/a	11,5	V	See NOTE 1
$V_{HYS_{D,M}}$	Hysteresis between input thresholds 'ON' and 'OFF'	0	n/a	n/a	V	Shall not be negative See NOTE 2
$V_{IL_{D,M}}$	Permissible voltage range 'OFF'	$V_{0_{D,M}} - 1,0$	n/a	n/a	V	With reference to relevant negative supply voltage
$V_{IH_{D,M}}$	Permissible voltage range 'ON'	n/a	n/a	$V_{+_{D,M}} + 1,0$	V	With reference to relevant positive supply voltage.

NOTE 1 Thresholds are compatible with the definitions of type 1 digital inputs in IEC 61131-2.

NOTE 2 Hysteresis voltage $V_{HYS} = V_{THH} - V_{THL}$

Figure 22 demonstrates the switching thresholds for the detection of OFF and ON signals.

NOTE 'OFF' and 'ON' correspond to 'L' (Low) and 'H' (High) in IEC 61131-9.

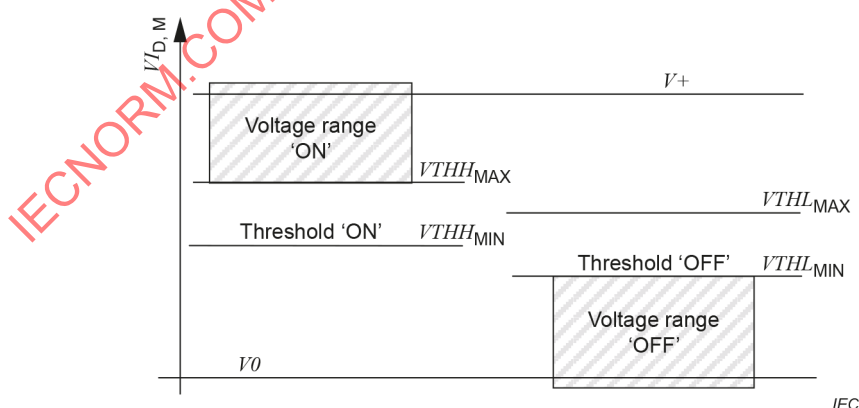


Figure 22 – Switching thresholds for FS-Device and FS-Master receivers

The FS-Master ignores pulses below 11 V (max. 15 mA or max. 30 mA) that are shorter than 1 ms.

5.4 Electric and dynamic characteristics of an FS-Device

In general, the specified values and ranges of IEC 61131-9 apply (see Figure 23).

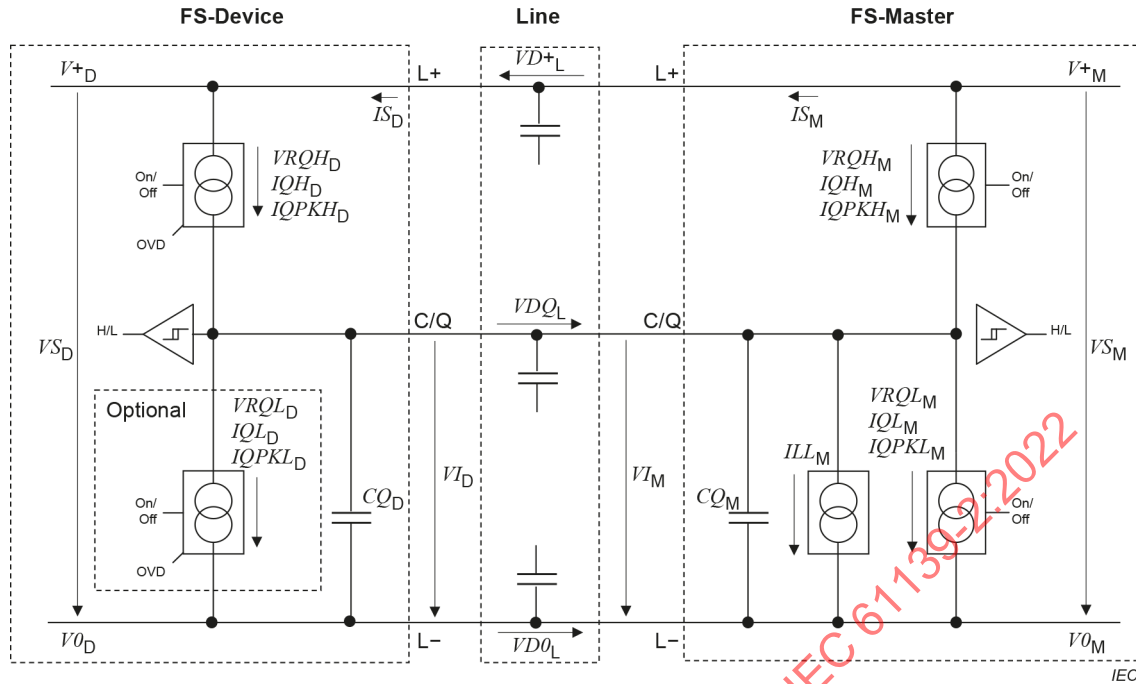


Figure 23 – Reference schematics (one OSSDe channel)

The subsequent illustrations and parameter tables refer to the voltage level definitions in Figure 24.

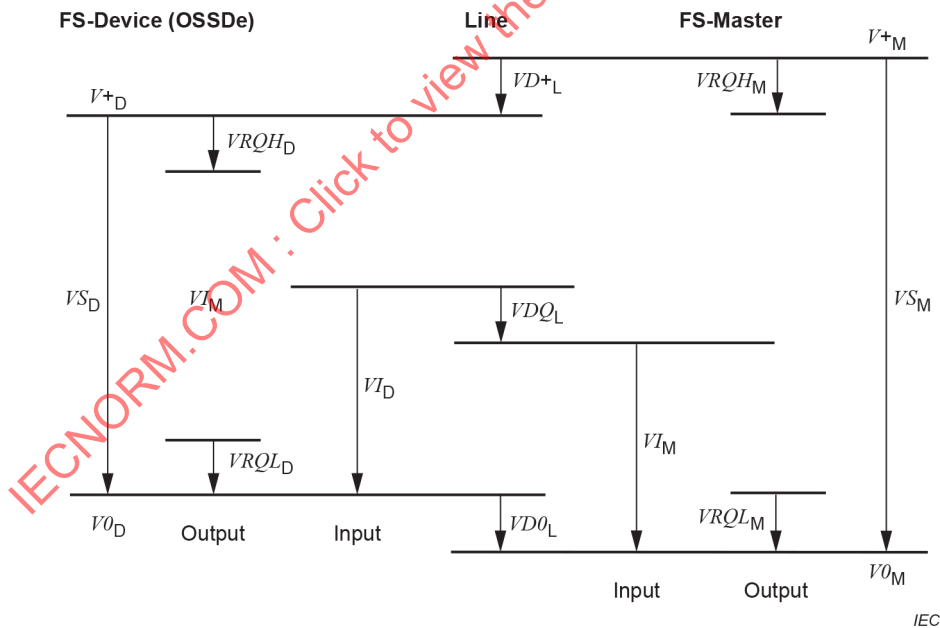


Figure 24 – Voltage level definitions

The electric and dynamic parameters for the OSSDe interface of an FS-Device are specified in Table 7 in accordance with IEC 61131-9.

Table 7 – Electric and dynamic characteristics of the FS-Device (OSSDe)

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
V_{SD}	Supply voltage	18	24	30	V	See Figure 24
ΔV_{SD}	Ripple	n/a	n/a	1,3	V _{pp}	Peak-to-peak absolute value limits shall not be exceeded. $f_{\text{ripple}} =$ DC to 100 kHz
I_{SD}	Supply current	n/a	n/a	1000	mA	See 5.9
Q_{ISD}	Power-up consumption	n/a	n/a	70	mAs	See Equation (1) and associated text
$VRQH_D$	Residual voltage 'ON'	n/a	n/a	3	V	Voltage drop compared with V_{+D} (see [9])
$VRQL_D$	Residual voltage 'OFF'	n/a	n/a	3	V	Voltage drop compared with V_{0D} NOTE 1
IQH_D	DC driver current P-switching output ('ON' state)	50	n/a	minimum (I_{QPKLM})	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 (I_{QPKHM})	mA	Only for push-pull output stages
IQQ_D	Quiescent current to V_{0D} ('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. See IEC 61131-9 for constraints on transmission rates.
t_{2R}	Time to Ready pulse	n/a	n/a	5 (default), > 5 to 655 as exception	s	See Figure 21 and A.2.11
t_{RP}	Duration of Ready pulse	500	n/a	1000	µs	See Figure 21
t_{RW}	End of Ready pulse to ready for Wake-up	n/a	n/a	50	µs	See Figure 27
t_{RO}	End of Ready pulse to OSSD mode	1,1	n/a	Data sheet	s	See Figure 21
t_i / T_P	Test pulse ratio	Data sheet	n/a	1/100	–	See [7] and Figure 19
t_i	Test pulse duration	n/a	n/a	1000	µs	See Figure 19
Δt_c	Time-shift	0	n/a	Data sheet	ms	See Figure 19
t_{disD}	Discrepancy time	n/a	n/a	1	ms	Demands may occur during tests

NOTE 1 Pull-down of residual voltage with deactivated high-side output driver stage and activated low-side driver stages (if available e.g. push-pull drivers) with externally limited DC driver current of 50 mA maximum.

NOTE 2 Characteristics in this table assume OSSD type "C", class "1" or class "2" according to [7] and interface type 1 according to IEC 61131-2.

It is the responsibility of the FS-Device designer to select appropriate ASICs according to IEC 61131-9 and/or to provide mitigating circuitry to meet the requirements of IEC 61496-1.

The FS-Device shall be able to reach a stable operational state (ready for Wake-up: T_{RDL}) while consuming the maximum charge (see Equation (1)).

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

5.5 Electric and dynamic characteristics of an FS-Master port (OSSDe)

In general, the specified values and ranges of IEC 61131-9 apply (see Figure 23 and Figure 24). The definitions in Table 8 are valid for the electrical characteristics of an FS-Master port.

Table 8 – Electric and dynamic characteristics of the Port interface

Property	Designation	Minimum	Typical	Maximum	Unit	Remark
V_{S_M}	Supply voltage for FS-Devices	20	24	30	V	See Figure 24
I_{S_M}	Supply current for FS-Devices	200	n/a	1000	mA	Rules in 5.9. shall be considered
$ISIR_M$	Current pulse capability for FS-Devices	400	n/a	n/a	mA	See Figure 25
ILL_M	Load or discharge current for 0 V < V_{I_M} < 5 V 5 V < V_{I_M} < 15 V 15 V < V_{I_M} < 30 V	0 5 5	n/a n/a n/a	15 15 15	mA mA mA	See NOTE 1
$VRQH_M$	Residual voltage 'H'	n/a	n/a	3	V	Voltage drop relating to V_{+M} at maximum driver current IQH_M
$VRQL_M$	Residual voltage 'L'	n/a	n/a	3	V	Voltage drop relating to V_{0M} 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
t_{disM}	Discrepancy time	n/a	n/a	3	ms	See NOTE 3 and Table 7
t_i	Test pulse duration	n/a	n/a	1	ms	See Figure 19
T_P	Period of test pulses	10	n/a	n/a	ms	See Figure 19

NOTE 1 Currents are compatible with the definition of type 1 digital inputs in IEC 61131-2. However, for the range 5 V < V_{I_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 (See IEC 61131-9:—, 5.3.3.3).

NOTE 3 Characteristics assume OSSD type "C", class "1" in [7]

The Master shall provide a charge of at least 20 mAs within the first 50 ms after a power-on time of 5 ms without any overload-shutdown (see Figure 25). After 50 ms the current limitations for I_{S_M} in Table 8 apply.

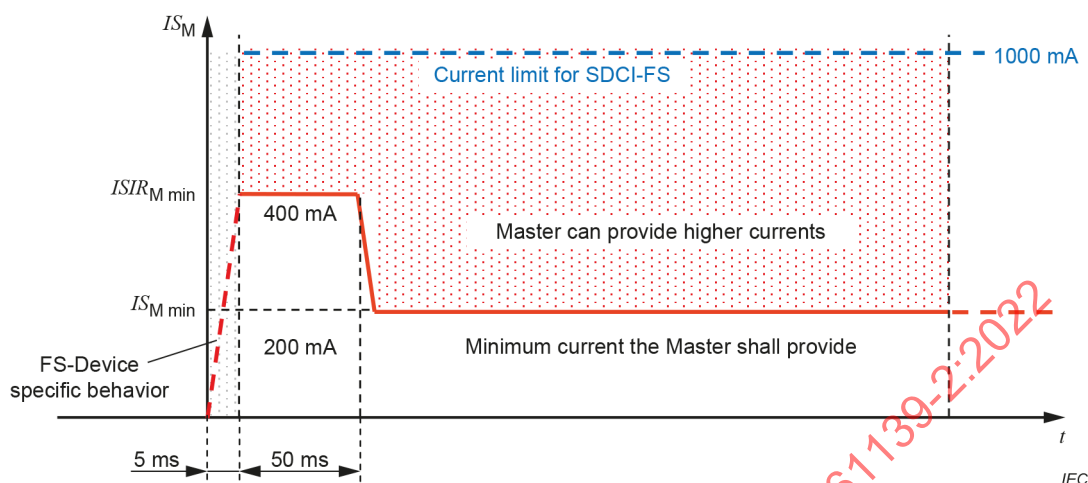


Figure 25 – Charge capability at power-up

5.6 FS-Master port FS-DI interface

Since OSSD safety sensors can provide different test pulse patterns, the FS-Master Port shall have a suitable input filter, or evaluation algorithm. For the sake of EMC considerations, a combination of both can be used. This means, that the time, in which the signal is below U_{Hmin} shall be less than the maximum allowed test pulse duration.

Any state different to both signals "high", except test pulses, shall be interpreted as safe state.

The EMC levels shall be considered for the layout of an input filter. The communication transmission rate 230 kbit/s conflicts with the input filter. Possible conflict resolution is shown in Figure 26.

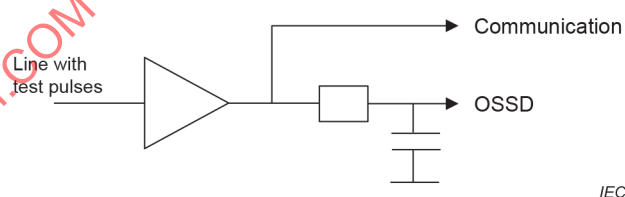


Figure 26 – OSSDe input filter conflict resolution

In general, the specified values and ranges of IEC 61131-9 apply. Basis is interface type 1 of IEC 61131-2. Deviating and supplementary electric and dynamic parameters for the FS-DI interfaces are specified in Table 8.

5.7 Wake-up coordination

Figure 27 shows the start-up of an FS-Device (see IEC 61131-9 for standard timing definitions). After accomplished self-tests, it indicates its readiness for Wake-up through an ON/Ready pulse on the C/Q line (see A.2.11). If no Wake-up occurs within a defined time frame, it starts with test pulses (see Figure 20).

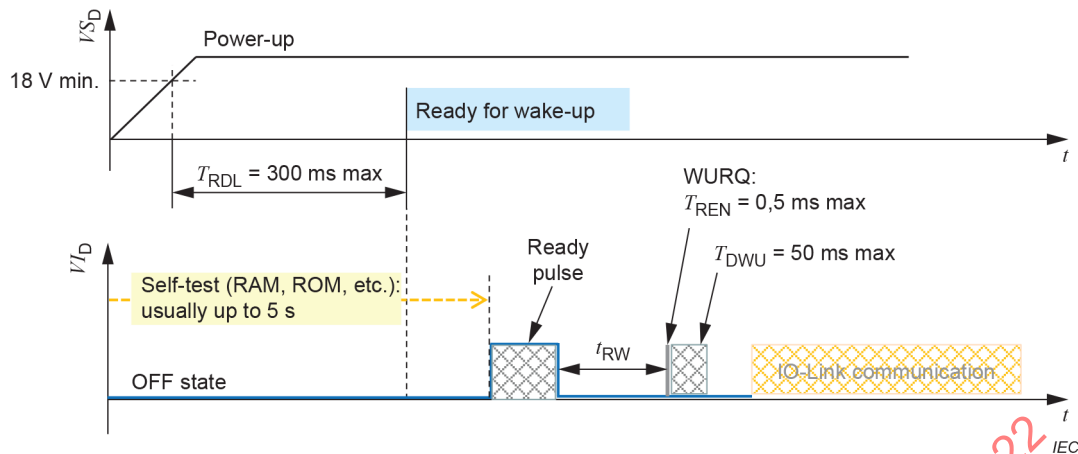


Figure 27 – Start-up of an FS-Device

NOTE Actually, some safety light curtain vendors offer activation of functionality if some connection conditions are activated during start-up phase (e.g. override).

5.8 Fast start-up

Figure 28 illustrates requirements for certain functional safety applications such as for a tool changer on a robot. A so-called fast start-up in non-safety cases shall be achieved within 0,5 s and in case of functional safety within 1 s.

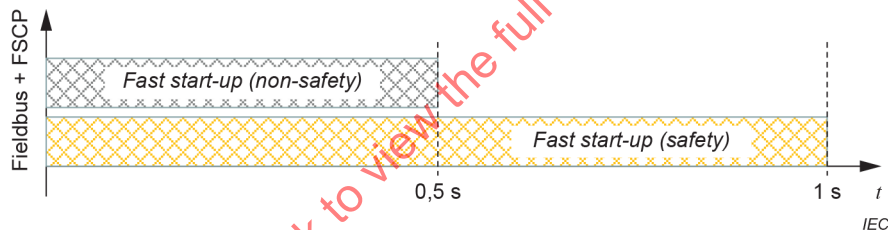


Figure 28 – Required fast start-up timings

NOTE Current safety devices usually need up to 5 seconds for self-testing prior to functional safe operation.

The Ready-pulse concept allows for easier achievable realizations of these requirements (see A.2.11).

5.9 Power supply

An FS-Master port shall be able to switch its power supply on and off. This enables the FS-Master to restart an FS-Device once it failed to establish communication and started OSSDe operation instead.

The FS-Master port is the only power supply for SDCI related parts of the FS-Device. Any external power source of the FS-Device shall be totally nonreactive to these parts.

FS-Master shall provide all ports with a minimum supply of 200 mA and at least one port with a minimum supply of 1 000 mA. The FS-Master shall specify the total maximum current consumption of all its ports and the derating rules.

Higher currents can conflict with the power switching components and cause interference with the signal lines. The "ripple" requirement in Table 7 shall be considered. The overall cable loop resistance shall be not more than 1,2 Ω (see Table 8 and Table 9).

5.10 Medium

5.10.1 Constraints

For the sake of simplicity in technology and commissioning, SDCI-FS expects a wired point-to-point connection or equivalent consistent transmission and powering between FS-Master and an FS-Device. No storing elements in between are permitted.

5.10.2 Connectors

Connectors as specified in IEC 61131-9 for Class A are permitted.

5.10.3 Cable characteristics

Table 9 shows the cable characteristics for SDCI-FS and non-safety Devices if higher power supply currents than 200 mA are applied.

Table 9 – Cable characteristics

Property	Designation	Minimum	Typical	Maximum	Unit
L	Cable length	0	n/a	20	m
RL_{eff}	Overall loop resistance	n/a	n/a	1,2	Ω
CL_{eff}	Effective line capacitance	n/a	n/a	3,0	nF (<1 MHz)
NOTE These characteristics can deviate from the original characteristics defined in IEC 61131-9.					

6 Extensions to SIO

SIO is only defined for Pin 4 of the Master/Device port in IEC 61131-9. OSSDe requires inclusion of Pin 2 as specified in Clause 5. Configuration can be performed within the Master/Device applications layer (see Figure 32 and Figure 34).

7 Extensions to the data link layer (DL)

7.1 Overview

Figure 32 and Figure 34 show the DL building blocks of FS-Device and FS-Master. No new or changed services are required. However, both DL-mode handlers are extended by the Ready-pulse feature as shown in 7.2 and 7.3.

7.2 State machine of the FS-Master DL-mode handler

Figure 29 shows the modifications of the FS-Master DL-mode handler versus the Master DL-mode handler in IEC 61131-9.

A new state "WaitOnReadyPulse_10" considers the requirement for the FS-Master to wait on the Ready-pulse of an FS-Device (see 5.7) prior to establish communication via DL_SetMode_STARTUP.

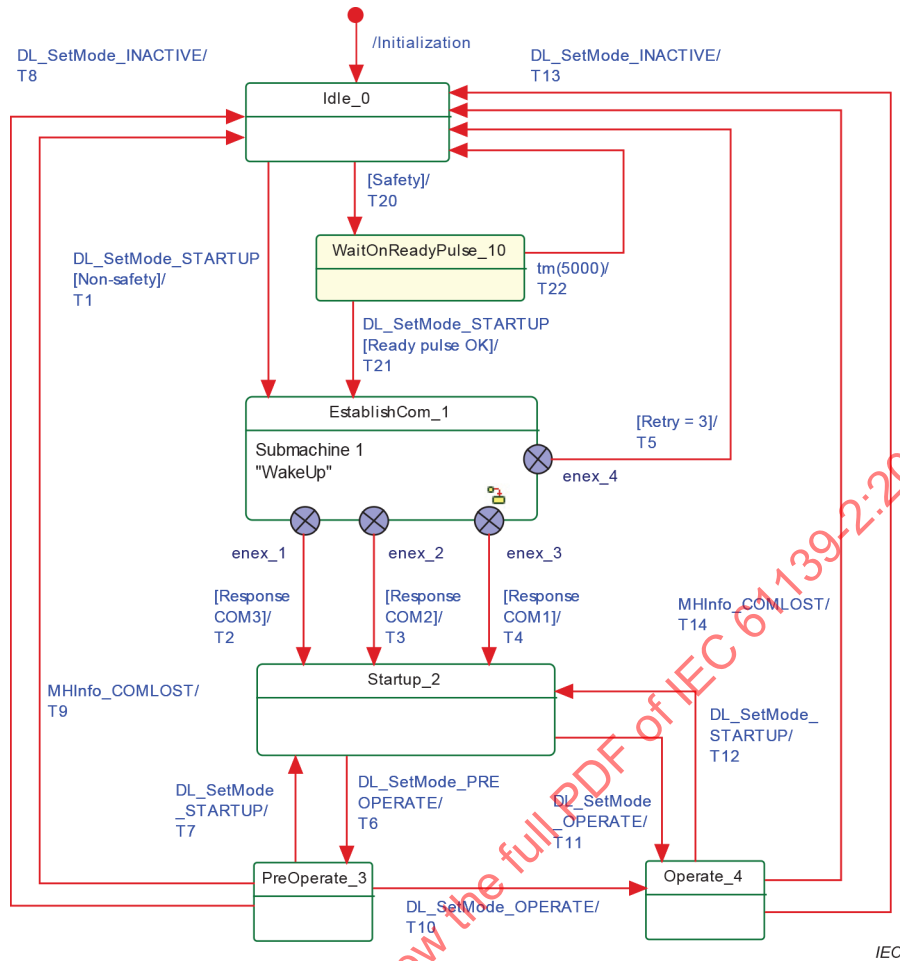


Figure 29 – State machine of the FS-Master DL-mode handler

The maximum waiting time is t_{2R} as defined in Table 7. Whenever the time expired, the FS-Master shall run a power-OFF/ON cycle for the connected FS-Device in order to initiate a retry for another Ready-pulse.

The criterion to use the extra path is the guard [safety], which is derived from the new port configuration "FS_PortModes" (see 10.4.2).

Table 10 shows the additional state and transitions as well as internal items considering the Ready-pulse feature.

Table 10 – State transition tables of the FS-Master DL-mode handler

STATE NAME		STATE DESCRIPTION	
Idle_0 to SM: Retry_9		See IEC 61131-9:—, Table 42	
WaitOnReadyPulse_10		Waiting on the Ready-pulse from FS-Device. A timer is started with the given value of the parameter FSP_TimeToReady within FSPortConfigList (see A.2.11 and 10.3.4).	
TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1 to T19	*	*	See IEC 61131-9:—, Table 42
T20	0	10	This path is taken only if the new configuration parameter "Safety" has been assigned to "SafetyCom" or "MixedSafetyCom" respectively
T21	10	1	Set Retry = 0.
T22	10	0	FS-Master was not able to detect a Ready-pulse within 5 s. It will initiate a power OFF/ON cycle for the FS-Device to retry the Ready-pulse.
INTERNAL ITEMS		TYPE	DEFINITION
MH_xxx to xx_Conf...		Call	See IEC 61131-9:—, Table 42
Safety		Guard	New configuration parameter "Safety": either value "SafetyCom" or "MixedSafetyCom"
Ready pulse OK		Guard	Ready pulse detected

7.3 State machine of the FS-Device DL-mode handler

Figure 30 shows the modifications of the FS-Device DL-mode handler versus the Device DL-mode handler in IEC 61131-9.

A new state "SelfTesting_5" considers the requirement for the FS-Device to indicate its readiness for a wake-up procedure after its internal safety self-testing via a test pulse in pin 4. Self-testing may actually take more than the maximum permitted start-up time T_{RDL} of a non-safety Device (see 5.7).

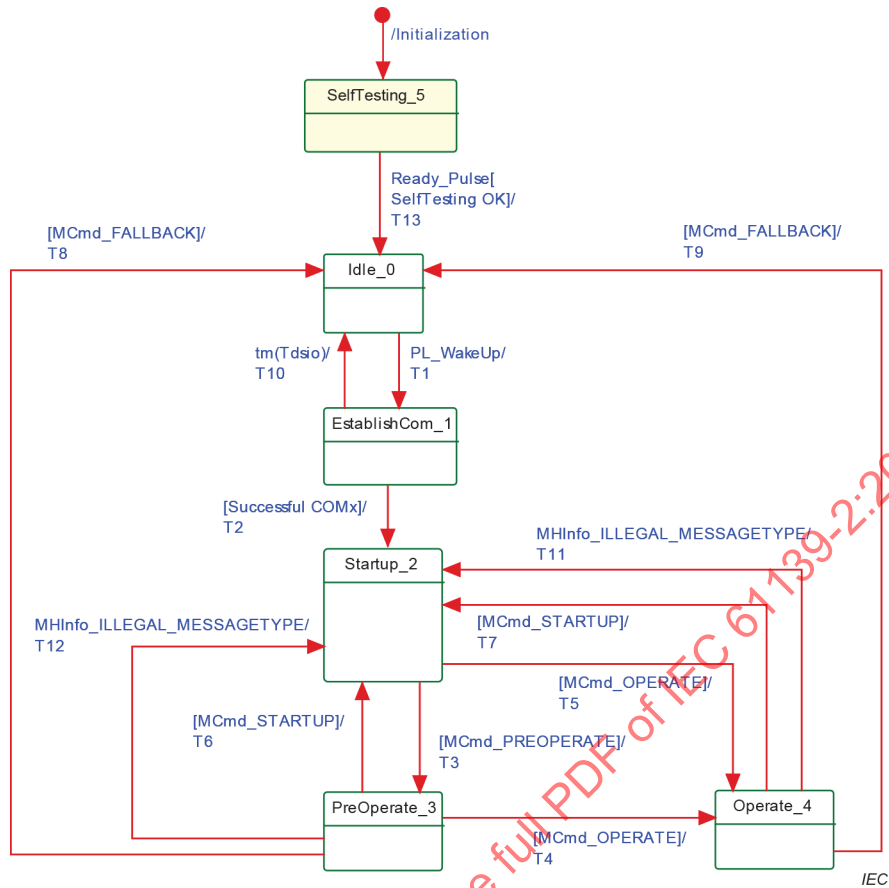


Figure 30 – State machine of the FS-Device DL-mode handler

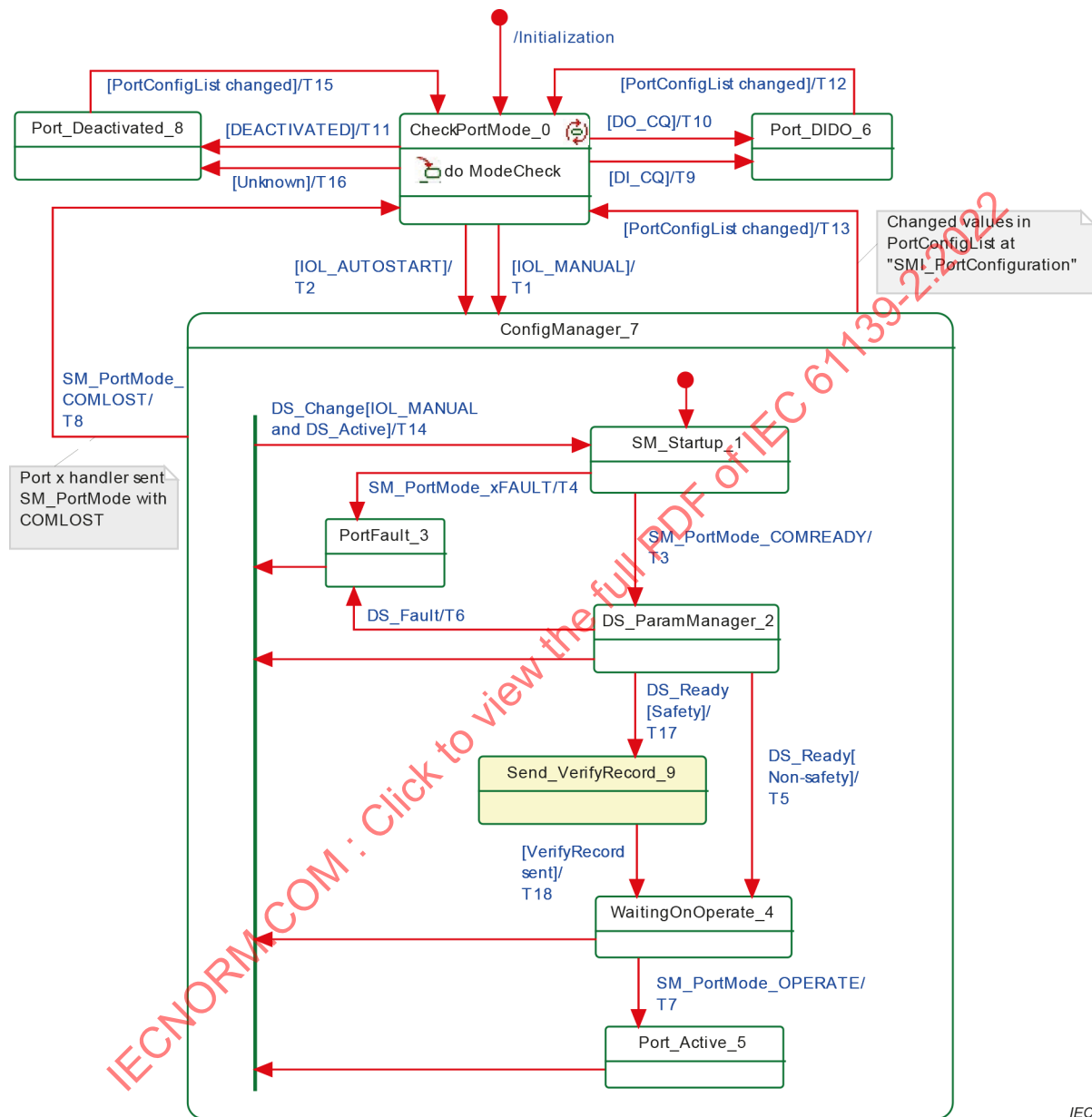
Table 11 shows the additional state and transitions as well as internal items considering the self-testing time of an FS-Device.

Table 11 – State transition tables of the FS-Device DL-mode handler

STATE NAME		STATE DESCRIPTION	
Idle_0 to Operate_4		See IEC 61131-9:—, Table 43	
SelfTesting_5		Safety check through self-testing of μ C, RAM, etc. This may take more than the permitted start-up time T_{RDL} of a non-safety Device (see A.2.11).	
TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1 to T12	*	*	See IEC 61131-9:—, Table 43
T13	5	0	Create a signal (Ready_Pulse) on pin 4 for duration of t_{RP} , when self-testing is completed (see t_{2R} in Table 7).
INTERNAL ITEMS		TYPE	DEFINITION
T_{RDL}		Time	See IEC 61131-9:—, Table 10
t_{RP}		Time	See Table 7
Self-testing OK		Guard	Self-testing completed

8 Extensions to the Master Configuration Manager (CM)

One part of the integrity measures is a verification record (VerifyRecord) an FS-Master sends to the FS-Device during start-up as explained in 11.7.4 and shown in Figure 57. This requires an extension to the Configuration Manager as shown in Figure 31.



IEC

Figure 31 – Extension to the Configuration Manager (VerifyRecord)

A new state "Send_VerifyRecord_9" considers the requirement for the FS-Master to send the VerifyRecord and Table 12 the additional state, transitions, and internal items.

Table 12 – State transition tables of the Configuration Manager

STATE NAME		STATE DESCRIPTION	
CheckPortMode_0 to Port_Deactivated_8		See IEC 61131-9:—, Table 126	
Send_VerifyRecord_9		FS_Master sends its stored FSP_VerifyRecord to "hidden" Index 0x4202 (see A.2.10)	
TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1 to T16	*	*	See IEC 61131-9:—, Table 43
T17	2	9	–
T18	9	4	SM_Operate
INTERNAL ITEMS		TYPE	DEFINITION
Safety		Guard	FS-Device in FS mode
Non-safety		Guard	FS-Device in non-FS mode

9 Extensions of the FS-Device

9.1 Principle architecture and models

9.1.1 FS-Device architecture

Figure 32 shows the principle architecture of the FS-Device. It does not include safety measures for implementation such as redundancy for the safety-related parts.

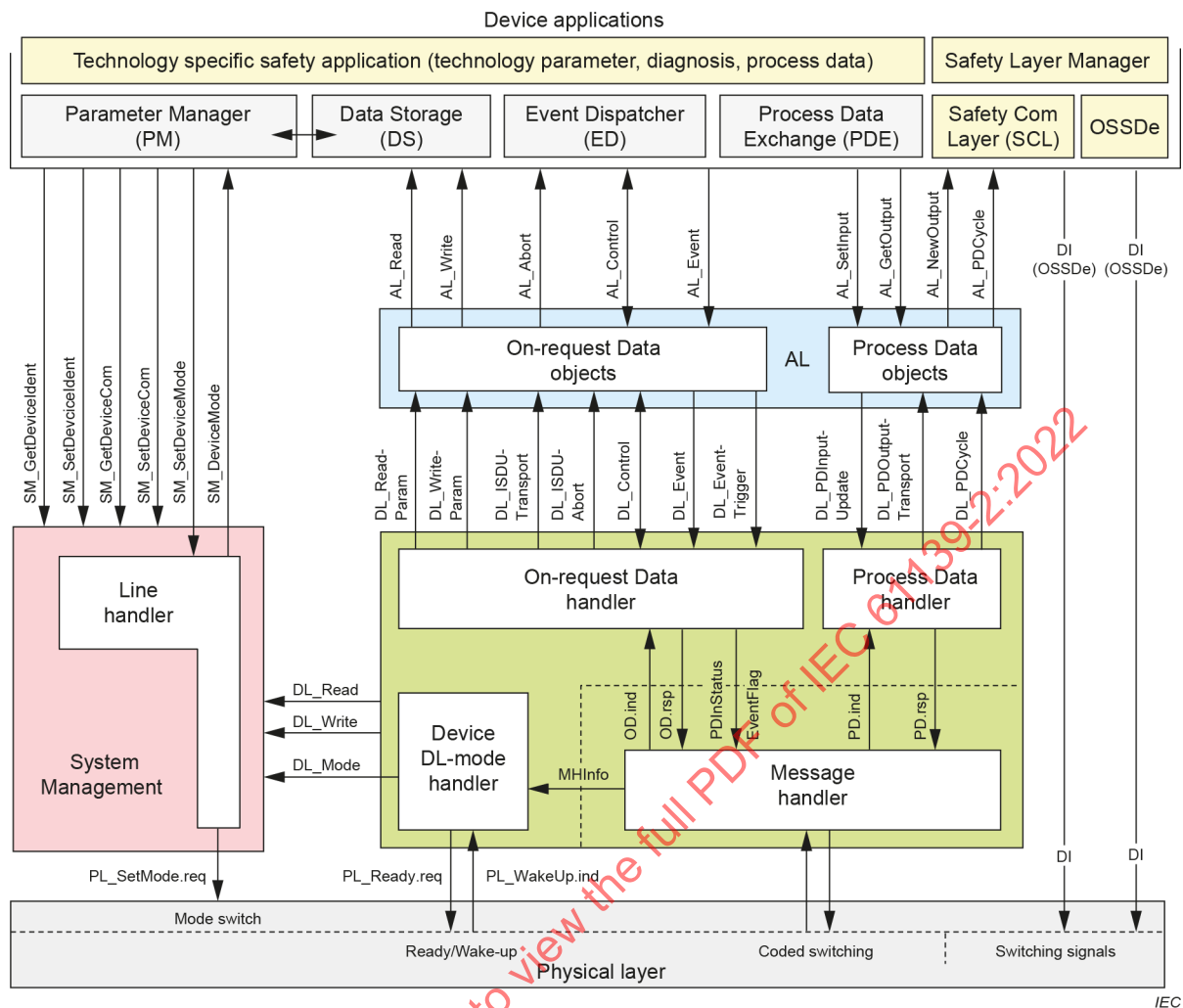


Figure 32 – Principle architecture of the FS-Device

An FS-Device comprises first the technology specific functional safety application. "Emergency switching off" safety devices for example can be designed such that "classic" OSSDe operation or safety communication can be configured. A Safety Layer Manager is responsible for the handling of a safety bit via the OSSDe building block or a safety PDU using the Safety Communication Layer (see Clause 11).

It checks correctness of parameters at each start-up of the FS-Device whenever the FSP_VerifyRecord has been written during PREOPERATE. The safety communication layer (SCL) engine will be started if all parameters are verified to be correct. Otherwise, an error message will be indicated and the SCL remains inactive or stops.

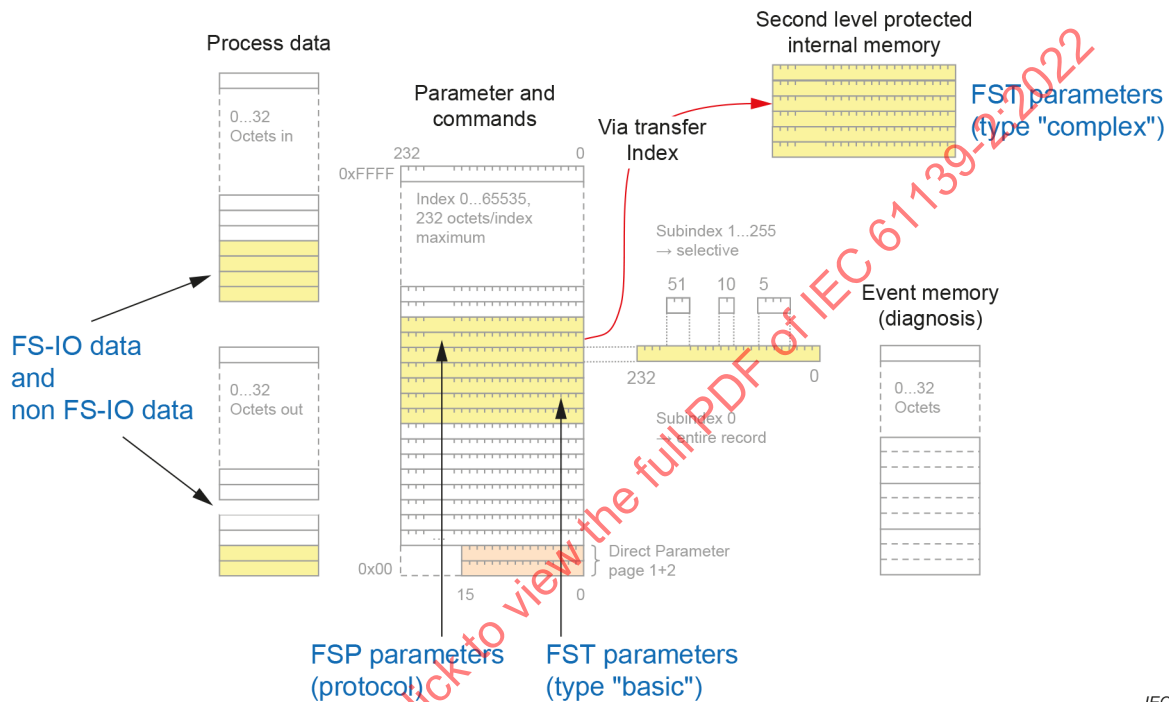
9.1.2 FS-Device model

According to the requirement of mixed NSR and SR parameter and process data, the FS-Device model has been modified and adapted.

That means the FS-Device Index model is split into an NSR and an SR part. Figure 33 shows the areas of concern. The allocation of the SR part ("FSP" parameter) is defined within the IODD of the FS-Device.

During commissioning, the assignment of FSP parameter values take place. These instance values are secured by CRC signatures and transferred as record to the FS-Master and to the FS-Device (see 11.7.4). At each start-up of an FS-Device, the stored entire verification record (VerifyRecord) in the FS-Master is transferred in a diverse manner and the FS-Device can check the locally stored instance parameter values for integrity via comparison and CRC signatures. This check includes technology specific "FST" parameters, which are not transferred at each start-up. The FS-Device displays its FSP parameters at predefined Indices (see Figure 33).

Technology specific parameters (FST) could be handled either in an open manner to a certain extend as standard non-safety parameters (see 11.7.8) or in a protected manner in hidden internal memory (see 11.7.9).



IEC

Figure 33 – The FS-Device model

The maximum space for FS-I/O data and non-FS-I/O data to share is 32 octets. The space shall be filled with FS-I/O data first followed by the non-FS-I/O data. The border is variable. Assuming a maximum safety protocol trailer of 6 octets, the maximum possible space for FS-I/O data is 25 octets.

9.2 Parameter Manager (PM)

There are no extensions or modifications of the Parameter Manager required.

9.3 Process Data Exchange (PDE)

Depending on "Safety" configuration, Process Data Exchange takes over or passes FS-Process Data (see 11.4.3 Safety PDU) from/to the Safety Layer Manager.

9.4 Data Storage (DS)

9.4.1 General considerations and extensions including safety

The technology specific (FST) parameters are secured by a particular CRC signature (FSP_TechParCRC) included in the FSP parameter set. Additional authenticity parameters are used in case of FS-Device replacement.

The Data Storage mechanism for FS-Devices is based on the general mechanism for non-safety-related Devices as specified in IEC 61131-9. This version of Data Storage requires that Device Access Lock (Index 0x000C) bit "0" and "1" shall always be unlocked (= "0").

A small extension is required to the Data Storage (DS) state machine of the FS-Device with respect to the "Back-to-box" mechanism (authenticity values = "0"). Transition T8 in IEC 61131-9:—, Table 100 considers this check as shown in Table 13.

Table 13 – Extension to Data Storage (DS) state machine

TRANSITION	SOURCE STATE	TARGET STATE	ACTION
T1 to T7	*	*	See IEC 61131-9:—, Table 100
T8	2	3	In case of Safety: If back-to-box conditions are incorrect, return 0x8022 – <i>service temporarily not available – Device control</i> . If correct: Lock local parameter access, set State_Property = "Upload" or "Download"
T9 to T11	*	*	See IEC 61131-9:—, Table 100

9.4.2 Backup levels

Table 14 lists the Data Storage backup levels specified in IEC 61131-9. Subclause 9.4.2 describes some specialties to be considered for functional safety.

Table 14 – Data Storage Backup Levels

Backup Level	Behavior
Commissioning ("Disable")	Any change of active parameters within the Device will <i>not</i> be copied/saved. Device replacement <i>without</i> automatic/semi-automatic Data Storage.
Production ("Backup/Restore")	Changes of active parameters within the Device will be copied/saved. Device replacement <i>with</i> automatic/semi-automatic Data Storage supported.
Production ("Restore")	Any change of active parameters within the Device will <i>not</i> be copied/saved. If the parameter set is marked to be saved, the "frozen" parameters will be restored by the Master. However, Device replacement <i>with</i> automatic/semi-automatic Data Storage of <i>frozen parameters</i> is supported.

In case of functional safety, commissioning cannot be completed without verification and validation of FSP and FST parameters as well as of entire safety functions according to the relevant safety manuals.

The difference to "Criteria for backing up parameters" in IEC 61131-9 is that in case of local parameter modifications (for example by means of teach-in or touch panel), the FSP_TechParCRC shall be assigned with the help of the FS-Master tool and "Dedicated Tool".

10 Extensions of the FS-Master

10.1 Principle architecture

Figure 34 shows the principle architecture of the FS-Master offering the extended Standard Master Interface (SMI) according to IEC 61131-9. It allows for a stringent separation of the standard Master as "Black Channel" and the functional safety parts of SDCI-FS and an FSCP x that can be "encapsulated" within the Gateway Application layer.

Table 15 – SMI services used for FS-Master

Service name	Master	ArgBlockID	Remark
SMI_MasterIdentification	R	0x0001	–
SMI_FSMasterAccess	R	0x0100	See 10.2.2
SMI_PortConfiguration	R	0x8100	See IEC 61131-9
SMI_ReadbackPortConfiguration	R	0x8100	See IEC 61131-9
SMI_PortStatus	R	0x9100	See IEC 61131-9
SMI_DSToParServ	R	0x7000	Data Storage to parameter server
SMI_ParServToDS	R	0x7000	Data Storage from parameter server
SMI_DeviceWrite	R	0x3000	ISDU transport
SMI_DeviceRead	R	0x3001	ISDU transport
SMI_PortPowerOffOn	R	0x7003	–
SMI_DeviceEvent	I	–	–
SMI_PortEvent	I	–	–
SMI_PDIn	R	0x1001	–
SMI_PDOut	R	0x1002	–
SMI_PDInOut	R	0x1003	–
SMI_SPDUIn	R	0x1101	See 10.2.3
SMI_SPDUOut	R	0x1102	See 10.2.4
SMI_FSPDInOut	R	0x1103	See 10.2.5
SMI_PDInIQ	R	0x1FFE	–
SMI_PDOutIQ	R	0x1FFF	–
Key I Initiator of service R Receiver (Responder) of service			
			 yellow marked services are either extended or additional ones for SDCI-FS

Figure 35 provides an overview of the important SMI services used for FS-Master, the safety layers within the Gateway and details of the FS-Master applications.

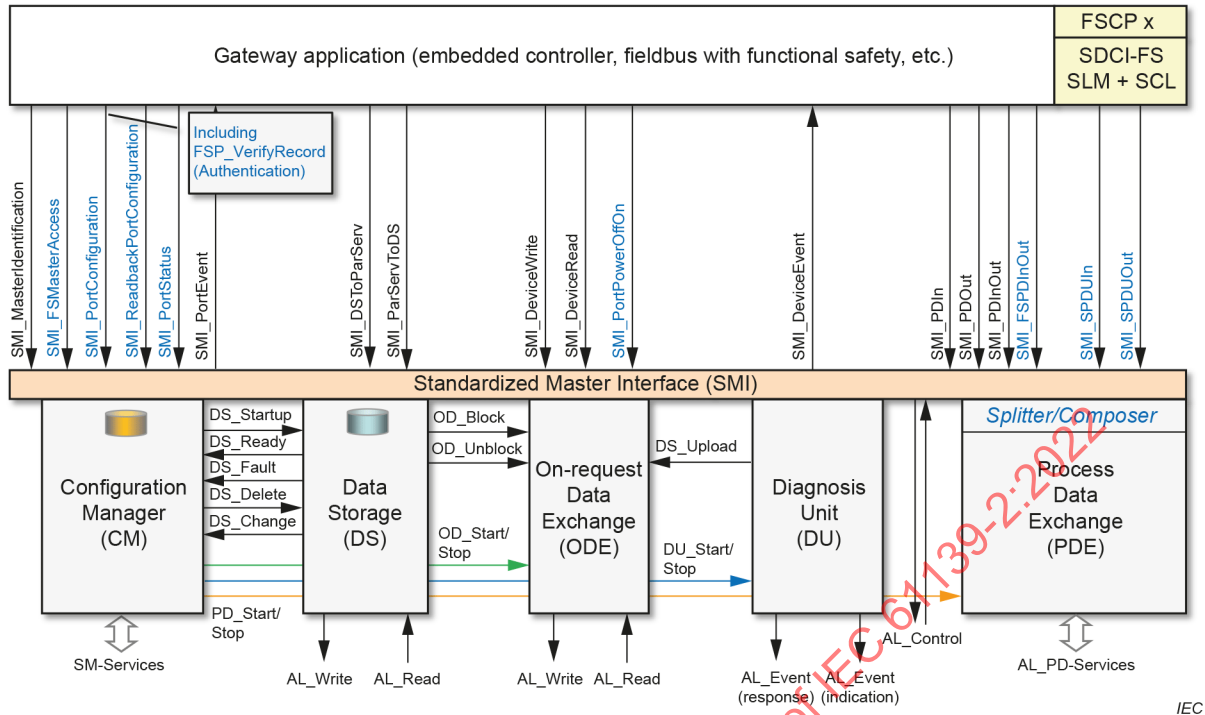


Figure 35 – SMI service extensions

The SMI_MasterIdentification presents as MasterType an FS-Master (= "3" according to IEC 61131-9). The corresponding SMI_FSMasterAccess service provides the FSCP Authenticity codes of the FS-Master being an FSCP device on a safety fieldbus. The SMI services for configuration and port status are only expanded by using different Arguments (ArgBlocks) as shown in 10.3. By means of the SMI service "SMI_PortConfiguration", for example, the authenticity, protocol, and IO data structure information is transferred to the Configuration Manager and stored there. See 10.4 on how this information is used to accommodate the Safety Communication Layers and to authenticate safety operation. The Port command service "SMI_PortPowerOffOn" is responsible for switching OFF and ON power of a particular port.

Two new SMI services provide access to the safety parts of a mixed SR and NSR I/O process data structure as shown in 10.2.3, 10.2.4, and 10.5.

10.2.2 SMI_FSMasterAccess

User role and corresponding password can be provided to the FS-Master safety projects and MasterType specific information can be retrieved by this SMI service (see Figure 35).

Table 16 shows the structure of the service following the rules defined in IEC 61131-9.

Table 16 – SMI_FSMasterAccess

Parameter name	.req	.cnf
Argument		
ClientID	M	
PortNumber	M	
ExpArgBlockID (0x0101)	M	
ArgBlockLength	M	
ArgBlock ("FSMasterAccess": 0x0100)	M	
Result (+)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID of request ArgBlock 0x0100)		M
ArgBlockLength		M
ArgBlock (associated to ExpArgBlockID)		M
Result (-)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID of request ArgBlock 0x0100)		M
ArgBlockLength		M
ArgBlock (JobError: 0xFFFF)		M

Argument

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

ClientID**PortNumber****ExpArgBlockID**

This parameter contains the ArgBlockID of "FSCPAAuthenticity" (0x0101, see Table 20)

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock

ArgBlock

This parameter contains the ArgBlock "FSMasterAccess" (0x0100, see Table 19)

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 (0x0100)

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Table 20)

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 (0x0100)

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF, see IEC 61131-9)

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)

10.2.3 SMI_SPDUIn

This service allows for cyclically reading Safety Protocol Data Units (SPDU) from an FSInBuffer (see 10.5) and shall only be available for internal use by the SCL. Coding of this SMI service follows the definitions in IEC 61131-9.

The expected ArgBlockID is "0x1101". The ArgBlock is specified in 10.3.6.

10.2.4 SMI_SPDUOut

This service allows for cyclically writing Safety Protocol Data Units (SPDU) to an FSOutBuffer (see 10.5) and shall only be available for internal use by the SCL. Coding of this SMI service follows the definitions in IEC 61131-9.

The ArgBlockID is "0x1102". The ArgBlock is specified in 10.3.7.

10.2.5 SMI_FSPDUInOut

This service allows for periodically reading input from an FSInBuffer and InBuffer and output from an FSOutBuffer and OutBuffer (see 10.5). Table 17 shows the structure of the service.

Table 17 – SMI_FSPDInOut

Parameter name	.req	.cnf
Argument		
ClientID	M	
PortNumber	M	
ExpArgBlockID (0x1103)	M	
ArgBlockLength	M	
ArgBlock (VoidBlock: 0xFFFF0)	M	
Result (+)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID of request ArgBlock 0xFFFF0)		M
ArgBlockLength		M
ArgBlock (associated to ExpArgBlockID)		M
Result (-)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID of request ArgBlock 0xFFFF0)		M
ArgBlockLength		M
ArgBlock (JobError: 0xFFFF)		M

Argument

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

ClientID**PortNumber****ExpArgBlockID**

This parameter contains the ArgBlockID of "FSPDInOut" (0x1103, see Table 25)

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock

ArgBlock

This parameter contains the ArgBlock "VoidBlock" (0xFFFF0)

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 (0xFFFF0)

ArgBlockLength

This parameter contains the length of the subsequent ArgBlock

ArgBlock

This parameter contains the ArgBlock associated to the ExpArgBlockID (see Table 25)

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 (0xFFFF0)

ArgBlockLength

This parameter contains the length of the "JobError" ArgBlock

ArgBlock

This parameter contains the ArgBlock "JobError" (0xFFFF)

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)

10.3 ArgBlock extensions

10.3.1 Overview

Table 18 shows six new ArgBlock types for FS-Masters: "FSMasterAccess", "FSCPAuthenticity", "FSPortConfigList", "FSPortStatusList", "SPDUIn", "SPDUOut", and "FSPDInOut".

Table 18 – ArgBlock types and ArgBlockIDs

ArgBlock type	ArgBlockID	Remark
FSMasterAccess	0x0100	See 10.3.2
FSCPAuthenticity	0x0101	See 10.3.3
PDIn	0x1001	–
PDOut	0x1002	–
PDInOut	0x1003	–
SPDUIn	0x1101	See 10.3.6
SPDUOut	0x1102	See 10.3.7
FSPDInOut	0x1103	See 10.3.8
DS_Data	0x7000	Data Storage object
PortPowerOffOn	0x7003	–
PortConfigList	0x8000	–
FSPortConfigList	0x8100	See 10.3.4
PortStatusList	0x9000	–
FSPortStatusList	0x9100	See 10.3.5
Key  yellow marked ArgBlocks are additional ones for SDCI-FS		

10.3.2 FSMasterAccess

The ArgBlock "FSMasterAccess" in Table 19 shows the password for FS-Master access and the corresponding password to reset the entire FS-Master project including the existing password.

Table 19 – FSMasterAccess

Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x0100
2	FSMasterPassword	This element contains the password for accessing the entire FS-Master project. Default is 0x0000.	Unsigned32	–
6	FSResetMasterPW	This element contains the password for resetting the entire FS-Master project including the FSMasterPassword. Default is 0x0000.	Unsigned32	–
10	FSUserRole	Reserved. Default: 0x00	Unsigned8	–

10.3.3 FSCPAuthenticity

The ArgBlock "FSCPAuthenticity" in Table 20 shows FSCP authenticity codes assigned to the FS-Master through the upper-level FSCP engineering tool or via DIP switches.

Table 20 – FSCPAuthenticity

Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x0101
2	FSP_Authenticity1	FSCP A-Code part1	Unsigned32	–
6	FSP_Authenticity2	FSCP A-Code part2	Unsigned32	–

10.3.4 FSPortConfigList

Table 21 shows the ArgBlockType "FSPortConfigList" suitable for FS-Masters. It considers additional PortModes and expands by Safety PDU lengths, the FSP_VerifyRecord (see 10.3.3 and A.2.10) as well as the FS I/O data structure description (see 11.7.7 and Table A.4).

Table 21 – FSPortConfigList

Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x8100
2	PortMode	This element contains the port mode expected by the SMI client, e.g. gateway application. All modes are mandatory, except OSSDe. They shall be mapped to the Target Modes of "SM_SetPortConfig" (see IEC 61131-9). 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: SAFETYCOM (implying IOL_MANUAL behavior) 50: MIXEDSAFETYCOM (implying IOL_MANUAL behavior) 51: OSSDE (Pin 2 + Pin 4 at M12) (Not for FS-Masters of type Port Class B; Values in offset 14 to 35 are don't care; SPDULength in offset 38 = 1 octet; value in offset 39 is don't care) 52 to 96: Reserved for extensions such as Safety or Wireless) 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)	This element defines the behavior of the I/Q signal (Pin2 at M12 connector). All assignments are "don't care" if PortMode is chosen to be OSSDE. 0: Disabled 1: Digital Input 2 to 4: Reserved 5: Power 2 (Port Class B) 6 to 255: Reserved	Unsigned8	0 to 0xFF

Offset	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" 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 IEC 61131-9:—, 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	Unsigned16	1 to FFFF
8	DeviceID	This element contains the 3 octets long DeviceID expected by the SMI client	Unsigned32	1 to FFFFFFF
12	FSP_TimeToReady	This element provides the time from power-up to the Ready pulse of the FS-Device such that the FS-Master knows how long to wait on it. Default maximum time is 5 s (see A.2.11)	Unsigned16	1 to 7FFF (see Table A.1)
14	FSP_MinShutDown Time	This element provides the minimum time for shut down of the FS-Device after Port power off prior to a restart	Unsigned16	64 to 03E8 (see Table A.1)
16	FSP_Authenticity1	FSCP A-Code part1 (see IEC 61784-3 series [1])	Unsigned32	–
20	FSP_Authenticity2	FSCP A-Code part2 (see IEC 61784-3 series [1])	Unsigned32	–
24	FSP_Port	Port number	Unsigned8	1 to 0xFF
25	FSP_AuthentCRC	CRC signature across complete authentication	Unsigned16	–
27	FSP_ProtVersion	Version of the SDCI-FS protocol	Unsigned8	1 to 0xFF
28	FSP_ProtMode	SDCI-FS protocol mode	Unsigned8	–
29	FSP_WatchdogTime	Watchdog time of FS-Master and FS-Device	Unsigned16	1 to FFFF
31	FSP_IO_StructCRC	CRC signature across FS IO data description	Unsigned16	–
33	FSP_TechParCRC	CRC signature across technology parameter	Unsigned32	–
37	FSP_ProtParCRC	CRC signature across protocol parameter	Unsigned16	–
39	IO_DescVersion	Version of this generic structure description	Unsigned8	1
40	SPDUInLength	OSSDe data (1 octet) or length of incoming SPDU (m); see 10.5 and Table A.4	Unsigned8	1 or 5 to (32 – n) octets
41	TotalOfInBits	Set of input BooleanT (bits)	Unsigned8	0 to 0xFF
42	TotalOfInOctets	Set of input BooleanT (octets)	Unsigned8	–
43	TotalOfInInt16	Input IntegerT(16)	Unsigned8	–
44	TotalOfInInt32	Input IntegerT(32)	Unsigned8	–
45	SPDUOutLength	Length of outgoing SPDU (o); see 10.5 and Table A.4	Unsigned8	5 to (32 – p) octets
46	TotalOfOutBits	Set of output BooleanT (bits)	Unsigned8	0 to 0xFF
47	TotalOfOutOctets	Set of output BooleanT (octets)	Unsigned8	–
48	TotalOfOutInt16	Output IntegerT(16)	Unsigned8	–
49	TotalOfOutInt32	Output IntegerT(32)	Unsigned8	–
^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 don't care except for "InputDataLength" and "OutputDataLength".				

10.3.5 FSPortStatusList

Table 22 shows the ArgBlockType "FSPortStatusList" suitable for FS-Masters. It allows only for the status report of the "Black Channel" part of the FS-Master. Content of "FSPortStatusInfo" shall be derived from "PortMode" in IEC 61131-9.

Table 22 – FSPortStatusList

Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x9100
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. 1: DEACTIVATED Port configuration DEACTIVATED is set. 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" is set. 6: DO_C/Q Port configuration "DO" is set. 7: OSSDE (DI at C/Q and I/Q) 8: SCL_ENABLED (Port ready for safety data exchange) 9 to 253: Reserved 254: PORT_POWER_OFF Shutdown of Port is active caused by SMI_PortPowerOffON 255: NOT_AVAILABLE (Port status currently not available)	Unsigned8 (enum)	0 to 0xFF
3	PortQualityInfo	This element contains status information on Process Data 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 0: NOT_DETECTED (No communication at that port) <>0: Copied from Direct Parameters page, address 4	Unsigned8	0 to 0xFF

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.	Unsigned8	–
7	InputDataLength	This element contains the input data length as number of octets of the Device provided by the PDIn service	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	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 FFFFFFF
15	NumberOfDiags	This element contains the number x of diagnosis entries (DiagEntry0 to DiagEntry x)	Unsigned8	0 to 0xFF
16	DiagEntry0	This element contains the "EventQualifier" and "EventCode" of a diagnosis (Event).	Struct Unsigned8/16	–
19	DiagEntry1	Further entries up to x if applicable...	...	–

10.3.6 SPDUIIn

Table 23 shows the structure of the ArgBlock "SPDUIIn" as illustrated in 10.5.

Table 23 – SPDUIIn

Offset	Element name	Definition	Data type
0	ArgBlockID	0x1101	Unsigned16
2	SPDUIIn0	Safety Protocol Data Unit in (octet 0)	Unsigned8
3	SPDUIIn1	Safety Protocol Data Unit in (octet 1)	Unsigned8
...			
SPDUIInLength + 2	SPDUIIn m	Safety Protocol Data Unit in (octet m)	Unsigned8

10.3.7 SPDUIOut

Table 24 shows the structure of the ArgBlock "SPDUIOut" as illustrated in 10.5.

Table 24 – SPDUIOut

Offset	Element name	Definition	Data type
0	ArgBlockID	0x1102	Unsigned16
2	SPDUOut0	Safety Protocol Data Unit out (octet 0)	Unsigned8
3	SPDUOut1	Safety Protocol Data Unit out (octet 1)	Unsigned8
...			
SPDUOutLength + 2	SPDUOuto	Safety Protocol Data Unit out (octet o)	Unsigned8

10.3.8 FSPDInOut

Table 25 shows the structure of the ArgBlock "FSPDInOut" as illustrated in 10.5.

Table 25 – FSPDInOut

Offset	Element name	Definition	Data type	Values
0	ArgBlockID	Unique ID	Unsigned16	0x1103
2	PQI	Port Qualifier Information	Unsigned8	–
3	OE	Output Enable	Unsigned8	–
4	InputDataLength ($m+n+2$)	This element contains the length of the Device's input Process Data contained in the following elements (SR + NSR).	Unsigned8	0 to 0x20
5	SPDUIn0	Safety Protocol Data Unit in (octet 0)	Unsigned8	0 to 0xFF
6	SPDUIn1	Safety Protocol Data Unit in (octet 1)	Unsigned8	0 to 0xFF
...				
($m+1$) + 4	SPDUIn m	Safety Protocol Data Unit in (octet m)	Unsigned8	0 to 0xFF
($m+1$) + 5	PDI0	Input Process Data (octet 0)	Unsigned8	0 to 0xFF
($m+1$) + 6	PDI1	Input Process Data (octet 1)	Unsigned8	0 to 0xFF
...				
($m+1$) + ($n+1$) + 4	PDIn	Input Process Data (octet n)	Unsigned8	0 to 0xFF
($m+1$) + ($n+1$) + 5	OutputDataLength ($o+p+2$)	This element contains the length of the Device's output Process Data contained in the following elements (SR + NSR).	Unsigned8	0 to 0x20
($m+1$) + ($n+1$) + 6	SPDUOut0	Safety Protocol Data Unit out (octet 0)	Unsigned8	0 to 0xFF
($m+1$) + ($n+1$) + 7	SPDUOut1	Safety Protocol Data Unit out (octet 1)	Unsigned8	0 to 0xFF
...				
($m+1$) + ($n+1$) + ($o+1$) + 5	SPDUOuto	Safety Protocol Data Unit out (octet o)	Unsigned8	0 to 0xFF
($m+1$) + ($n+1$) + ($o+1$) + 6	PDO0	Output Process Data (octet 0)	Unsigned8	0 to 0xFF
($m+1$) + ($n+1$) + ($o+1$) + 7	PDO1	Output Process Data (octet 1)	Unsigned8	0 to 0xFF
...				
($m+1$) + ($n+1$) + ($o+1$) + ($p+1$) + 5	PDO p	Output Process Data (octet p)	Unsigned8	0 to 0xFF

10.4 Safety Layer Manager (SLM)

10.4.1 Purpose

The Safety Layer Manager takes care of the safety PDU, whenever safety communication has been configured or of one safety bit, whenever OSSDe has been configured for a particular Port. It uses SMI services for this purpose as specified in 10.2.3 and 10.2.4.

It holds the FSP parameters consisting of the authenticity record and the protocol record (see 11.7.4) as well as of the FS I/O structure description (see Table A.1 and E.5.5) for the FS_IO_Data_Mapper.

After verification of all parameters, the safety communication layer (SCL) engine will be started and PortStatusInfo will be set to "8" (SCL_ENABLED) (see 10.3.5).

SLM or SCL respectively use the Diagnosis Unit (DU) in Figure 34 to propagate the PortEventCodes of Table B.2.

10.4.2 FS_PortModes

The FS-Master shall support the following PortModes of standard NSR Masters:

- DEACTIVATED
- IOL_MANUAL (basis of SCL operation)
- IOL_AUTOSTART (usually only in case of totally unknown connected Devices)
- DI_C/Q
- DO_C/Q (Devices and FS-Devices are short-circuit proof)

In addition, the FS-Master shall support two FS_PortModes and one optionally:

SAFETYCOM

This setting enables pure safety communication without NSR Process Data of a port.

MIXEDSAFETYCOM

This setting enables safety communication of SR and NSR Process Data of a port.

OSSDE

This setting enables OSSDe operation of a port. This mode is optional.

All these PortModes can be set up via the SMI_PortConfiguration (see 10.2.1) and the ArgBlock "FSPortConfigList" (see 10.3.4).

10.4.3 FSP parameter

10.4.3.1 FSP parameter use cases

Figure 36 illustrates some use cases related to the FSP parameters (see Clause A.1).

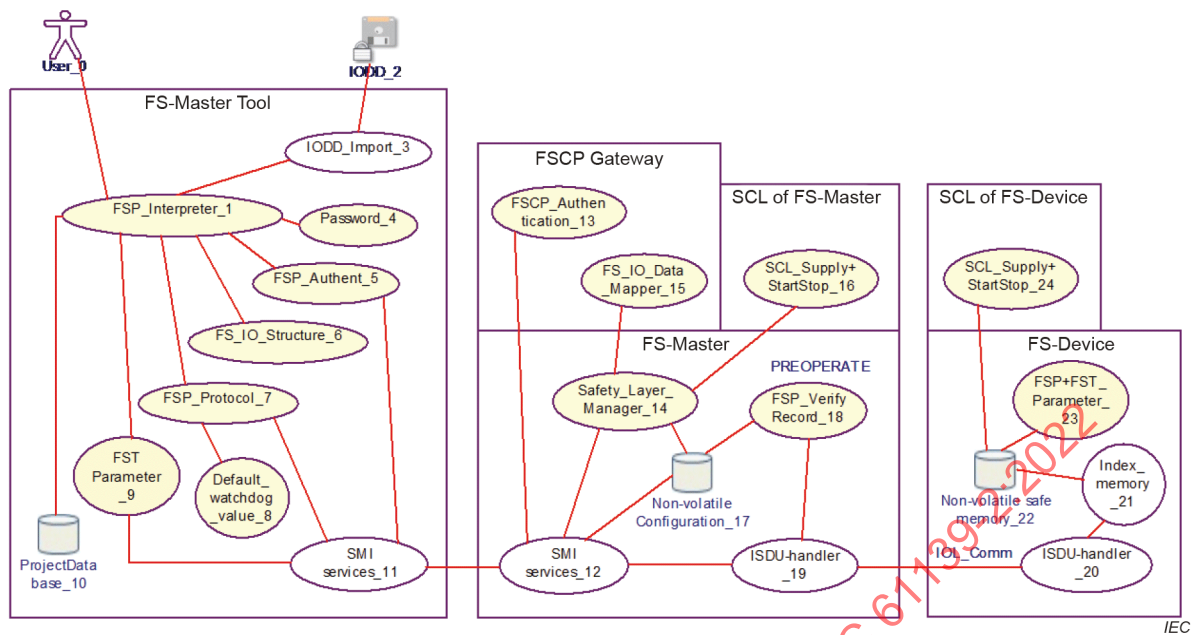


Figure 36 – FSP parameter use cases

Table 26 shows a listing of the items in Figure 36 and references clauses and subclauses within this document or IEC 61131-9.

Table 26 – Use case reference table

No.	Item	Type	Reference	Remarks
0	User	Roles: Observer Maintenance - Specialist	–	Responsibility of the software tool manufacturer
1	FSP_Interpreter	GUI-functions	E.g. Figure 58	–
2	IODD (secured)	Device description	Annex E	–
3	IODD_Import	Activity	Annex E	–
4	Password	Activity	10.4.3.2	Access protection
5	FSP_Authent	Activity	11.7.5	–
6	FS_IO_Structure	FS I/O description	Clause A.1	–
7	FSP_Protocol	Activity	11.7.6	–
8	Default_watchdog_value	Activity	A.2.6	–
9	FST Parameter	Activity	–	–
10	ProjectDatabase	FS-Master Tool	–	Proprietary
11	Standardized Master Interface (SMI)	Communication	10.4.3.1	–
12	Standardized Master Interface (SMI)	Communication	10.4.3.1	–
13	FSCP_Authentication	Activity	11.7.5	–
14	Safety_Layer_Manager	Activity	10.4	–
15	FS_IO_Data_Mapper	Gateway application	12.1	FSCP Integration
16	SCL_Supply+StartStop	FS-Master SCL	11.5.2	–
17	Non-volatile memory	FS-Master	–	Implementation
18	FSP_VerifyRecord	Verification	11.7.4	–
19	ISDU-Handler	FS-Master DL	IEC 61131-9	Standard SDCI

No.	Item	Type	Reference	Remarks
20	ISDU-Handler	FS-Device DL	IEC 61131-9	Standard SDCI
21	Index_memory	Activity	IEC 61131-9	Standard SDCI
22	Non-volatile memory	FS-Device	–	Implementation
23	FSP+FST parameter	Activities	–	–
24	SCL_Supply+StartStop	FS-Device SCL	11.5.3	–

In the following, a typical parameterization session of a project in the ProjectDatabase is described, where a new FS-Device is planned, configured, and parameterized for a particular Port. After installation of IODD and associated Dedicated Tool, the user of an FS-Master Tool opens the parameter tab page (see illustration in Figure 58). After entry of the password for safety projects (see 10.4.3.2), FSP parameters are enabled to be displayed and Dedicated Tools are enabled to be launched.

When online, the FS-Master Tool uses the Standardized Master Interface (SMI) to the FS-Master specified in IEC 61131-9. Any transmission error (see Table 27) can falsify the message bits and thus, each FSP parameter record is secured by CRC signature.

The choice of the SMI service call technology is the responsibility of the respective integration into a fieldbus (see IEC 61131-9).

The *authenticity parameter* values carry "0" as default within the IODD of an FS-Device. FS-Master Tool acquires FSCP Authenticity values with the help of the SMI_FSMasterAccess service (see 10.2.2) and suggests these as actual values. For details see 10.4.3.3.

The IODD contains the *I/O data structure description* of the safety Process Data as a record secured by CRC signature (see A.2.7 and E.5.6). This information is used for the mapping to FSCP I/O data and checked by FS-Device (see 11.7.7).

Most of the *protocol parameter* values are preset by default values provided by the FS-Device manufacturer within the IODD, except for the value of FSP_TechParCRC, which has a particular responsibility. A value of "0" followed by a Port power OFF/ON means commissioning/test (see Annex G). The consequences are:

- only verification of FSP_TechParCRC = 0 is required to start SCL;
- changes of FST parameters become effective right upon acceptance by the FS-Device;
- no Data Storage backup.

From now on the SDCI-FS system can run in "monitored operational mode". That means personnel are required to watch the machine.

The user can now enter and test the technology specific parameters (see illustration in Figure 58). After verification and validation, the user launches the Dedicated Tool, confirms the value assignments, and transfers the CRC signature to the FSP_TechParCRC field. The corresponding SMI_PortConfiguration cares for the FSP_VerifyRecord within the FS-Master. With an FSP_TechParCRC value of $\neq$ "0", the system can be armed:

- Data Storage;
- Verification of authenticity, protocol, and technology parameters, as well as IO data description at start-up.

After parameter assignment, the FSP and FST parameter instance values can be stored in the ProjectDatabase.

Another Port power OFF/ON will cause the FS-Device to perform safety selftests prior to communication and the FS-Master to transmit the FSP_VerifyRecord to the FS-Device. Its Safety Layer Manager verifies all parameters, passes relevant protocol parameters, and launches the SCL. In case of mismatch, a corresponding Event is activated and the SCL will not operate.

The SLM propagates the I/O structure description to the FS_IO_DataMapper. The FSP_VerifyRecord is propagated to the local FS-Master safety communication layer (SCL). It verifies all parameters, passes relevant protocol parameters, launches the SCL, and sets PortStatusInfo to "8" (SCL_ENABLED). In case of mismatch a corresponding Event is activated and the SCL will not operate.

10.4.3.2 Protection

An FS-Master Tool creates and maintains safety projects and the FS-Master stores safety parameters within configuration data. Both require protection from easy manipulation. FS-Master Tool can use password parameters within the SMI_FSMasterAccess service (10.2.2) for that purpose. Usage depends on the security concept of the upper-level system and on the inheritance concept of password protection. It is optional.

Dedicated Tools may have password mechanisms for their FS-Devices independent from the FS-Master (see 10.2.2).

10.4.3.3 FSP authenticity parameter record

FSP authenticity parameters are specified in Clause A.1. The FSP authenticity record includes the FSCP authenticity code, a Port number, and a CRC signature. An FS-Master Tool shall always update the CRC signature when changes occur and only write entire consistent records.

For stand-alone FS-Masters the entry of unique and unambiguous values per FS-Master is required per machine or production center if there is a possibility to misconnect FS-Devices amongst different FS-Masters.

10.4.3.4 FSP protocol parameter record

FSP protocol parameters are specified in Clause A.1. Manufacturer/vendor presets values and defines ranges within the IODD for protocol version and mode, port mode, watchdog, and TechParCRC.

Manufacturer/vendor shall determine the preset value for the watchdog timer considering the FS-Device response time at the indicated transmission rate (COMx). The FS-Master Tool can calculate and suggest a value based on the performance data of the used FS-Master and on the preset value from the IODD.

An FS-Master Tool shall always update the CRC signature when changes occur and only write entire consistent records.

10.4.3.5 FS I/O data structure description

With the help of this information, the mapping process within the FSCP gateway can be controlled or monitored (see 11.7.7 and A.2.7).

The FS-Device shall check the validity of its implemented safety PDin/PDout structure via the FSP_IO_StructCRC signature provided by the FSP_VerifyRecord.

10.4.3.6 Verification record

The FS-Master takes the FSP_VerifyRecord from the SMI_PortConfiguration service (see 10.3.3 and 11.7.4).

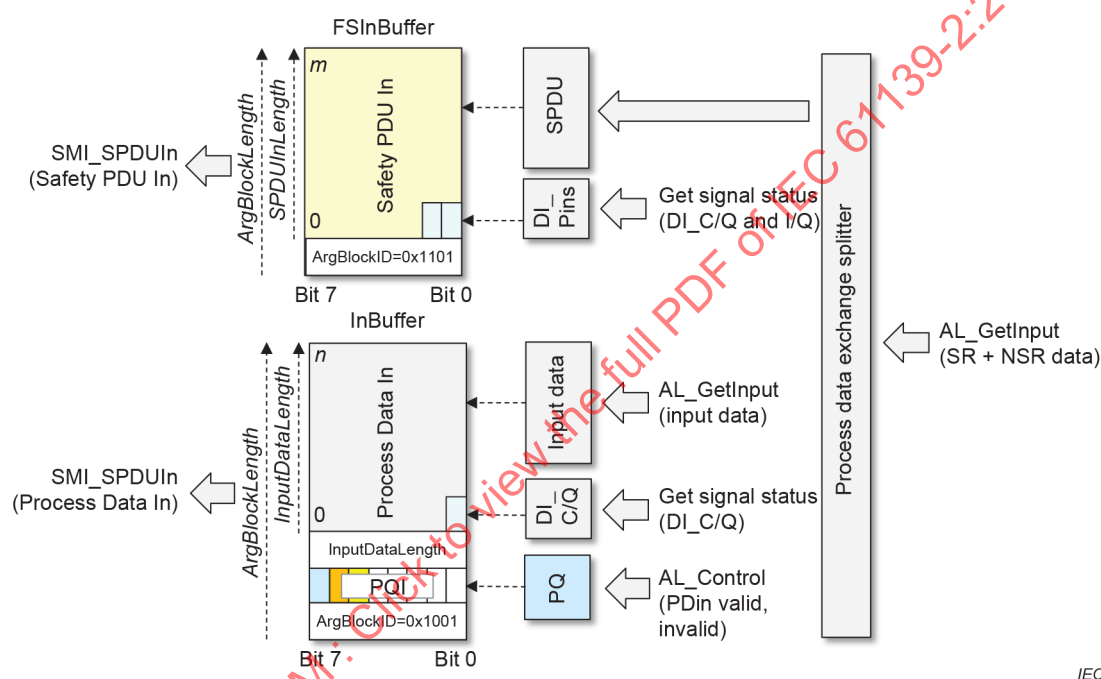
10.5 Process Data Exchange (PDE)

The FS-Master application Process Data Exchange (PDE) provides additional features called "Splitter" and "Composer".

Figure 37 shows the mechanism of splitting the SPDUI part and the Input Data from the complete SR and NSR data. The SR part is stored within an FSInBuffer and the NSR part within the InBuffer already specified in IEC 61131-9.

In case of PortConfiguration "OSSDE", the signal status of C/Q is stored as "OSSDe1" in Bit "0" of octet "0", and signal status of I/Q is stored as "OSSDe2" in Bit "1" of octet "0".

See IEC 61131-9 for definitions of "PQI" and "PQ". PDIn valid/invalid only refer to non-safety data.



IEC

Figure 37 – PDE Splitter

Figure 38 shows the mechanism of composing the complete SR and NSR data. Both are prepared for the AL_SetOutput service out of the SPDU Out part from the FSOutBuffer and out of the Process Data Out from the OutBuffer already specified in IEC 61131-9.

See IEC 61131-9 for definitions of "OE" and "OE detect".

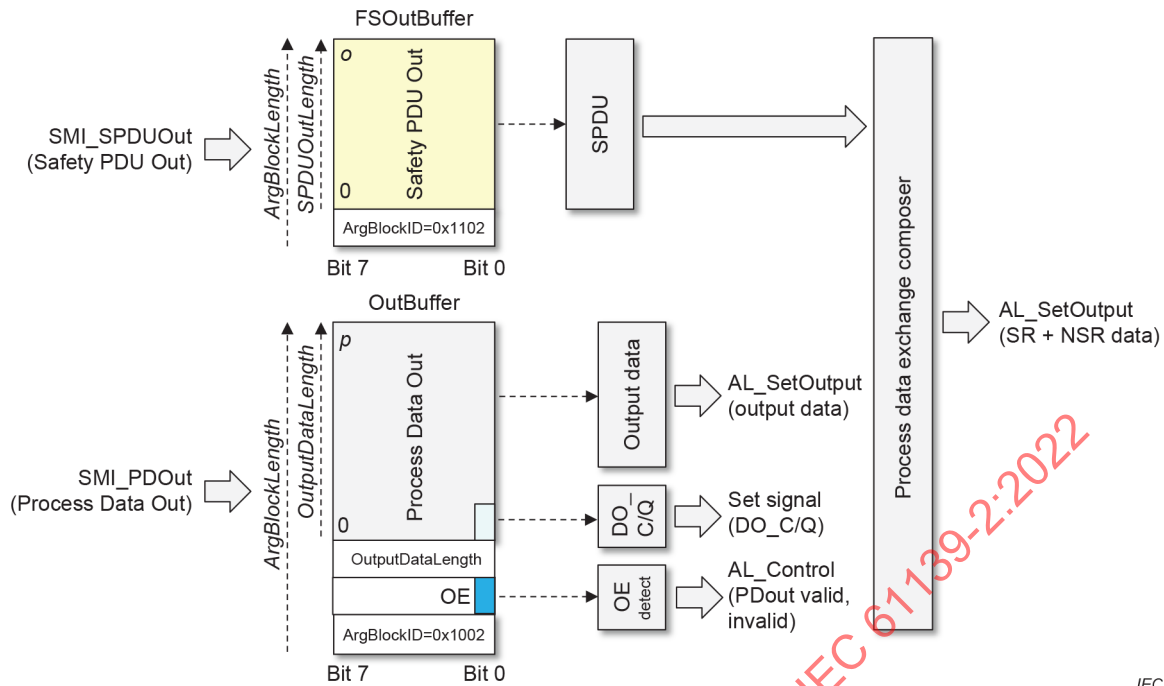


Figure 38 – PDE Composer

10.6 Data Storage (DS)

Data Storage for the purpose of parameter backup and Device/FS-Device replacement is specified in IEC 61131-9. A brief description and adaptations for SDCI-FS can be found in 9.4.

11 Safety communication layer (SCL)

11.1 Functional requirements

The functional requirements for SDCI-FS are laid down in [10]. Main application area is "safety for machinery". Usually this means operational stop of a machine until clearance or repair and restart only after an operator acknowledgement. Primarily relevant are IEC 62061 and ISO 13849.

Other major requirements are suitability for up to SIL3/PL_e safety functions, Port specific passivation, and parameterization using Dedicated Tools. Safety measures and residual error rates for timeliness, authenticity, and data integrity ("TADI") of safety messages (safety PDUs) shall be compliant with IEC 61784-3:2021.

11.2 Communication errors and safety measures

The point-to-point communication basis of SDCI allows for a very lean protocol type and a hardware independent safety communication layer stack with a small memory footprint. Table 27 shows the communication errors to be considered and the chosen safety measures:

- (sequence) counter / inverted counter;
- watchdog timer and receipt messages;
- connection validation at commissioning, start-up, and repair, and
- cyclic redundancy check for data integrity.

Table 27 – Communication errors and safety measures

Communication error	Protocol safety measures			
	Counter/Inverted counter	Timeout with receipt	PortNum + Connection validation ^a	Cyclic redundancy check (CRC)
Corruption	–	–	–	X
Unintended repetition	X	X	–	–
Incorrect sequence	X	–	–	–
Loss	X	X	–	–
Unacceptable delay	–	X	–	–
Insertion	X	–	–	–
Masquerade	–	–	X	X
Addressing	–	–	X	–
Loop-back of messages	X	–	X	–

^a Connection validation comprises an FSCP authenticity (see A.2.1) and the FS-Master Port number.

It is assumed, that there are no storing elements within the SDCI communication path between FS-Master and FS-Device. Thus, a three-bit counter is sufficient as a safety measure. A value 0b000 of this counter on FS-Master side indicates a start or reset position of this counter. In cyclic mode it counts to 0b111 and returns to 0b001.

The message "send" and "receive" concept of SDCI allows for a simple watchdog timer and message receipt safety measure concept corresponding to the "de-energize to trip" principle.

It is assumed that an FS-Master is the owner of a functional safety connection ID of the upper-level FSCP communication system (FSCP authenticity) similar to an FS-DI-Module within a remote I/O. A customer is required to perform a validation procedure, whenever a change occurred with the connected safety devices. SDCI-FS relies on such a concept. Additionally, due to the standard "data storage" mechanism of SDCI and the functional safety nature of the FS-Master, it is possible to provide a more convenient mechanism.

A CRC signature is used for the data integrity check of transmitted safety PDUs. Two options can be configured. A 16-bit CRC signature for safety I/O data up to 3 octets or a 32-bit CRC signature for safety IQ data up to 25 octets can be chosen.

11.3 SCL services

11.3.1 Positioning of safety communication layers (SCL)

Figure 39 shows the positioning of the SDCI-FS Safety Communication Layer (SCL).

For each port with a connected FS-Device an instance of the SCL is required. The SCLs are exchanging safety PDUs consisting of output Process Data (PDout) together with safety code to the FS-Device and input Process Data (PDin) together with safety code from the FS-Device. The SCLs are using standard SDCI communication as a "black channel".

Sufficient availability through for example correct installations, low-noise power supplies, and low interferences are preconditions for this "black channel" to avoid so-called nuisance/spurious trips. These trips cause production stops and subsequently may cause management to remove safety equipment.

This document does not specify implementation related safety measures such as redundant microcontrollers, RAM testing, etc. It is the responsibility of the manufacturer/vendor to take appropriate measures against component failures or errors according to IEC 61508 (all parts).

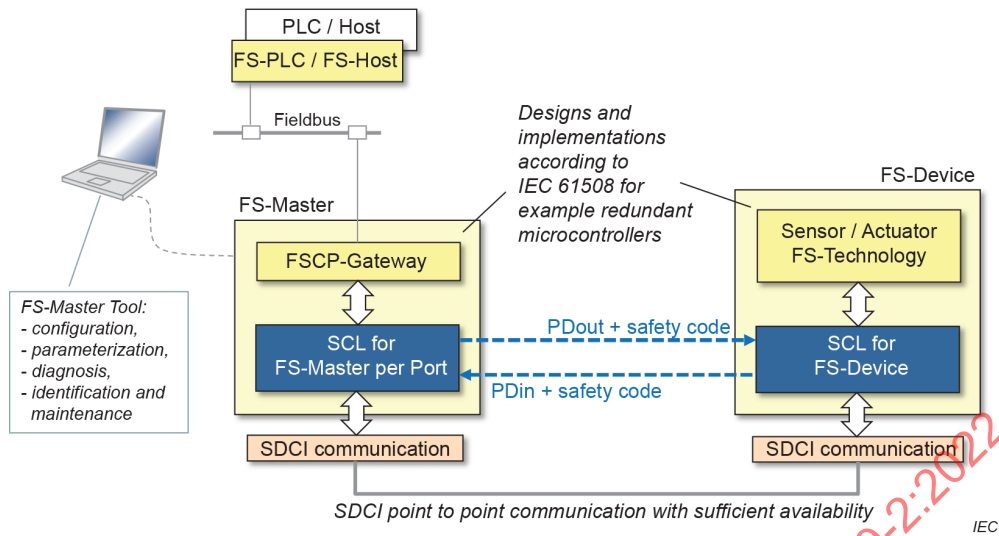


Figure 39 – Positioning of the SDCI-FS Safety Communication Layer (SCL)

11.3.2 FS-Master SCL services

SDCI-FS applications include (but are not limited to) connections to upper-level FSCP fieldbus systems. FSCPs usually also provide safety codes and control/monitoring services (signals).

Figure 40 shows the FS-Master Safety Communication Layer signals (services) depicted by arrows in the upper part of the figure. For each FSCP to be connected to, a mapping or emulation of corresponding SCL services is required.

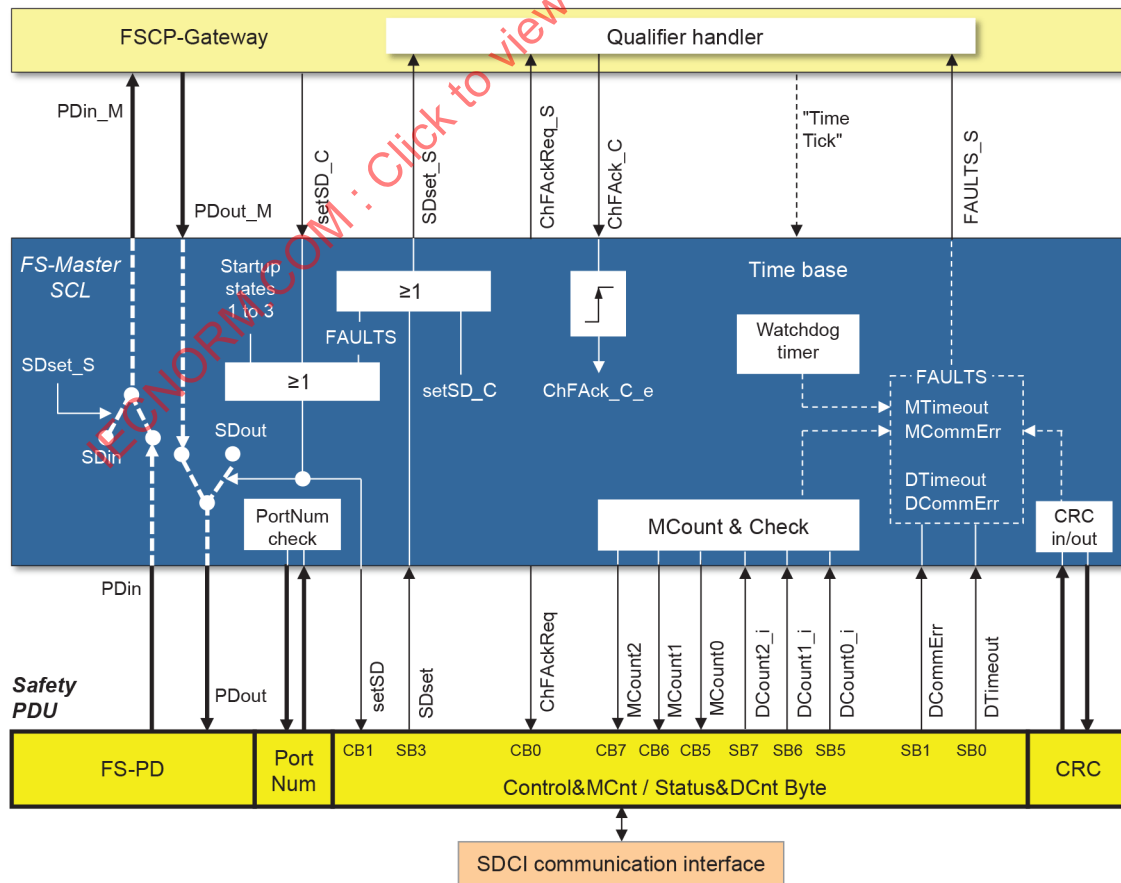


Figure 40 – FS-Master Safety Communication Layer services

A service name carries either an extension "_C" (Control), if it controls the safety communication activities or an extension "_S" (Status), if it is reporting on the activities.

Some of the service names correspond to the signal names of the Control Byte or Status Byte (see lower part of the figure and 11.4.5). That means they are correlated, but there is some control logic of the SCL in between. This control logic is time discrete and not continuous even if it is depicted as logic OR ("≥") box. Definitive are the state charts and the state transition tables of the SCL (see 11.5.2).

The following services in Table 28 shall be available to the FSCP gateway or to a programmer of an FS-Master system.

Table 28 – SCL services of FS-Master

Service/signal	Definition
PDin_M, PDout_M	These services carry the actual Process Data values, both SDin (all bits "0") and SDout (all bits "0") in case of safe state or the real process values from or to the FS-Device.
SDin, SDout	These services carry Process Data values all zero.
setSD_C	In case of emergency, safety control programs usually set output Process Data (PDout_M) for an actuator to "0". However, in some cases, for example burner ventilators, shut down may not be a safe state. This service, if set to "1", is additional information allowing an FS-Device to establish a safe state no matter what the values of Process Data are. Independent from PDout_M, this service causes the SCL to send SDout values to the FS-Device and to send SDin to the FSCP gateway (PDin_M) via SDset_S.
SDset_S	This service, if set to "1", causes the qualifier handler to set the qualifier bit for the Process Data of the connected FS-Device (see 11.11.4). In addition, it causes the SCL to send SDin to the FSCP gateway (PDin_M).
ChFAckReq_S	The FS-Master SCL sets this service to "1" in case of disappeared FAULTS or FS-Master timeouts. It shall be propagated via FSCP and indicated to the operator.
ChFAck_C	After check-up and/or repair, the operator is requested to acknowledge a "ChFAckReq_S" service via a "1". This is a precondition for the SCL to resume regular operation after 3 transmission cycles with SDin and SDout values. The operator shall release the pressed acknowledgment button to enable further acknowledgments. See "Internal Items" in Table 38.
FAULTS_S	Any communication error (counter mismatch or CRC signature error) and/or timeouts cause the qualifier handler to set the qualifier bit for the Process Data of the connected FS-Device (see 11.11.4).

The lower part of the figure shows a combined input and output safety PDU specified in 11.4.3 and 11.4.5.

11.3.3 FS-Device SCL services

Figure 41 shows the FS-Device Safety Communication Layer services depicted by arrows in the upper part of the figure.

A service name carries either an extension "_DC" (Device Control) if it controls the FS-Device technology or an extension "_DS" (Device Status) if it is reporting its status.

Some of the service names correspond to the signal names of the Control Byte or Status Byte (see lower part of the figure and 11.4.5). That means they are correlated, but there is some control logic of the SCL in between. This control logic is time discrete and not continuous even if it is depicted as logic OR ("≥") box. Definitive are the state charts and the state transition tables of the SCL (see 11.5.3).

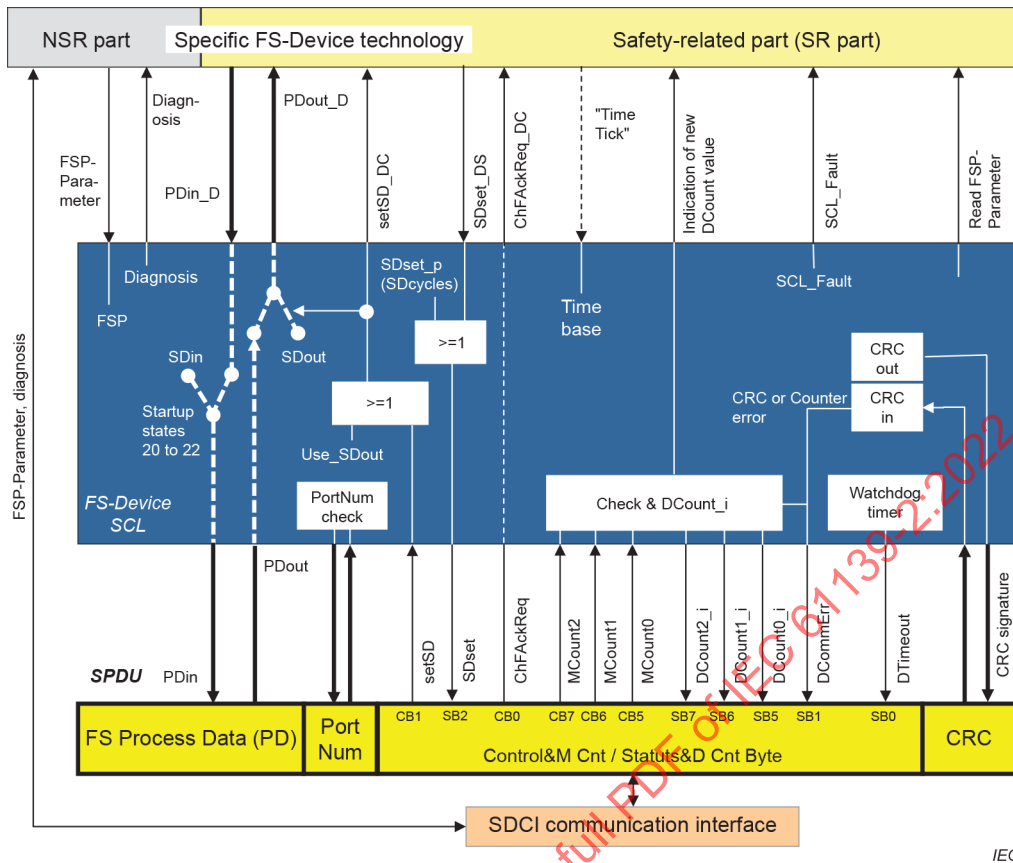


Figure 41 – FS-Device Safety Communication Layer services

The following services in Table 29 shall be available to the safety-related part of the FS-Device technology. Some services are non-safety-related and shall be available to the non-safety-related part of the FS-Device.

Table 29 – SCL services of FS-Device

Service/signal	Definition
PDin_D, PDout_D	These services carry the actual Process Data values. Real process values from the FS-Device and SDout (all bits "0") in case of safe state or the real process values to the FS-Device.
SDin, SDout	These services carry Process Data values all zero. Signal Use_SD indicates the usage of Process Data all zero.
setSD_DC	In case of emergency, safety control programs usually set output Process Data (PDout) for an actuator to "0". However, in some cases, for example burner ventilators, shut down may not be a safe state. This service, if set to "1", is additional information allowing an FS-Device to establish a safe state no matter what the values of Process Data are. Independent from PDout, this service causes the SCL to send SDout values to the FS-Device.
SDset_DS	This service, if set to "1", indicates that the FS-Device either reacts on a setSD_DC = "1" when the safe state is established or has been forced to establish safe state due to error or failure and delivers input Process Data values "0" (PDin_D).
ChAckReq_DC	This service, if set to "1", indicates a pending operator acknowledgment. This signal is not safety-related and can be used to control an indicator, for example LED (light emitting diode).
Time tick	The SCL can be designed totally hardware independent if a periodic service call controls a time base inside the SCL.

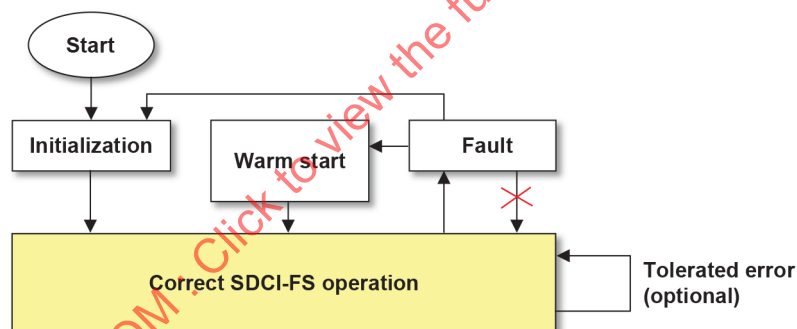
Service/signal	Definition
Indication of new DCount value	Short demands of FS-Devices may not trip a safety function due to its chain of independent communication cycles across the network. Therefore, a demand shall last for at least two SCL cycles. This service provides the necessary information to implement the demand extension if required.
SCL_Fault	This service provides faults (errors) of the SCL software.
Read_FSP_Parameter	This service allows the FS-Device technology for reading the current FSP (protocol) parameter
Non-safety-related services:	
FSP_Parameter	The FS-Master transmits the FSP parameter record (block) at each start-up during PREOPERATE to the FS-Device. These parameters are propagated to the SCL using this service.
Diagnosis	SCL diagnosis information can be propagated to the SDCI Event system using this service.

The lower part of Figure 41 shows a combined input and output safety PDU specified in 11.4.3 and 11.4.5.

11.4 SCL protocol

11.4.1 Protocol phases to consider

Figure 42 shows the principle protocol phases to consider for the design according IEC 61784-3:2021.



IEC

Figure 42 – Protocol phases to consider

The principle protocol phases and the corresponding requirements are listed in Table 30.

Table 30 – Protocol phases to consider

Phase	Activities	Requirements
Initialization	Establish communication, transfer FSP parameter to FS-Device, SD cycles	– Actuator shall be de-energized – SDout values shall be used during the first 3 SCL communication cycles
Setup or change	Commissioning, FST parameter backup	– As long as the FSP_TechParCRC is set to "0", cyclic data exchange of PD values is enabled.
Operation	Process Data exchange, power-down of FS-Device	– It is the responsibility of the FS-Device technology to detect undervoltages and to set SD values.
Restart after transition from fault	Timeout, operator acknowledgment	– Operator acknowledgment is required prior to a restart – MCounter reset (resynchronization) – SDout values shall be used during the first 3 SCL communication cycles
Warm start after transition from fault	CRC or counter error, operator acknowledgment	– Operator acknowledgment is required prior to a restart – SCL communication is not reset – SDout values shall be used during the first 3 SCL communication cycles
Shutdown	Contact bouncing, EMC voltage dips/changes	– It is the responsibility of the FS-Device technology to detect undervoltages and to set SD values.

11.4.2 FS-Device faults

The SCL protocol copes with faults occurring during transmission of safety PDUs such as CRC errors or timeouts. It is the responsibility of the designer of the FS-Device to cope with FS-Device faults and to make sure that the necessary functional safety actions will take place, for example setting of safety Process Data and the SDset_DS service.

11.4.3 Safety PDU (SPDU)

Figure 43 shows the structure of SPDUs of the FS-Master and FS-Device together with standard input and output data. The design follows the concept of explicit transmission of the safety measures for timeliness and authenticity according to IEC 61784-3:2021 in contrast to the implicit transmission via inclusion in the overall CRC signature calculation.

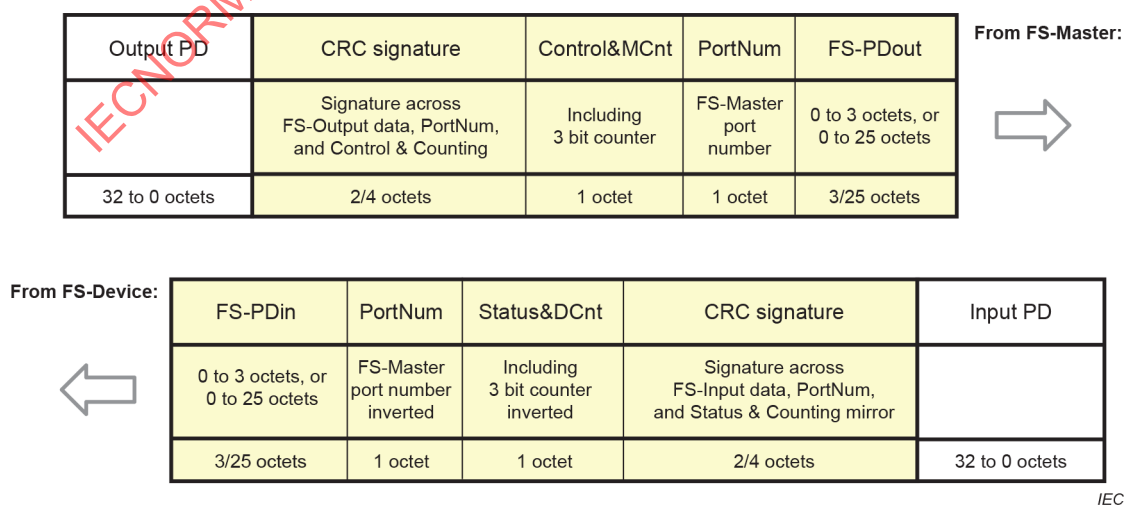


Figure 43 – Safety PDUs of FS-Master and FS-Device

The timeliness measure is represented by a 3-bit counter within the protocol management octets (see 11.4.6).

With respect to authenticity, only the FS-Master Port number is included in cyclic checking due to requested usage of unchanged SDCI implementations ("Black channel"). However, complete authenticity checking is performed during commissioning and at start-up.

The design is an enhancement of the original "de-energize to trip" principle. In case of a timeout, or a CRC error, or a counter error, or a PortNum error, the associated qualifier bit will be set. It will be only released after an explicit operator acknowledgment on the FS-Master side. After a CRC error, a warm start is possible.

11.4.4 FS-Input and FS-Output data

The maximum possible size of the FS-Input and FS-Output data reaches from 0 to 25 octets depending on the amount of required standard SDCI data. See 11.4.7 for optimization issues and trade-offs. The possible data types are listed in Table 34.

NOTE Currently the safety code consists of only 4 or 6 octets and theoretically 26 octets could be available.

11.4.5 Port number

One octet carries the FS-Master port number or value of FSP_Port respectively (see A.2.2). FS-Device returns the inverted value of the port number.

11.4.6 Status and control

One octet is used in both transmission directions for the protocol flow of SDCI-FS. Table 31 shows the signals to control the protocol layer of an FS-Device and a counter value for the timeliness check together with a local watchdog timer adjusted through the "FSP_Watchdog" parameter (see A.2.6).

Table 31 – Control and counting (Control&MCnt)

CB7	CB6	CB5	CB4	CB3	CB2	CB1	CB0
Sequence counter, bit 2	Sequence counter, bit 1	Sequence counter, bit 0	Reserved ("0")	Reserved ("0")	Reserved ("0")	Activate safe state	Channel fault acknowledge request (indication)
MCount2	MCount1	MCount0	–	–	–	SetSD	ChFAckReq

Table 32 shows the feedback of the protocol layer of an FS-Device and the inverted counter value for the timeliness check. The counter values are inverted to prevent from undetected loop-back errors.

Table 32 – Status and counting mirror (Status&DCnt)

SB7	SB6	SB5	SB4	SB3	SB2	SB1	SB0
Sequence counter, bit 2; inverted	Sequence counter, bit 1; inverted	Sequence counter, bit 0; inverted	Reserved ("0")	Reserved ("0")	Safe state activated	Communication error: CRC or counter /port incorrect	Communication fault: Timeout
DCount2_i	DCount1_i	DCount0_i	–	–	SDset	DCommErr	DTimeout

Table 33 shows the values of MCount and DCount_i during protocol operation.

Table 33 – MCount and DCount_i values

Phase	MCount		DCount_i	
	Dec	Bin	Dec	Bin
Initial or after timeout	0	000	7	111
Cyclic	1	001	6	110
	2	010	5	101
	3	011	4	100
	4	100	3	011
	5	101	2	010
	6	110	1	001
	7	111	0	000

11.4.7 CRC signature

For the design of the CRC mechanism and the calculation of the residual error probability/rate several parameters and assumptions are required:

- No multi-drop, multi-channel, or encrypted transmission in SDCI. Thus, the Binary Symmetric Channel (BSC) model can be applied.
- Explicit transmission of safety measures as opposed to implicit transmission. In this case, formulas are available within IEC 61784-3:2021.
- The sampling rate of safety PDUs is assumed to be a maximum of 1 000 sampled safety PDUs per second.
- The monitoring times for errors in safety PDUs are listed in Table 41. Any detected CRC error within the safety communication layer shall trip the corresponding safety function (safe state). During the monitoring time only one nuisance trip is permitted. Maintenance is required.
- The generator polynomials in use shall be proven to be proper within the SPDU range.
- The seed value to be used for the CRC signature calculation is "1" (see D.3.6).
- In case the result of the CRC signature calculation leads to a "0", a "1" shall be sent and evaluated at the receiver side correspondingly.
- The assumed bit error probability for calculations is 10^{-2} .

Figure 44 shows the so-called 1 % share rule of the IEC 61784-3:2021. For SDCI-FS it means, the residual error rate of an SDCI-FS logical connection shall not exceed 1 % of the average frequency of a dangerous failure (PFH) of that safety function with the highest SIL the safety communication is designed for, which is SIL3. This value is $10^{-9}/h$.

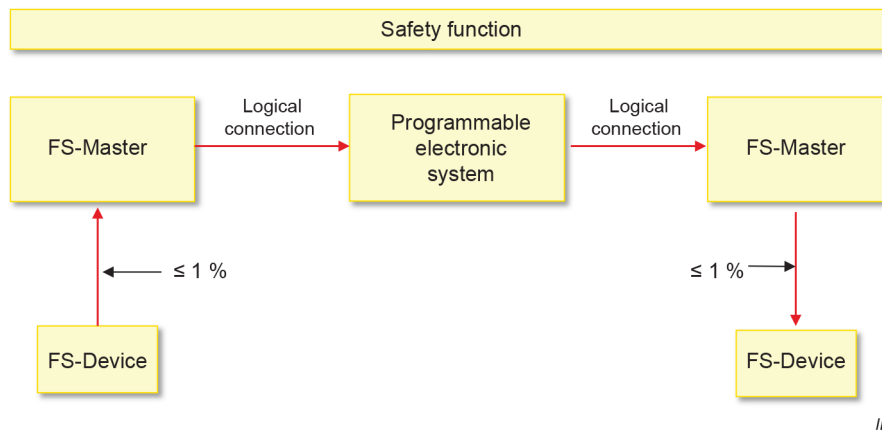


Figure 44 – The 1 % share rule of IEC 61784-3:2021

Calculations under the above conditions have shown the following possibilities (see Annex D):

- For a CRC-16 proper polynomial ($0x4EAB$) 3 octets of Process Data (safety PDU length = 7 octets),
- For a CRC-32 proper polynomial ($0xF4ACFB13$) 25 octets of Process Data (safety PDU length = 31 octets).

Thus, support of two variants is provided: CRC-16 with up to 3 octets of safety I/O data and CRC-32 with up to 25 octets of safety I/O data.

11.4.8 TADI safety considerations (informative)

In order for the SCL protocol to be compliant with IEC 61784-3:2021 (see 11.1), considerations and calculations shall be performed to prove that the overall PFH for

- timeliness,
- authenticity (masquerade, loopback), and
- data integrity,

is less than $10^{-9}/h$ (Figure 44).

The measures for timeliness are specified in 11.4.6 and consist of the 3-bit counter in combination with a watchdog timer within the SCL. This provides an added measure of data integrity and SCL state checking. However, in accordance with IEC 61784-3:2021, 5.8.8, due to the point-to-point nature of the SDCI-FS and no storage elements are allowed, the residual error rate for timeliness, RR_T is 0.

The measure for authenticity and masquerade checking is provided by the explicit transmission of a Port number in the SPDU that is checked by the receiving SCL endpoint. This provides an added measure of data integrity checking. However, in accordance with IEC 61784-3:2021, 5.8.7, due to the point-to-point nature of the SDCI-FS, the rate of occurrence for misdirected SPDUs (R_A) is 0, and therefore, the residual error rate for authenticity, RR_A is 0. Similarly, the rate of occurrence for masqueraded SPDUs (R_M) is 0, and therefore, the residual error rate for masquerade, RR_M is 0.

The measures for data integrity are provided by the CRC signature across safety process data, Port number, and Control&MCnt or Status&DCnt respectively as shown in 11.4.3. The calculation of the residual error probability can be performed at a bit error probability of 10^{-2} using the information in Annex D of this document. Designers shall observe the maximum sample rate specified in 11.4.7. The PFH monitor limits the maximum number of detected corrupted SPDUs for a given time interval (see 11.5.5).

11.4.9 Data types for SDCI-FS

11.4.9.1 General

The cyclically exchanged functional safety data structures between FS-Device and FS-Master comprise FS process I/O data and the SDCI-FS protocol trailer. They are transmitted in Safety PDUs.

Acyclically exchanged functional safety data structures are transmitted in SDCI On-request Data (OD) containers either from a Dedicated Tool or from a user program within an FS-PLC. In this case additional securing mechanisms (e.g. CRC signature) are required at each and every transfer or after a parameter block.

11.4.9.2 FS process I/O data (PDin and PDout)

For the FS process I/O data a well-defined set of data types and a corresponding description is defined for both FS-Device and FS-Master for correct processing and mapping to the upper-level FSCPs. Table 34 lists the three permitted data types (see Annex C).

Table 34 – FS process I/O data types

Data type	Coding	Length	See IEC 61131-9:—	Device example
BooleanT/bit	BooleanT ("packed form" for efficiency, no WORD structures); assignment of signal names to bits is possible.	1 bit	F.2.2, Table F.22 and Figure F.9	Proximity switch
IntegerT(16)	IntegerT (enumerated or signed)	2 octets	F.2.4, Table F.4, Table F.7 and Figure F.3	Protection fields of laser scanner
IntegerT(32)	IntegerT (enumerated or signed)	4 octets	F.2.4, Table F.4, Table F.6, and Figure F.3	Encoder or length measurement ($\approx \pm 2$ km, resolution 1 μ m)

11.4.9.3 Qualifier

FS-Devices normally do not require qualifiers (see 11.11.2). The qualifier bits are configured together with the Process Data (or Safe Data = SD) during the mapping to the upper-level FSCP system. The data structures depend on the rules of these FSCP systems.

In case of FS-Terminals (see 11.11.3) the rules in Table 35 for the layout of binary and digital data and their qualifier bits apply.

Table 35 – Rules for the layout of values and qualifiers

No.	Rule
1	Only Boolean (DI, DO) and IntegerT(16) or IntegerT(32) (AI, AO) data types shall be used. Any value shall be assigned to one of these categories.
2	Boolean values precede Integer values
3	IntegerT(16) precedes IntegerT(32) values
4	Values precede qualifier in an octet-wise manner
5	Qualifiers follow directly input values. In case of no input values only the qualifiers for output values are placed.
6	Qualifier for input values precede qualifier for output values
7	Qualifiers for each category (DI, DO, AI, AO) are packed separately in an octet-wise manner
8	If data types are missing the remaining data types catch up

Table 36 shows the ranking of values and qualifiers.

Table 36 – Order of values and qualifier

Order	To FS-Master	To FS-Device
1	Value DI	Value DO
2	Value AI	Value AO
3	Qualifier DI	–
4	Qualifier AI	–
5	Qualifier DO	–
6	Qualifier AO	–

11.4.9.4 SDCI-FS protocol trailer

The data types for the protocol trailer ("safety code") are specified in Clause C.5.

11.4.9.5 FSP and FST parameter

No particular data type definitions are required.

11.5 SCL behavior

11.5.1 General

The state machines for the FS-Master and the FS-Device safety communication layer are designed using the chosen safety measures in Table 27 and the protocol signals in 11.4.6.

11.5.2 SCL state machine of the FS-Master

Figure 45 shows the FS-Master state machine for wired SDCI point-to-point communication.

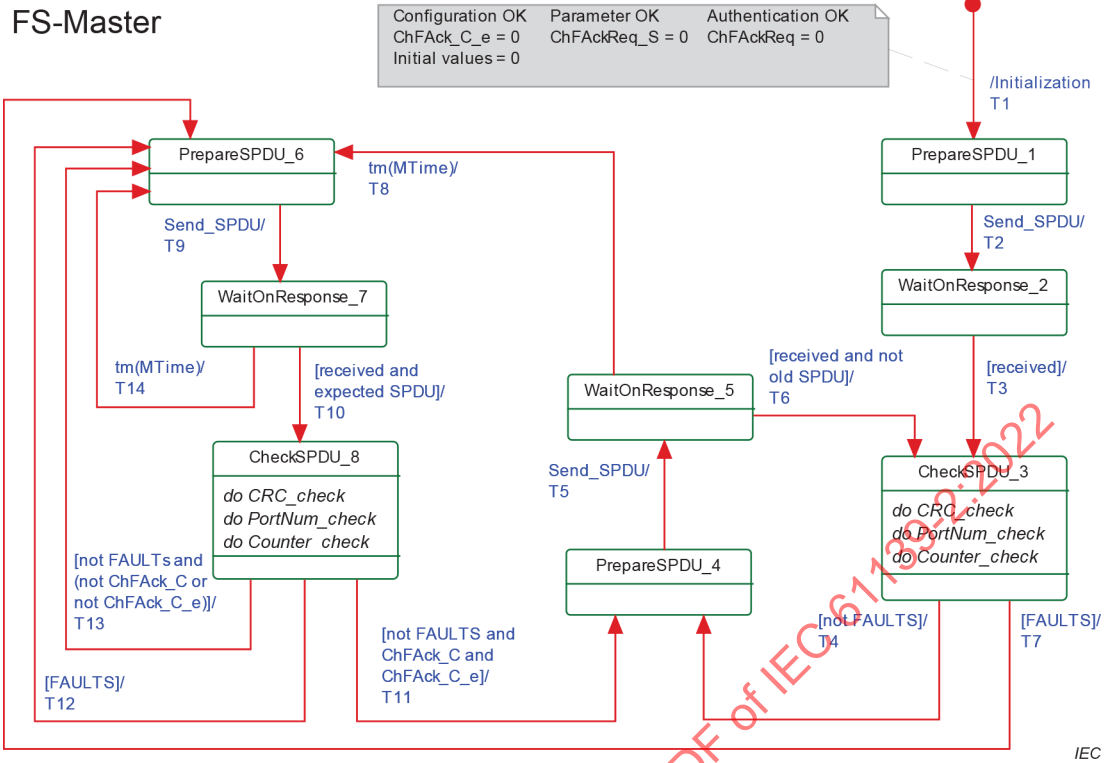


Figure 45 – SCL state machine of the FS-Master

The terms used in Figure 45 are defined in Table 37.

Table 37 – Definition of terms used in SCL state machine of the FS-Master

Term	Definition
ChFack_C	Operator acknowledgment for the safety function via the FS-Gateway
FAULTS	MTimeout: FS-Master timeout when waiting on an FS-Device SPDU response, or MCommErr: FS-Master detects a corrupted FS-Device SPDU response (incl. counter/port error), or DTimeout: FS-Device reported a timeout of its SCL via Status&DCnt Byte, or DCommErr: FS-Device reported a CRC (incl. counter/port error) by its SCL via Status&DCnt Byte

Table 38 – FS-Master SCL states and transitions

STATE NAME		STATE DESCRIPTION	
Initialization		Initial state of the FS-Master SCL instance upon power-on (one per port).	
1 PrepareSPDU		Preparation of a (<i>regular</i>) SPDU for the FS-Device. Send SPDU when prepared.	
2 WaitOnResponse		SCL is waiting on SPDU from FS-Device. SPDU with all octets "0" shall be ignored.	
3 CheckSPDU		Check received SPDU for not FAULTS (→ T4). In case of FAULTS: errors within the Status&DCnt Byte (DCommErr, DTimeout, SDset) → T7	
4 PrepareSPDU		Preparation of a (<i>regular</i>) SPDU for the FS-Device. Send SPDU when prepared.	
5 WaitOnResponse		SCL is waiting on next SPDU from FS-Device not carrying the previous DCount_i. SPDU with all octets "0" shall be ignored.	
6 PrepareSPDU		Preparation of an (<i>exceptional</i>) SPDU for the FS-Device (due to MTimeout, missing OpAck, or FAULTS).	
7 WaitOnResponse		SCL is waiting on next SPDU from FS-Device not carrying the previous DCount_i. When received → T10, after MTimeout → T14.	
8 CheckSPDU		Check received SPDU for a CRC error (MCommErr) and for potential FS-Device faults within the Status&DCnt Byte (DTimeout, DCommErr). Once a fault occurred, no automatic restart of a safety function is permitted unless an operator acknowledgement signal (ChFAck_C) arrived (see Figure 40). Hint: A delay time may be required avoiding the impact of an occasional system shutdown.	
TRAN-SITION	SOURCE STATE	TARGET STATE	ACTION
T1	0	1	use SD, setSD = 1, SDset_S = 1 MCount = 0
T2	1	2	–
T3	2	3	–
T4	3	4	MCount = MCount + 1 if MCount = 8 then MCount = 1 if SDset = 1 or setSD_C = 1 then use SDin, SDset_S = 1 else use PDin, SDset_S = 0 if setSD_C = 1 then use SDout, setSD = 1 else use PDout, setSD = 0
T5	4	5	restart MTimer
T6	5	3	–
T7	3	6	use SD, setSD = 1, SDset_S = 1 MCount = MCount + 1 if MCount = 8 then MCount = 1
T8	5	6	use SD, setSD = 1, SDset_S = 1 MCount = 0
T9	6	7	restart MTimer
T10	7	8	–
T11	8	4	ChFAckReq = 0, ChFAckReq_S = 0, ChFAck_C_e = 0, MCount = MCount + 1 if MCount = 8 then MCount = 1 if SDset = 1 or setSD_C = 1 then use SDin, SDset_S = 1 else use PDin, SDset_S = 0 if setSD_C = 1 then use SDout, setSD = 1 else use PDout, setSD = 0
T12	8	6	ChFAckReq = 0, ChFAckReq_S = 0, ChFAck_C_e = 0, use SD, setSD = 1, SDset_S = 1 MCount = MCount + 1 if MCount = 8 then MCount = 1

TRAN-SITION	SOURCE STATE	TARGET STATE	ACTION
T13	8	6	ChFackReq =1, ChFackReq_S =1, /*set qualifier/acknowledgment request*/ if ChFack_C = 0 then ChFack_C_e =1 use SD, setSD =1, SDset_S =1 MCount = MCount + 1 if MCount = 8 then MCount = 1
T14	7	6	ChFackReq =0, ChFackReq_S =0, ChFack_C_e =0, use SD, setSD =1, SDset_S =1 MCount = 0

INTERNAL ITEMS	TYPE	DEFINITION
MTimer	Timer	This timer checks the arrival of the next valid SPDU from the FS-Device in time. The FS-Master Tool is responsible to define this watchdog time. Value range is 1 to 65 535 ms.
ChFack_C_e	Flag	By means of this auxiliary variable (bit) it is ensured that the safe state will be left only after a signal change of ChFack_C from 0 → 1 (edge). Without this mechanism an operator could overrule safe states by permanently actuating the ChFack_C signal.
FAULTS	Flags	Permanent storage of the following errors or failures can be omitted within the FS-Master, if it can be assumed that the upper-level FSCP system prevents from automatic restart of safety functions (no FS-Device persistence): - MCommErr or MTimeout - DCommErr, including counter/port error (Status&DCnt Bit 1 and PortNum) - DTimeout (Status&DCnt Bit 0)
Expected SPDU	Guard	Mirrored inverted counter (DCount_i = inverted MCount)
Not old SPDU	Guard	Counter value ≠ value of previous SPDU
do CRC_check	Activity	SCL calculates CRC signature across received SPDU while signature value = "0" and compares with received CRC signature
do PortNum_check	Activity	SCL checks whether PortNum carries the correct FS-Master port number
do Counter_check	Activity	SCL checks whether DCount carries an expected value (mirror)
NOTE Variables within ACTIONS are defined in 11.3		

11.5.3 SCL state machine of the FS-Device

Figure 46 shows the corresponding FS-Device state machine.

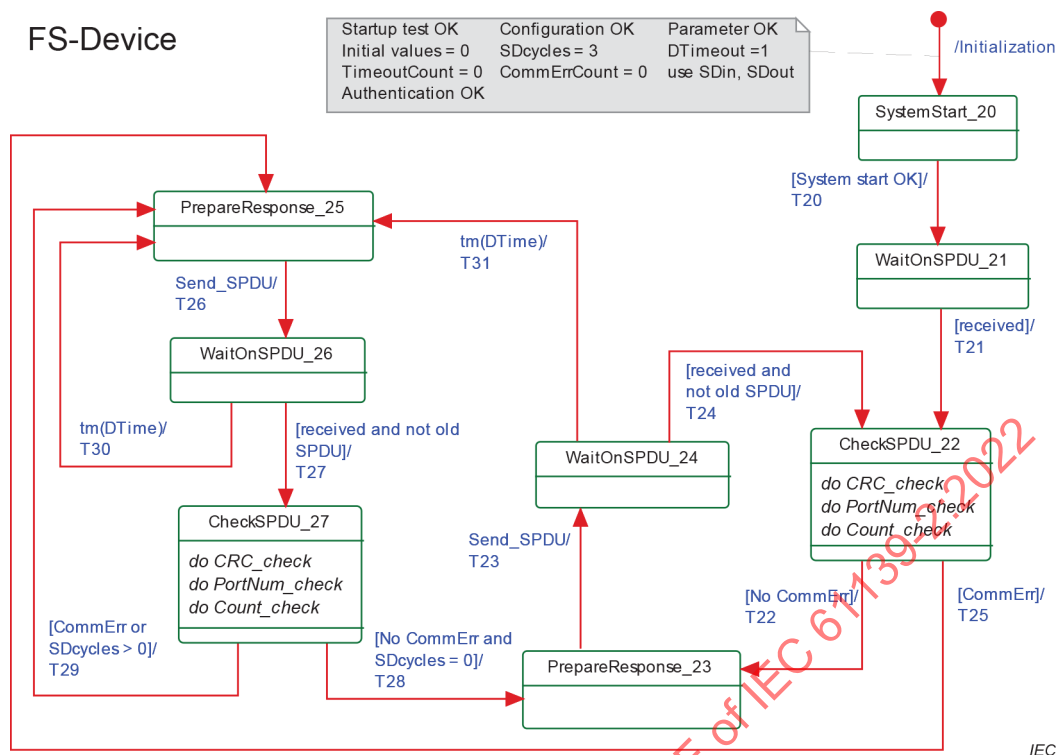


Figure 46 – SCL state machine of the FS-Device

The terms used in Figure 46 are defined in Table 39.

Table 39 – Definition of terms used in SCL state machine of the FS-Device

Term	Definition
CommErr	The SCL of the FS-Device detected a CRC or counter/port error in the received SPDU
CommErrCount	See INTERNAL ITEM in Table 40
SDcycles	See INTERNAL ITEM in Table 40
DTimeout	FSP_WatchdogTime expired
TimeoutCount	See INTERNAL ITEM in Table 40

Table 40 – FS-Device SCL states and transitions

STATE NAME		STATE DESCRIPTION	
Initialization		Initialization of the FS-Device upon power-on. Upon power-on, the FS-Device (actuator) sets the PDout to "0". Upon power-on the FS-Device (sensor) is sending "0".	
20 SystemStart		Immediately after FSP parameterization the FS-Device sets PDout to SDout values. Immediately after FSP parameterization it is sending Process Data (PD).	
21 WaitOnSPDU		SCL is waiting on next SPDU from FS-Master. SPDU with all octets "0" shall be ignored.	
22 CheckSPDU		Check received SPDU from FS-Master for CRC errors; set ChFAckReq_DC = ChFAckReq. When guard "No CommErr" = true → T22. When guard "CommErr " = true → T25	
23 PrepareResponse		Preparation of (<i>regular</i>) SPDU response for the FS-Master (response message)	
24 WaitOnSPDU		SCL is waiting on next (<i>regular</i>) SPDU from FS-Master not carrying the previous MCount. After FSP_WatchdogTime expired → T31. When SPDU received and guard "MCounter_incremented" = true → T24 (<i>regular</i> cycle)	
25 PrepareResponse		Preparation of (<i>exceptional</i>) SPDU response for the FS-Master (due to DTimeout or DCommErr = error report bits in Status&DCnt Byte)	
26 WaitOnSPDU		SCL is waiting on next SPDU from FS-Master not carrying the previous MCount. SPDU with all octets "0" shall be ignored. After FSP_WatchdogTime expired → T30. When SPDU received and guard "MCounter_incremented" = true → T27	
27 CheckSPDU		Check received SPDU from FS-Master for CRC errors; set ChFAckReq_DC = ChFAckReq. When guard "No CommErr and SDcycles =0" = true → T28. When guard "CommErr or SDcycles >0" = true → T29	
TRAN-SITION	SOURCE STATE	TARGET STATE	ACTION
T20	20	21	–
T21	21	22	–
T22	22	23	use PDin_D, DCommErr = 0, /*Status&DCnt, Bit 1*/ DTimeout = 0, /*Status&DCnt, Bit 0*/ DCount_i = MCount_inv, restart DTimer if SDcycles <> 0 then use SDout, setSD_DC=1, SDset = 1, /*during SDcycles: SDset_p =1*/ SDcycles = SDcycles – 1 else use PDout, setSD_DC=0, SDset = 0 if setSD =1 /*use_SD =1*/ then use SDout, setSD_DC=1, end if end if
T23	23	24	if SDset_DS = 1 /* FS-Device fault*/ then SDset = 1
T24	24	22	–
T25	22	25	use PDin_D, use SDout, setSD_DC=1, SDset = 1, DCommErr =1, /*Status&DCnt, Bit 1*/ CommErrCount = 1, DCount_i = MCount_inv, SDcycles = 3, restart DTimer
T26	25	26	–
T27	26	27	–
T28	27	23	use PDin_D, use SDout, setSD_DC=0, SDset = 0, DCount_i = MCount_inv, DCommErr =0, /*Status&DCnt, Bit 1*/ DTimeout =0, /*Status&DCnt, Bit 0*/ restart DTimer,

TRAN- SITION	SOURCE STATE	TARGET STATE	ACTION
T29	27	25	use PDin_D, use SDout, setSD_DC=1, SDset = 1, DCount_i = MCount_inv, restart DTimer if CommErr then DCommErr = 1, /*Status&DCnt, Bit 1*/ CommErrCount = 1, SDcycles = 3, else SDcycles = SDcycles -1 if CommErrCount = 1 then DCommErr = 1, /*Status&DCnt, Bit 1*/ CommErrCount = 0 else DCommErr = 0 /*Status&DCnt, Bit 1*/ if TimeoutCount = 1 then DTimeout = 1, /*Status&DCnt, Bit 0*/ TimeoutCount = 0 else DTimeout = 0 /*Status&DCnt, Bit 0*/
T30	26	25	use PDin_D, use SDout, setSD_DC=1, SDset =1, DTimeout =1, /*Status&DCnt, Bit 0*/ TimeoutCount =1, SDcycles = 3, restart DTimer, DCount_i = MCount_inv
T31	24	25	use PDin_D, use SDout, setSD_DC=1, SDset =1, DTimeout =1, /*Status&DCnt, Bit 0*/ TimeoutCount =1, SDcycles = 3, restart DTimer, DCount_i = MCount_inv

INTERNAL ITEM	TYPE	DEFINITION
MCount_inv	Variable	Inverse value of current MCount value
SDcycles	Counter	This decremental counter is used to cause the SCL setting SDout and SDset for at least 3 cycles during start-up and after a fault. Value range is 3 to 0.
CommErrCount	Counter	This decremental counter is used to guarantee the bit "DCommErr" within the Status&DCnt Byte is being set at least for 1 cycle or for a maximum of 2 cycles. Value range is 1 to 0.
TimeoutCount	Counter	This decremental counter is used to guarantee the bit "DTimeout" within the Status&DCnt Byte is being set at least for 1 cycle or for a maximum of 2 cycles. Value range is 1 to 0.
do CRC_check	Activity	SCL calculates CRC signature across received SPDU while signature value = "0" and compares with received CRC signature
do PortNum_check	Activity	SCL checks whether PortNum carries the correct FS-Master port number
do Counter_check	Activity	SCL checks whether MCount carries "0" (first SPDU) or either "0" or the expected subsequent value (all other SPDUs)

NOTE Variables within ACTIONS are defined in 11.3

It is very unlikely for an FS-Device to receive SPDUs with all octets "0". The SCL within the FS-Device shall ignore such an SPDU. Normally, at least the CRC signature will be "1" if Process Data and Control Byte are "0" according to the rules in 11.4.7.

11.5.4 Sequence charts for several use cases

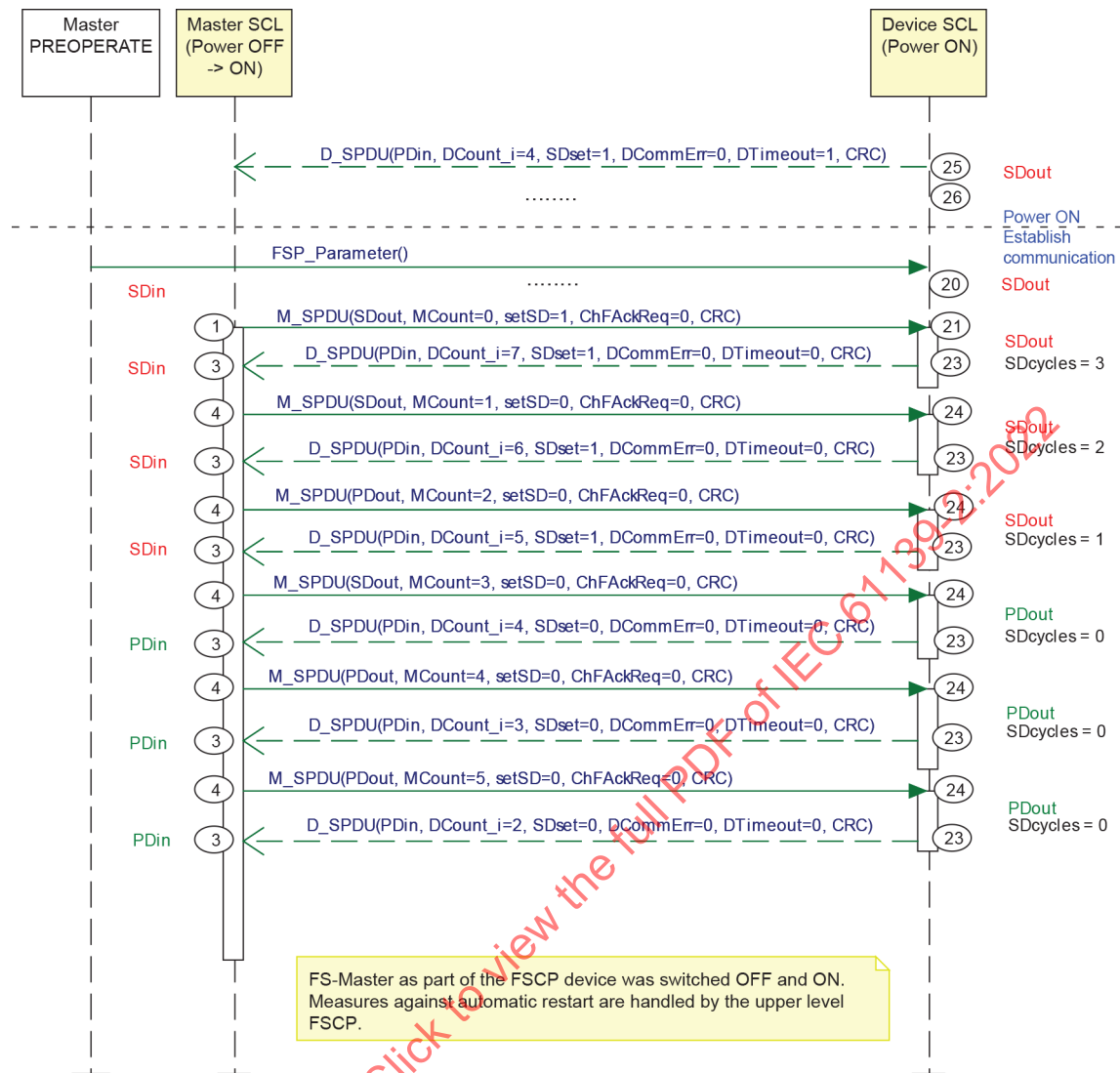
11.5.4.1 FS-Master and FS-Device both with power ON

Figure 47 shows the sequence chart of a regular start-up of both FS-Master and FS-Device.



Upon power-on both FS-Master and FS-Device are providing SDin (PDin = "0") and SDout (PDout = "0") respectively. Both keep these values for 3 communication cycles (SDcycles) before switching to the regular mode, where only the MCounter and DCounter values are changing.

Figure 48 shows the sequence chart of regular operation while FS-Master has been switched OFF and ON again.



IEC

Figure 48 – FS-Master power OFF → ON

The FS-Device communication part is always powered by the FS-Master. Thus, if the FS-Master is switched OFF and ON, the FS-Device is just following, and a regular start-up occurs. Since the FS-Master is part of an upper-level FSCP system, this FSCP system is responsible to prevent from automatic restart of safety functions in this case.

11.5.4.3 FS-Device with delayed SCL start

Figure 49 shows the sequence chart when the SCL start within the FS-Device is delayed.



Figure 49 – FS-Device with delayed SCL start

This diagram shows how an FS-Master SCL waits on the SCL of the FS-Device in case of delays. The initial SPDU of the FS-Master is repeated by the SDCI transmission system (black channel) until the SCL of the FS-Device is ready to process in state 21.

As long as the SCL of the FS-Device is not ready, the response SPDU contains all "0" and the FS-Master SCL will ignore such an SPDU. PDvalid/invalid of SDCI is reserved for the non-safety part of the entire message.

11.5.4.4 FS-Device with power OFF and ON

Figure 50 shows the sequence chart when the FS-Device switches power OFF and ON again.

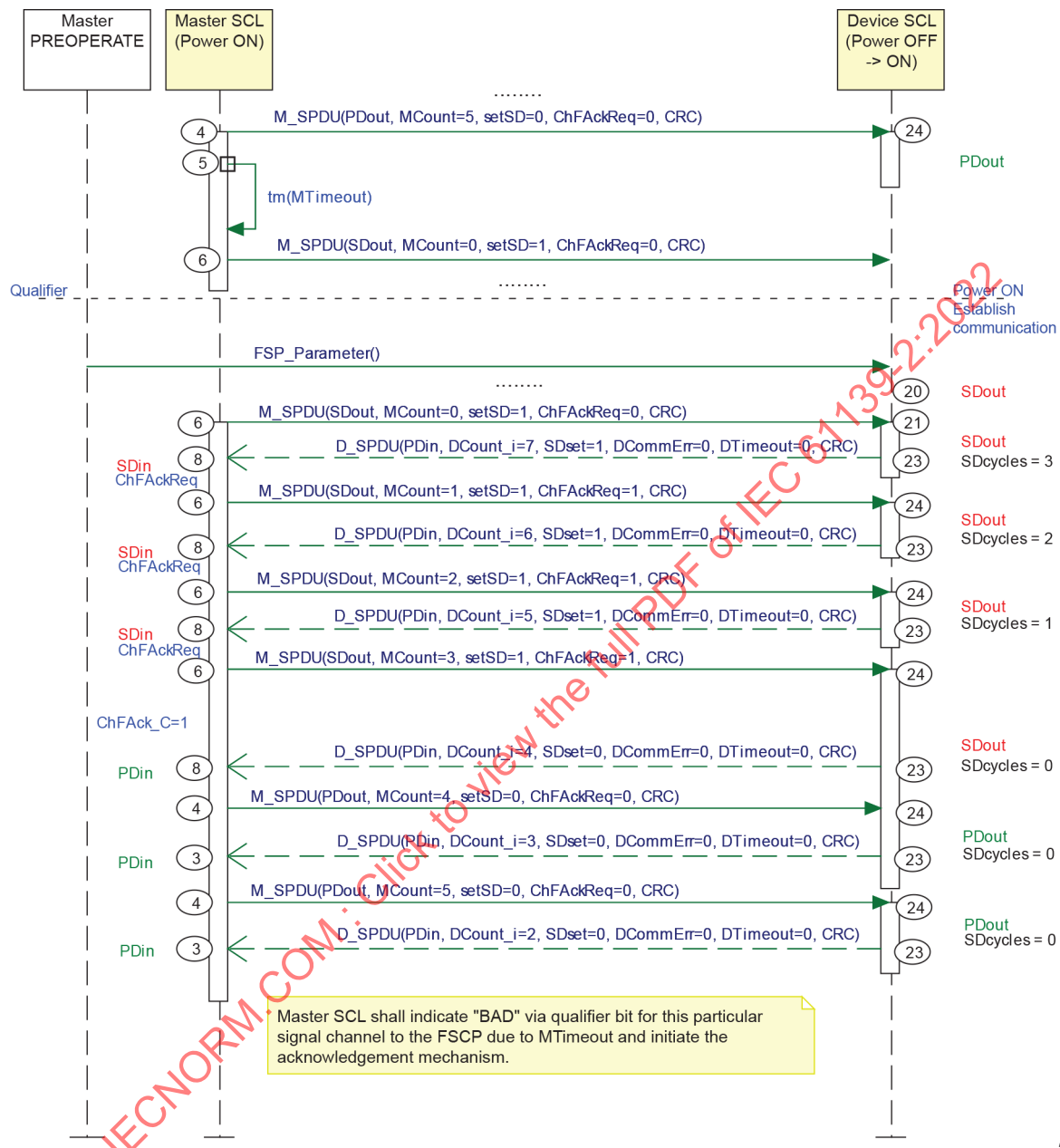


Figure 50 – FS-Device with power OFF and ON

This case assumes for example a short unplug and plug of the FS-Device causing a FAULT (MTimeout) on the FS-Master side. This FAULT causes a Qualifier bit to be set that requires via ChFackReq (=1) an acknowledgment via ChFack_C (=1). FS-Master and FS-Device keep SDin and SDout until this acknowledgment arrived.

11.5.4.5 FS-Master detects CRC signature error

Figure 51 shows the sequence chart when the FS-Master detects a CRC signature error.

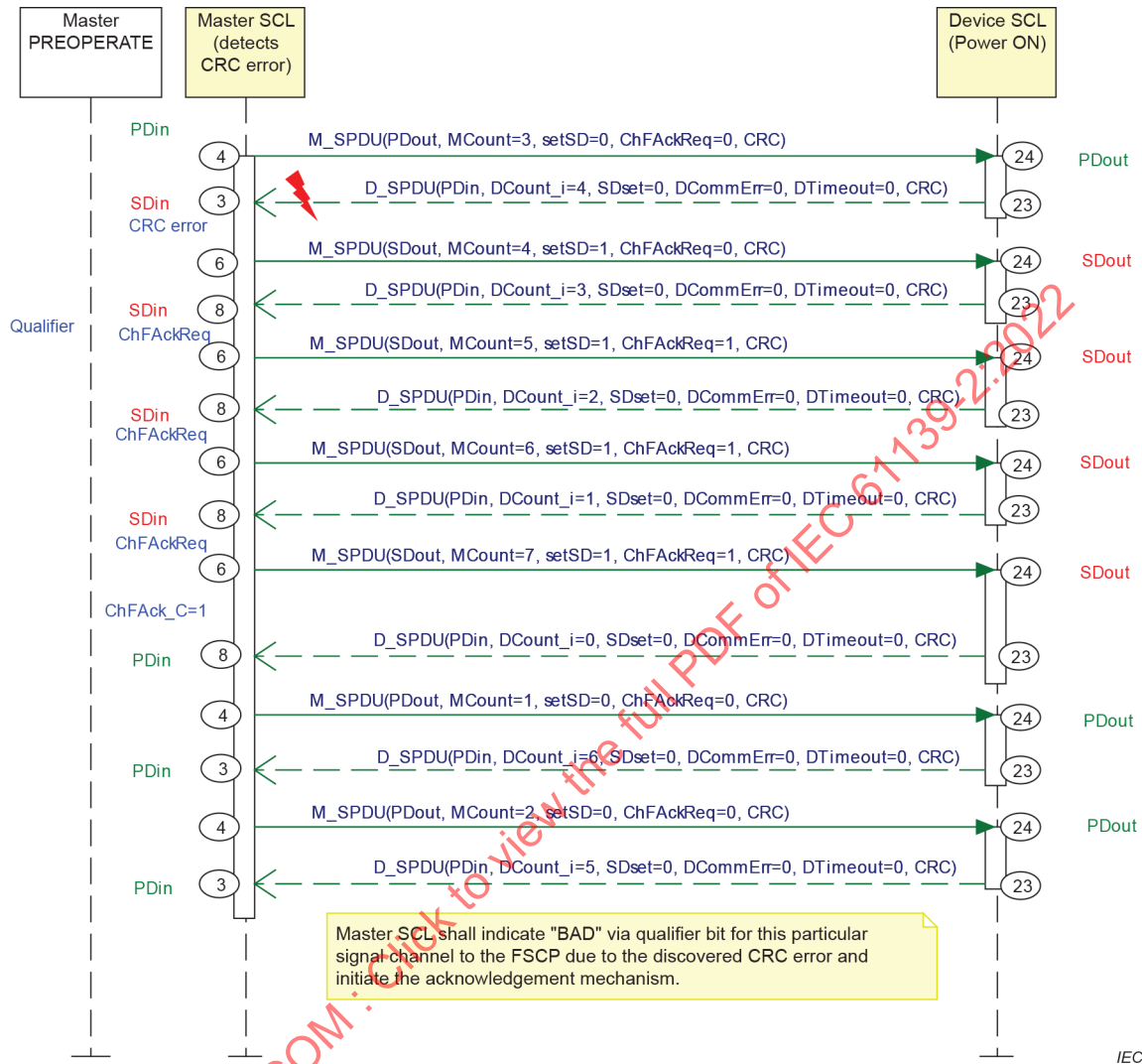


Figure 51 – FS-Master detects CRC signature error

FS-Master received an SPDU with falsified data or falsified CRC signature which results in a "CRC error" (MCommErr). Both FS-Master and FS-Device switch to SDin and SDout respectively and the FS-Master/Gateway creates a qualifier bit and indicates a ChFackReq signal. This signal is indicated also to the FS-Device via ChFackReq (=1) for indication via LED (light emitting diode) to the user.

FS-Master and FS-Device keep SDin and SDout until the acknowledgment ChFack_C (=1) arrived.

11.5.4.6 FS-Device detects CRC signature error

Figure 52 shows the sequence chart when the FS-Device detects a CRC signature error.

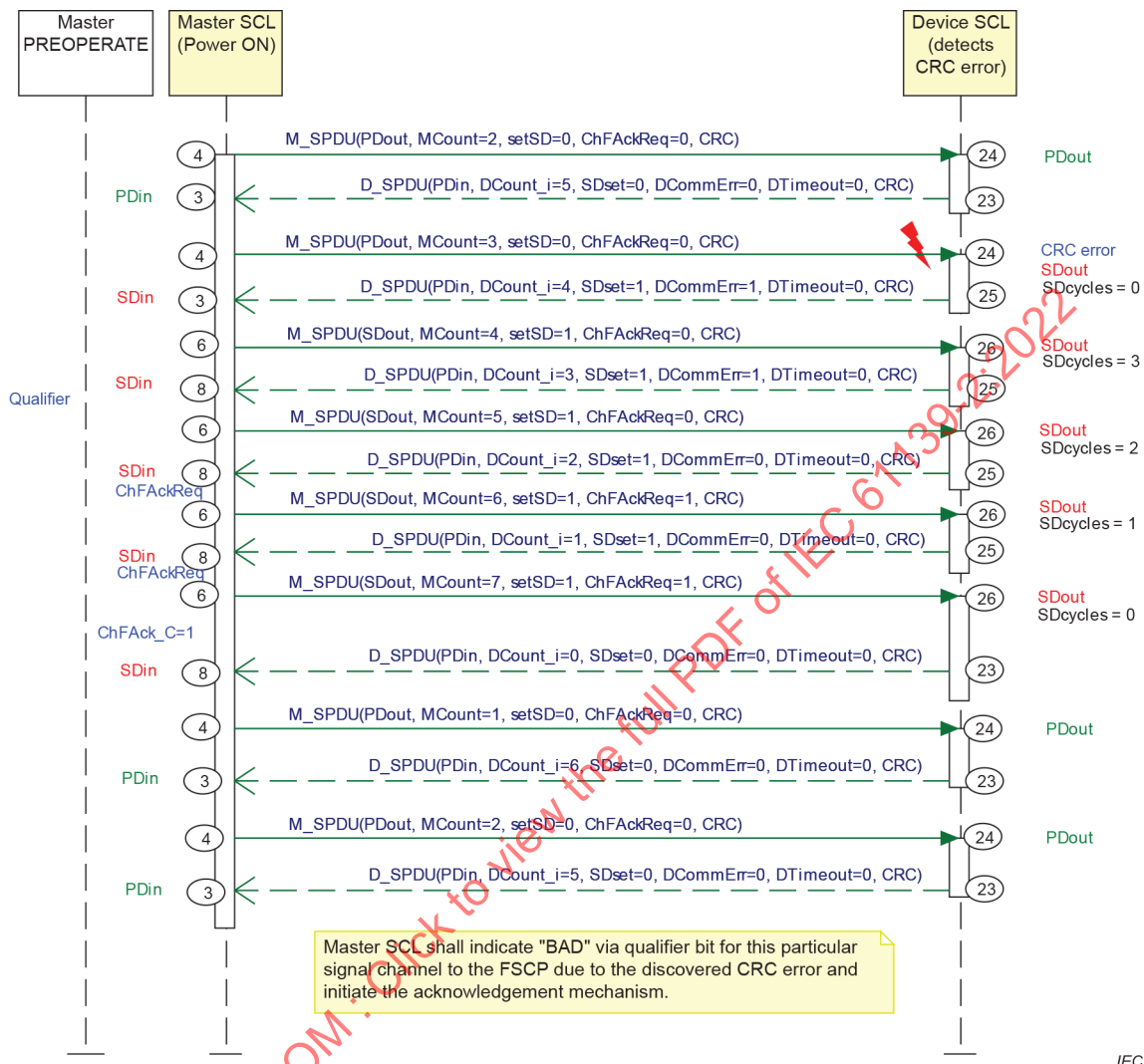


Figure 52 – FS-Device detects CRC signature error

FS-Device received an SPDU with falsified data or falsified CRC signature which results in a "CRC error" (DCommErr). Both FS-Master and FS-Device switch to SDin and SDout respectively caused by FS-Device Status Byte information (SDset=1 and DCommErr=1). The FS-Master/Gateway creates a qualifier bit and indicates a ChFackReq signal. This signal is indicated also to the FS-Device via ChFackReq (=1) for indication via LED (light emitting diode) to the user.

The FS-Device runs through 3 SDcycles and afterwards FS-Master and FS-Device keep SDin and SDout until the acknowledgment ChFack_C (=1) arrived.

11.5.5 Monitoring of safety times

Figure 53 illustrates SDCI times and safety times.

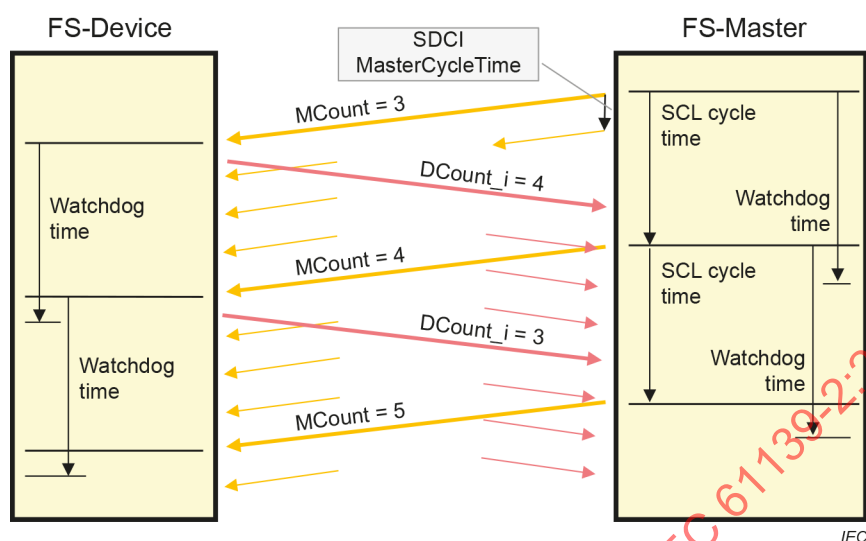


Figure 53 – Monitoring of the SCL cycle time

The base SDCI system ("black channel") transmits SPDUs within the MasterCycleTime (see IEC 61131-9) from the FS-Master to the FS-Device and back. The same SPDU, for example with MCount = 3, may be sent several times before the Safety Communication Layer (SCL) starts the next SCL cycle with MCount = 4. In the meantime, the FS-Master received the response SPDU from the FS-Device with DCount_i = 4.

Table 41 shows timing constraints.

Table 41 – Timing constraints

Item	Constraints	
Synchronization	At each start-up and after an FS-Master timeout, the FS-Master SCL uses MCount = 0	
SCL cycle time	The SCL cycle time comprises the transmission time of the FS-Master SPDU, the FS-Device processing time, the transmission time of the FS-Device response SPDU, and the FS-Master processing time until the next FS-Master SPDU (see Figure 53)	
Watchdog time	The entire SCL cycle time is monitored by the watchdog timer, whose time value is defined by the parameter FSP_Watchdog (see A.2.6).	
Counter check	The counter values are included in the cyclic CRC signature calculation. An incorrect CRC signature value will already lead immediately to a safe state. The immediate counter check in some states is used for discarding "outdated SPDUs".	
Repetition	Repetition in case of detected incorrect CRC signatures is not provided	
PFH-Monitor	The FS-Master holds the information about the reliability of both SPDU transmissions from the FS-Device and to the FS-Device (see Table 32, bit 1). Thus, the FS-Master monitors the average frequency of a dangerous failure within a given time frame (PFH-Monitor time). The FS-Master state machine is designed such that any corrupted SPDU leads always to a safe state. Whenever the unlikely event of a detected corrupted SPDU occurs during the shift of production or operation, the responsible operator is assigned to play the role of the PFH-Monitor and can tolerate the indication and acknowledge it. In case of frequent indications more often than once per PFH-Monitor time, a check of the installation or the transmission quality should be performed (see Clause H.6).	
PFH-Monitor time (h)	10	FSP_ProtMode = 0x01; CRC-16, see A.2.5
	10	FSP_ProtMode = 0x02; CRC-32, see A.2.5

11.5.6 Reaction in the event of a malfunction

11.5.6.1 General

Subclauses 11.5.6.2 to 11.5.6.10 specify possible communication errors. They are derived from 5.3 in IEC 61784-3:2021 and refer to Table 27 in this document. Additional notes are provided to indicate the typical behavior of the SDCI black channel.

11.5.6.2 Corruption

Messages may be corrupted due to errors within a communication participant, due to errors on the transmission medium, or due to message interference.

NOTE 1 Bit falsifications within messages during transfer is a normal phenomenon for any standard communication system, such errors are detected at receivers with high probability by use of a hash function, in case of SDCI a checksum (CKT or CKS), and the message is ignored (see IEC 61131-9:—, Clause A.1). After two retries, the Master initiates a complete restart with wake-up.

NOTE 2 If the recovery or repetition procedures take longer than a specified deadline, a message is classed as "Unacceptable delay" (see 11.5.6.6).

Countermeasures:

The CRC signature as specified in 11.4.7 detects the bit errors in messages between FS-Master and FS-Device to the extent required for SIL3 applications. The CRC signature is generated across the SPDU including the PD or SD data, the port number, and the Control&MCnt or Status&DCnt octet for cyclic communication.

At start-up, the FSP parameters are sent once to the FS-Device via ISDU services. They are secured by the 16-bit FSP_ProtParCRC signature. The frequency of its occurrence is assumed to be 1/day as parameter for the calculation of the residual error rate.

11.5.6.3 Unintended repetition

Due to an error, fault or interference, messages are repeated.

NOTE Repetition by the sender is a normal procedure when an expected acknowledgment/response is not received from a target station, or when a receiver station detects a missing message and asks for it to be resent.

Countermeasures:

The data within the black channel are transferred cyclically. Thus, an incorrect message/SPDU with the latest received counter value that is inserted once will be ignored. The thereby possible delay of a demand can be one DTime or MTime respectively.

11.5.6.4 Incorrect sequence

Due to an error, fault or interference, the predefined sequence (for example natural numbers, time references) associated with messages from a particular source is incorrect.

NOTE In SDCI only one sequence is active from one source, the message handler.

Countermeasures:

The receiver will detect any incorrect sequence due to the stringently sequential expectation of the MCount and DCount values.

11.5.6.5 Loss

Due to an error, fault or interference, a message or acknowledgment is not received.

Countermeasures:

Lost information will be detected by stringently changing and examining the MCount/DCount and/or MTime/DTime within the safety communication layer of the respective receiver.

11.5.6.6 Unacceptable delay

Messages may be delayed beyond their permitted arrival time window, for example due to bit falsifications in the transmission medium, congested transmission lines, interference, or due to communication participants sending messages in such a manner that services are delayed or denied (for example FIFOs in switches, bridges, routers).

NOTE 1 SDCI provides a point-to-point communication interface with defined message sequences and thus the probability for congestion and storage of messages is extremely low.

Countermeasures:

A consecutive counter in each message (MCount/DCount) together with a watchdog timer (MTime/DTime) will detect unacceptable delays.

11.5.6.7 Insertion

Due to a fault or interference, a message is received that relates to an unexpected or unknown source entity.

NOTE 1 These messages are additional to the expected message stream, and because they do not have expected sources, they cannot be classified as Correct, Unintended repetition, or Incorrect sequence.

NOTE 2 SDCI provides a point-to-point communication interface (Port) and thus the probability for insertion of messages is extremely low.

Countermeasures:

The receiver will detect any incorrect sequence due to the stringently sequential expectation of the MCount and DCount values.

11.5.6.8 Masquerade

Due to a fault or interference, a message is inserted that relates to an apparently valid source entity, so a misdirected non-safety related message may be received by a safety related participant, which then treats it as safety related correct message.

NOTE 1 Communication systems used for safety-related applications can use additional checks to detect Masquerade, such as authorised source identities and passphrases or cryptography.

NOTE 2 SDCI provides a point-to-point communication interface (Port) and thus the probability for insertion of messages is extremely low.

Countermeasures:

In case of NSR data instead of a regular SPDU, the CRC signature mechanism of the SCL will detect this incident.

After changes of wiring, the FS-Devices can detect misconnections through the FSP_Authenticity1/2 and FSP_Port parameters (see A.2.1 and A.2.2) at start-up.

11.5.6.9 Addressing

Due to a fault or interference, a safety related message is delivered to the incorrect safety related participant, which then treats reception as correct. This includes the so-called loopback error case, where the sender receives back its own sent message.

NOTE The probability of not detecting a misdirected non-safety related message is lower than the probability of not detecting a misdirected safety related message since the SPDU structures are similar due to the shared protocol procedures.

SDCI provides a point-to-point communication interface (Port) and thus the probability for insertion of messages is extremely low. However, FS-Master may use internal bus mechanisms to address ports.

Countermeasures:

Port addressing errors can be detected by the port number (PortNum) within the SPDU.

After changes of wiring, the FS-Devices can detect misconnections through the FSP_Authenticity1/2 and FSP_Port parameters (see A.2.1 and A.2.2) at start-up.

11.5.6.10 Loop-back

A special addressing error is the so-called loopback error case, where the sender receives back its own sent message.

Countermeasures:

SDCI-FS provides for inverted values of MCount as DCount and inverted values of the port number (PortNum) returned from the FS-Device.

11.5.7 Start-up (communication)

An FS-Device starts always after an FS-Master since the FS-Master shall be the only one to power-up at least the communication part of the FS-Device. Both devices usually require time for safety self-tests that may exceed the standard timings defined in IEC 61131-9.

Due to the initial behavior of an FS-Device as an OSSDe, the start-up is coordinated and specified in 5.7, 7.2, and 7.3.

The start-up of the underlying SDCI communication system is specified in IEC 61131-9 and illustrated in an abstract and simplified manner in Figure 57 for easier comprehension. Any deviating FSP authenticity or protocol parameter CRC signature shall lead to a safe state of the particular FS-Master Port and prevent the SCL from being started.

11.6 SCL management

11.6.1 Parameter overview (FSP and FST)

Annex A specifies several functional safety related parameters for communication protocol services (FSP) as well as for the handling and integrity purposes of FS-Device technology parameters (FST).

The parameters are subdivided into 4 groups:

- Authenticity
- Safety communication
- FS-I/O structure description
- Auxiliary parameters

The authenticity parameters combine the safety connection ID ("A-Code") of the FS-Master (assigned by the upper-level FSCP system) with the port number of the connected FS-Device. Due to the point-to-point nature of the FS-Device communication with its Master, a one-time check at start-up is sufficient to ensure authenticity (see 11.7.4).

The Safety Communication Layers (SCL) require parameters for protocol versions, protocol modes such as CRC-16 or CRC-32, watchdog for timeliness, CRC signature to secure technology parameters, and a CRC signature to secure the safety communication parameters.

The next group contains a description of the FS I/O data structure, the FS-Device wants to exchange with the FSCP-Host. This description facilitates the mapping to the description which some FSCP systems require for set-up. During the mapping process the FS-Master tool appends the qualifier bits, which are necessary for port-selective passivation.

Auxiliary parameters are specified for several purposes. For example, to secure the functional safety parameter description within the IODD, to support the automatic calculation of approximate values of safety function response times, and to inform about the start-up self-testing time of an FS-Device until the Ready pulse appears.

Figure 54 shows an overview of the components and the activities around parameterization.

An FS-Master as a gateway comes with an associated description file for the upper-level system (FSCP). With the help of an engineering tool and these parameters, the FS-Master receives during commissioning for example its FSCP connection ID ("A-Code" for authenticity) and its FSCP watchdog time ("T-Code" for timeliness). Thus, the FSCP communication cycles are independent from the SDCI-FS communication cycles between FS-Master and FS-Device.

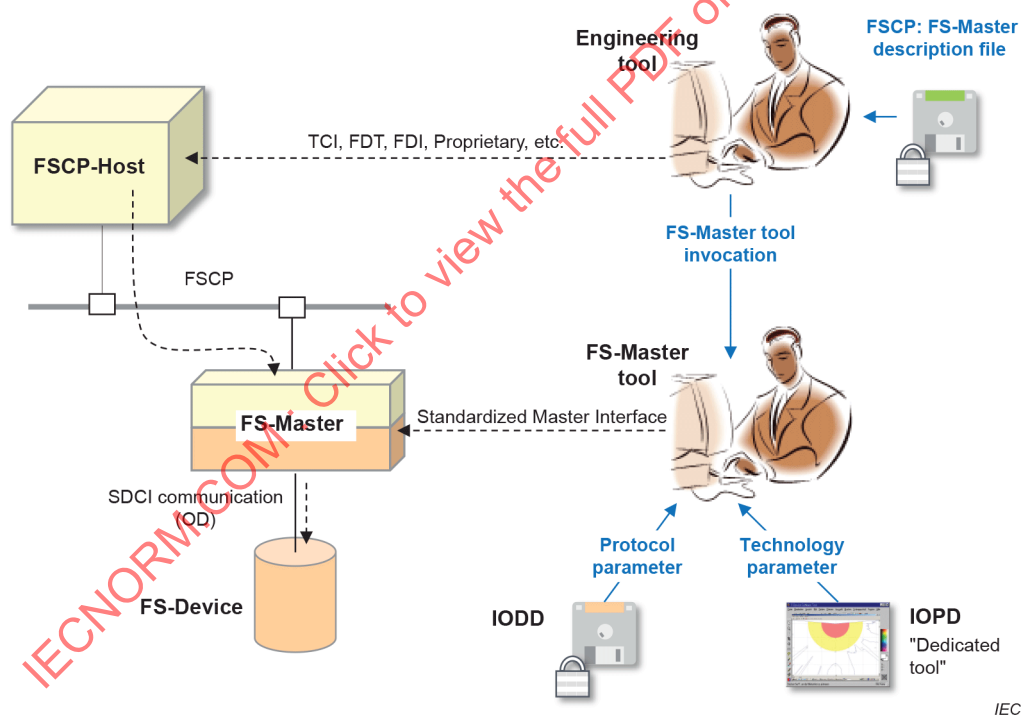


Figure 54 – Parameter types and assignments

An FS-Master with its SDCI side can be configured and parameterized with the help of its FS-Master tool. The IODD of an FS-Device contains besides the non-safety parameters also the safety section with the parameters in Annex A. The parameters can be set-up off-site or online the same way as with a non-safety system during "commissioning-test" (see Table G.1). The FSCP authenticity parameter can be copied from the engineering tool display to the FS-Master tool display (see A.2.1).

It is possible to describe a small set of technology parameters (FST) in a non-safety manner, thus allowing the usage of the SDCI standard Data Storage mechanism as described in 9.4.

However, a separate Dedicated (IOPD) Tool, developed according to IEC 61508-3 shall be used to calculate a CRC signature across the instances of the FST parameters. This CRC signature shall be entered into the respective FSP parameter (see A.2.8).

The IOPD tool uses a new standardized IOPD communication interface (DTI, see Annex F) and the same path to the FS-Device as the FS-Master tool itself.

11.6.2 Parameterization approaches

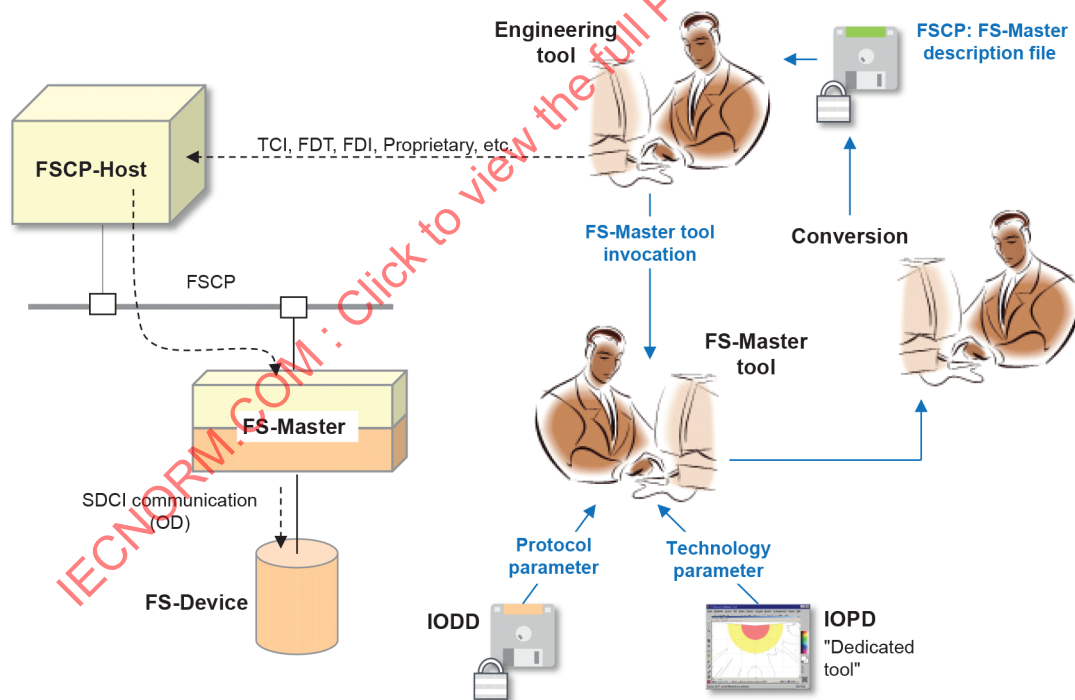
11.6.2.1 FS-Master-centric

The configuration and parameterization of a stand-alone SDCI-FS system corresponds mainly to the approach described in 11.6.1. The authenticity uses a default value in this case (see A.2.1).

Figure 54 shows a loosely coupled system, where the parameterization is performed within the SDCI-FS part. Within the FSCP system, predefined FS I/O data structures are available and can be selected during commissioning.

11.6.2.2 FSCP-Host-centric

Some automation application areas prefer an FSCP-Host-centric approach. In this case, all parameters are expected to be stored within the FSCP-Host and downloaded at start-up into the FS-Master (FSCP-subsystem) and further down into the FS-Device.



IEC

Figure 55 – FSCP-Host-centric system

Due to the fieldbus-independent design of SDCI and SDCI-FS, all parameters can for example be converted into the fieldbus device description file. It is one of the objectives of SDCI-FS to optimize the design of safety parameters such that an efficient conversion is possible.

11.7 Integrity measures

11.7.1 IODD integrity

The parameters specified in Annex A are coded in an IODD file using XML. In order to protect the safety parameter description within this file, a CRC signature ("FSP_ParamDescCRC") shall be calculated across its safety-related parts (see Annex D and Annex E.5.6). Usually, the IODD file travels many ways, and the CRC signature helps detecting potentially scrambled bits.

11.7.2 Tool integrity

When opening the IODD, the FS-Master tool (interpreter of the IODD file) shall calculate the CRC signature across the safety-related parts and compare the result with the parameter "FSP_ParamDescCRC".

During the data manipulations within the FS-Master tool as well as within Device Tools/IOPDs ("Dedicated Tools") such as display, intended modification, storage/retrieval, and down/upload, parameter values could become incorrect. It is the responsibility of the designer to develop the tools according to the software safety level requested in ISO 13849-1 or IEC 61508-3.

11.7.3 Transmission integrity

Since communication between the FS-Master tool and the FS-Device is proprietary, it is the responsibility of the FS-Master tool to ensure transmission integrity and authenticity, for example through CRC signatures and/or read back (see Table 27 and D.3.1).

11.7.4 Verification record

In either the FS-Master-centric or in the FSCP-Host-centric approach an FSP_VerifyRecord of parameter data is stored in the FS-Master per Port/FS-Device as shown in Figure 56.

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

FSP_VerifyRecord	FS-Device specific header		
FSP parameter	0		UIntegerT (32)
	1	FSP_Authenticity1	
	2	(FSCP dependent)	
	3		
	4		UIntegerT (32)
	5	FSP_Authenticity2	
	6	(extension)	
	7	(FSCP dependent)	
	8	FSP_Port	UIntegerT (8)
	9	FSP_AuthentCRC	UIntegerT (16)
	10		
	11	FSP_ProtVersion	UIntegerT (8)
	12	FSP_ProtMode	UIntegerT (8)
	13	FSP_WatchdogTime	UIntegerT (16)
	14		
	15	FSP_IO_StructCRC	UIntegerT (16)
	16		
	17	FSP_TechParCRC	UIntegerT (32)
	18		
	19		
	20		
	21	FSP_ProtParCRC	UIntegerT (16)
	22		
End of FSP_VerifyRecord			

IEC

Figure 56 – Structure of the FSP_VerifyRecord

The authenticity parameters are secured by FSP_AuthentCRC for transmission from FS-Master Tool to FS-Master and further to the FS-Device. The procedure of the FSCP authenticity acquisition from the FSCP gateway and subsequent handling of the FSP authenticity record is described in 10.4.3.3. FSP_ProtParCRC secures protocol parameters as described in 10.4.3.4.

11.7.5 Authentication

The SLM of the FS-Master uses the FSP_VerifyRecord received from Configuration Manager. Thus, the FSP_Authenticity codes within the record can be compared with the actual FSCP Authenticity values in the safety part of the Gateway.

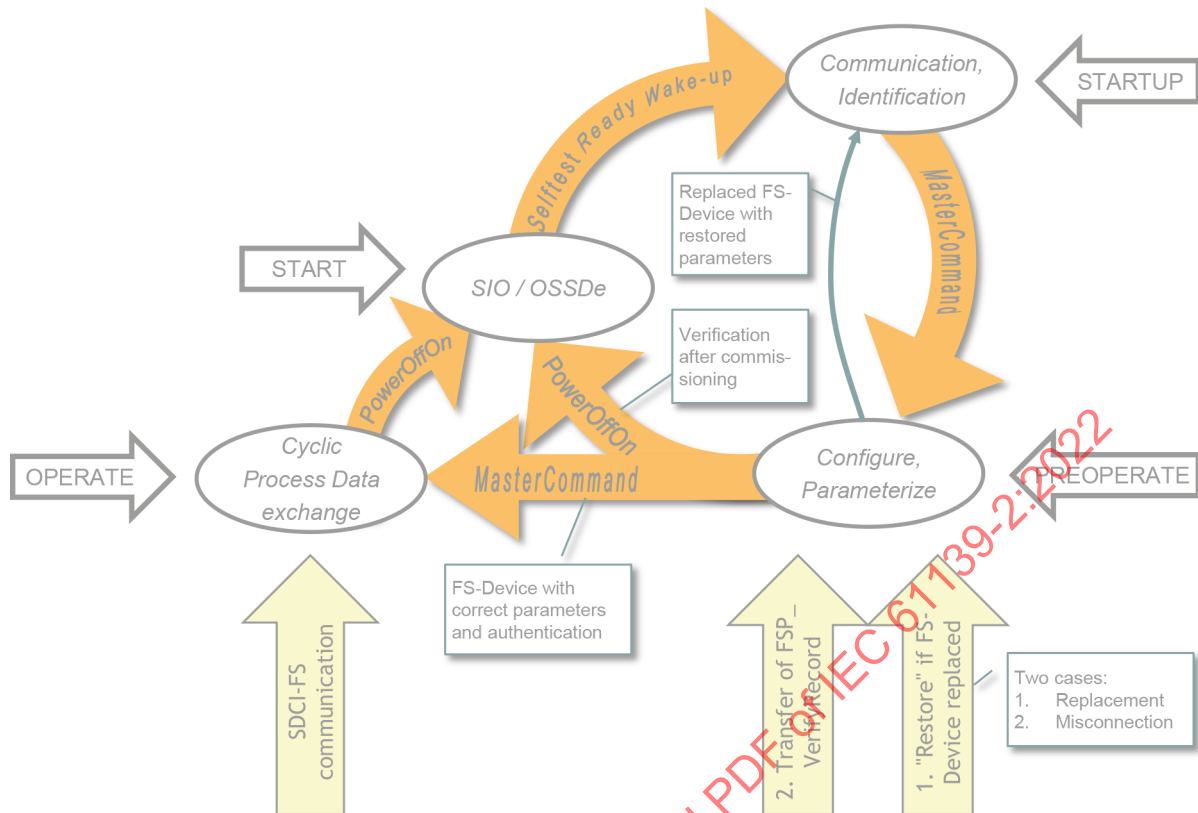
11.7.6 Storage integrity

Both records (authenticity and protocol) of Figure 56 are stored in both FS-Master and FS-Device and may fail over time (see also Table A.1).

At each regular start-up, the Configuration Manager transfers the FSP_VerifyRecord to the FS-Device during PREOPERATE as shown in Figure 57 and described in 10.4.3.1 and A.2.10. Figure 57 is derived from the Master message handler in IEC 61131-9 and provides an abstract and simplified picture of the essential states and transitions.

The FS-Device will detect a potential mismatch between the downloaded authenticity parameter set and the already stored values in the FS-Device, for example if FS-Devices are misconnected to a different Port or even to a different FS-Master (see Figure 36).

The protocol parameters are propagated to the Safety Communication Layer at each start-up.



IEC

Figure 57 – Start-up of SDCI-FS

In case the FS-Device has been replaced due to a failure, the technology specific parameters (FST) and the FSP parameters are "restored" from Data Storage if the FS-Device carries all authenticity parameters = "0". If Authenticity is not "0", the FS-Device shall ignore them and keep the existing (see 9.4, E.5.7, and step 1. in Figure 57). In this case a misconnection can be assumed, or the FS-Device has already been in use and requires testing and a reset of the authenticity parameters (see Annex G).

11.7.7 FS I/O data structure integrity

All I/O data of the connected FS-Devices should be mapped in an efficient manner into the FSCP I/O data as shown in 12.1.

Due to the additional qualifier bits required for Port-selective passivation, the original FS-Device specific data structure is not directly visible within the FSCP I/O data structure exchanged with the FSCP-Host.

The safety-related interpreter of the FS-Master Tool transfers the entire instance data together with the CRC signature to the FS I/O data mapper as shown in 10.4.3.1 (see also A.2.7).

11.7.8 Technology parameter (FST) based on IODD

One of the objectives of SDCI-FS is FS-Device exchange without tools by using the original data storage mechanism of SDCI. As a precondition, the FST-parameter description is required within the IODD (see E.5.7).

The FST parameters are displayed in this case within the FS-Master tool (see Figure 58, FST-Parameters section). Values can be assigned as with non-safety parameters only during "commissioning-test" (see Table G.1).

The user is responsible for correct values within the FS-Device using adequate validation procedures. The FS-Master Tool can assist for example via read back and display of the parameters.

Securing of the FST parameter via signature shall not be performed by the FS-Master Tool. A separate "Dedicated Tool" (Device Tool) provided by the FS-Device manufacturer shall be used instead as shown in Figure 59 and explained in the following.

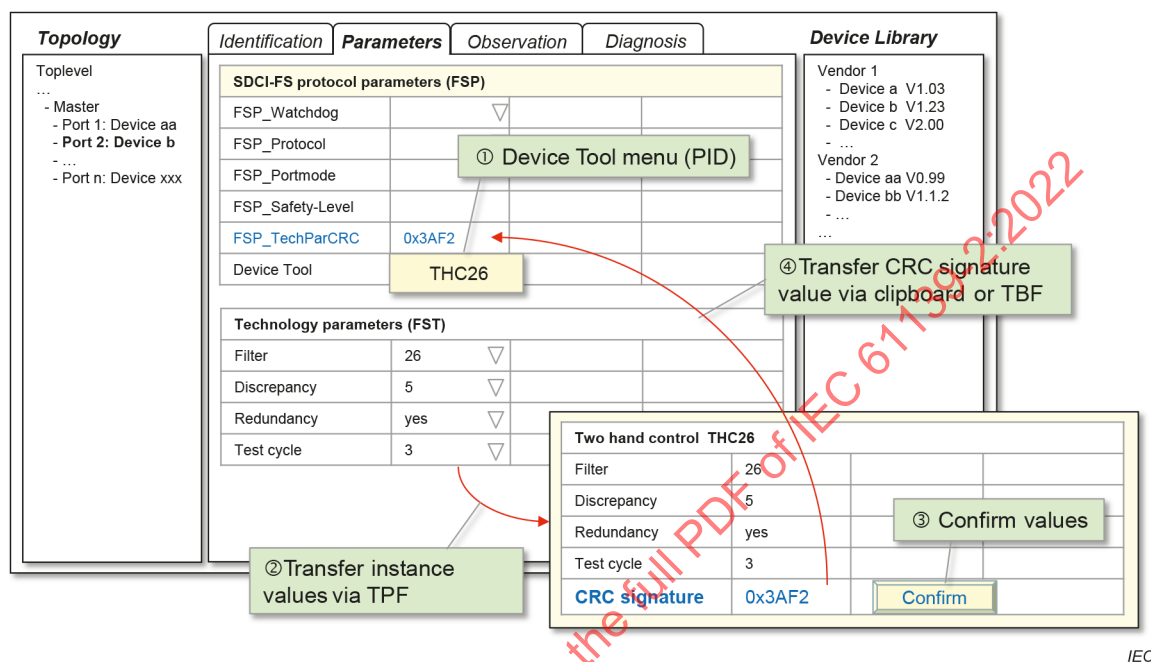


Figure 58 – Securing of FST parameters via dedicated tool

After test and validation, the Device-Tool is invoked via menu (step①). Instance values are transferred via TPF (step②) and displayed again. The user compares the instance values and confirms the correctness via the "Confirm" button (step③). The Device Tool then calculates the CRC signature across the instance data of the FST parameters (see "CRC signature" in Figure 58), which can be copied and pasted or transferred via TBF into the "FSP_TechParCRC" field of the FSP parameters (step ④).

Since this parameter is part of the FSP parameter block, the FS-Device can check the integrity of these FST parameters together with the protocol parameters.

11.7.9 Technology parameter (FST) based on existing Dedicated Tool (IOPD)

In cases, where existing safety devices already have their PC program with password protection, wizards, teach-in functions, verification instructions, online monitoring, diagnosis, special access to device history for the manufacturer, etc., an FST parameter description may not be available. Figure 59 shows an example.

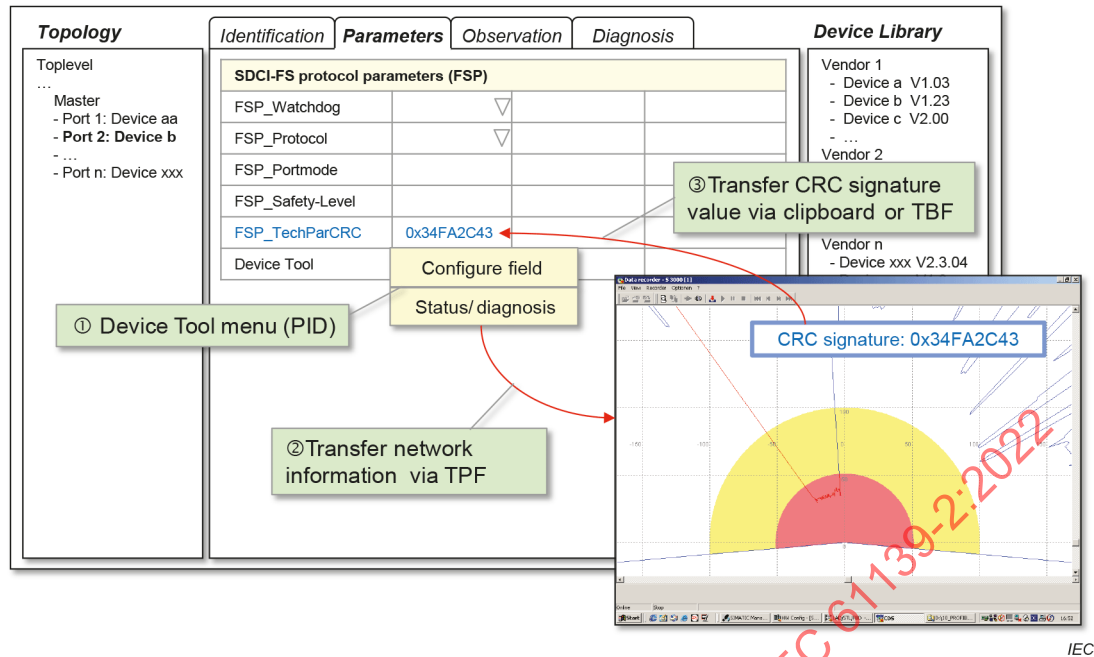


Figure 59 – Modification of FST parameters via Device Tool

Such a Device Tool requires communication with its particular FS-Device and therefore access to a Communication Server (see Annex F.5). It can be invoked via menu entries (step①) and thus jump directly into for example configuration or status/diagnosis functions. Network information is transferred via TPF (step②). After test and validation, it shall provide a display of the calculated CRC signature across the instance data, which can be copied and pasted into the "FSP_TechParCRC" field of the FSP parameters (step③).

These FS-Devices shall be supported by the data storage mechanism of SDCI and an FS-Device replacement without tools is possible.

The Data Storage limit per FS-Device is 2048 octets according to IEC 61131-9.

11.8 Creation of FSP and FST parameters

Standards for "Safety-for-Machinery" such as ISO 13849-1 and IEC 62061 require "Dedicated Tools" for the parameterization of safety devices. For the ease of development and logistics of software tools it is recommended to use the process described in Figure 60.

For FS-Devices with only a few FST parameters, no business logic, and no wizard and help systems, one particular "Dedicated Tool Framework" could be developed in a safe manner according to IEC 61508-3 and equipped with the necessary communication interfaces. Technology provider can provide such a framework for the FST parameters of a particular FS-Device (Option 1 in Figure 60). FS-Device developers can individualize the framework using the brand name, company name, and FS-Device identifiers to one dedicated tool (IOPD). This executable Dedicated Tool software can be certified by assessment bodies.

For FS-Devices with more complex FST parameters, the IOPD can be developed individually, or existing tools can be used. In both cases the tools can be equipped with the necessary communication interfaces (Option 2 in Figure 60).

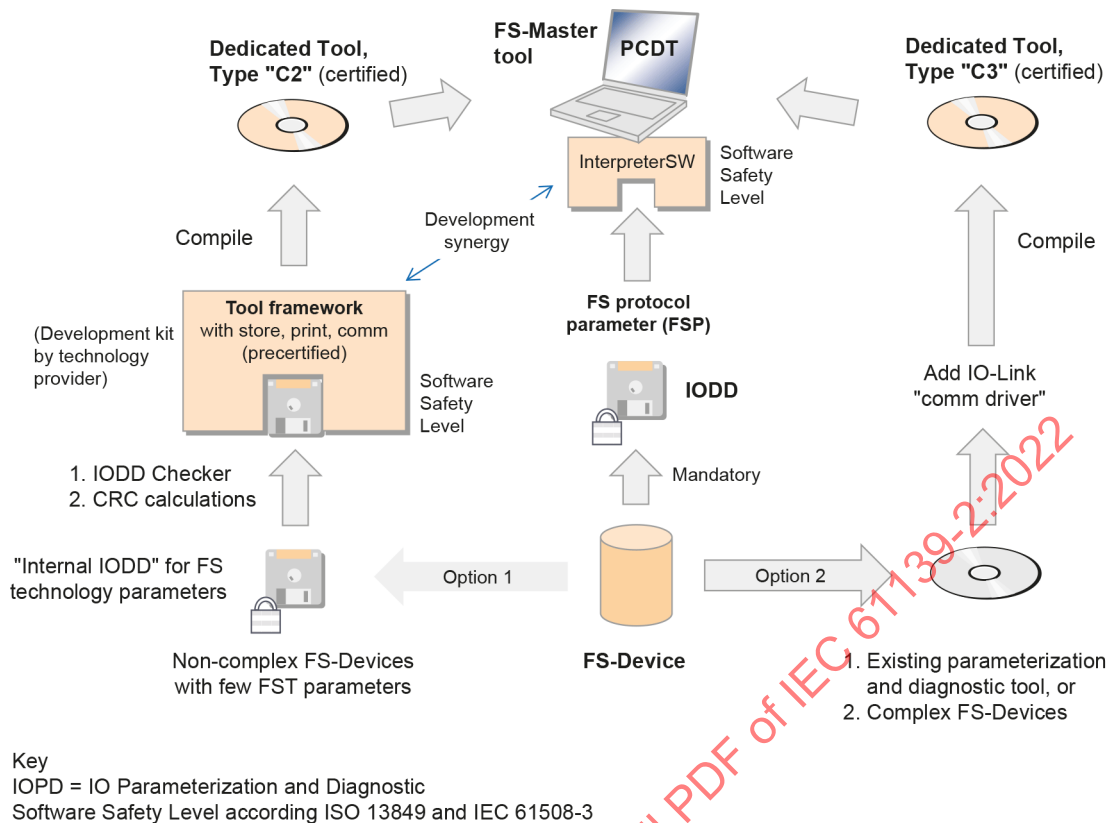


Figure 60 – Creation of FSP and FST parameters

In any case, the dedicated tool (IODD) shall calculate and display the CRC signature across all FST parameters. This signature can be copied into the entry field of the FSP parameter "FSP_TechParCRC", such that an FS-Device can verify the correctness of locally stored FST parameters after start-up and download of the FSP parameter set to the FS-Device.

For each and every FS-Device the same set of FSP (protocol) parameters shall be created in an extended IODD for SDCI-FS. This IODD is mandatory and contains the usual conventional parameters and additionally the FSP parameters.

11.9 Integration of Dedicated Tools (IODD)

11.9.1 IODD interface

Usually, a so-called Master-Tool (PCDT) provides engineering support for a Master and its Devices via Device descriptions in form of XML files (IODD). In principle, this is the same for FS-Master and FS-Device. For functional safety besides an extended IODD it is necessary for an FS-Device vendor to provide an additional Dedicated Tool (IODD) as shown in Figure 60.

In order for the IODD to communicate with its FS-Device a new standardized communication interface is required.

11.9.2 Standard interfaces

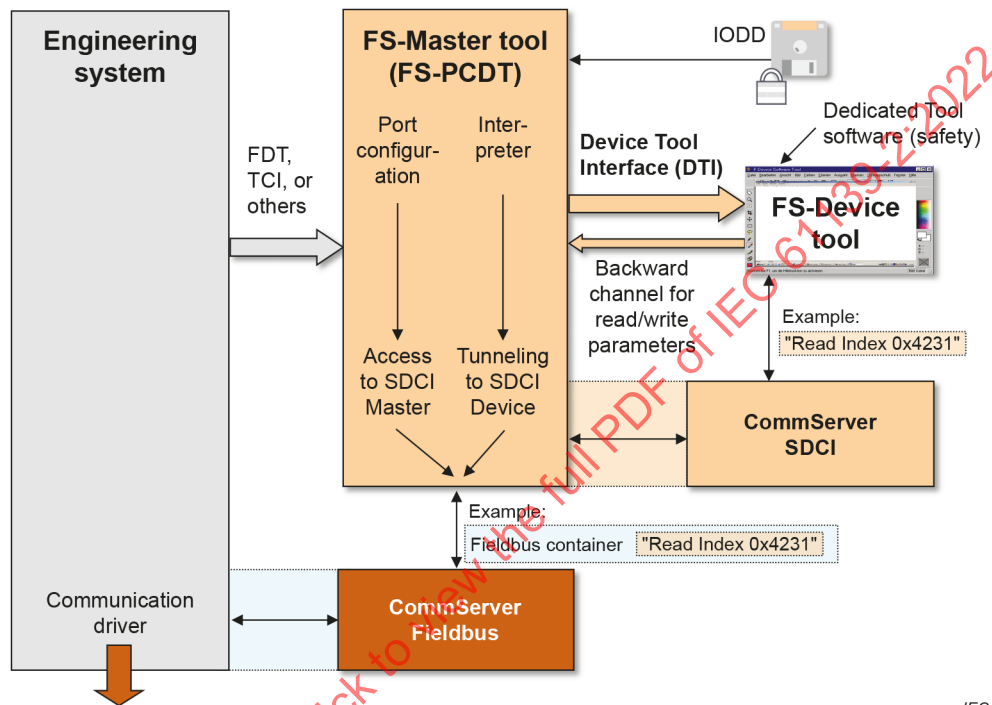
Usually, Master Tools are integrated using existing standards such as FDT, the FDI (see [11]), or proprietary solutions. Such a variety is not acceptable for FS-Devices and therefore, easy, and proven-in-use technology has been selected and adopted for SDCI-FS. It is called "Device Tool Interface" (DTI).

Annex F provides the specification for this interface.

Figure 61 illustrates the communication hierarchy of FDT and others for the fieldbus as well as the connection via the "Device Tool Interface" and the underlying SDCI communication.

The FS-Device Tool (IOPD) does not have to know about the fieldbus environment it is connected to. The example in Figure 61 illustrates how it sends a "Read Index 0x4231" service and how the FS-Master Tool packs this service into a fieldbus container and passes it to the fieldbus communication server.

The addressed FS-Master is connected to the fieldbus and receives this container. It unpacks the SDCI Read service and performs it with the addressed FS-Device connected to a Port.



IEC

Figure 61 – Example of a communication hierarchy

11.9.3 Backward channel

An FS-Master vendor does not know in advance which FS-Devices a customer wants to connect to the FS-Master ports. As a consequence, the fieldbus device description of such an FS-Master can only provide predefined "containers" for the resulting I/O data structure of the FS-Device ensembles connected to the ports. In functional safety this is even more complicated since the description of the data structures shall be coded and secured.

Because of the variety of different configurations and parameterizations of a particular FS-Device, it usually for example

- requires different I/O data structures to exchange with PLCs or hosts,
- has different reaction times due to configured high or low resolutions, and
- can have different SIL, PL, category, or PFH values impacting the overall safety level of a safety function.

The classic "fieldbus device description" to inform an engineering system is not flexible in this respect. Its advantage is the testability and certification for its interoperability with engineering tools.

Nevertheless, a "backward channel" within the tool interfaces allows for nowadays flexible manufacturing and ease of engineering and commissioning. An example is specified in [12], 4.15.5.

F.3.5 and F.9.4 specify the features for this "backward channel".

11.10 Validation

It is the responsibility of the FS-Device designer to specify the necessary verification and validation steps (for example tests; see H.6) within the user/safety manual and/or within the "Dedicated Tool" (IOPD).

11.11 Passivation

11.11.1 Motivation and means

Figure 62 illustrates the motivation for Port selective passivation. Usually for efficiency reasons, the signals 0 to 7 of FS-Devices connected to Ports are not mapped individually to an FSCP SPDU, but rather packed into one FSCP SPDU. Each of these signals can be assigned to a separate safety function n to $n+7$. If a fault occurs in one of the signal channels, a collective passivation for the entire FSCP SPDU would be necessary causing all safety functions to trip.

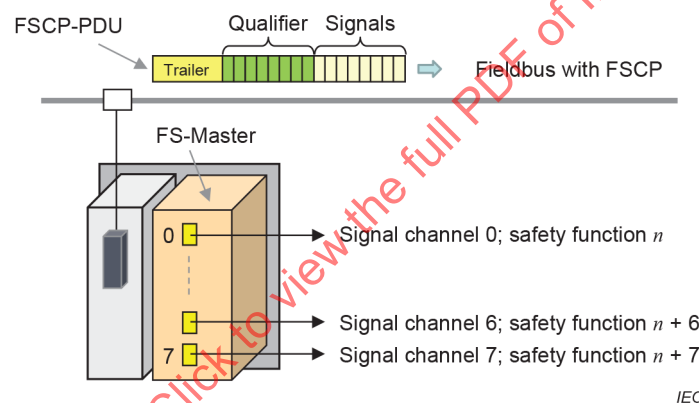


Figure 62 – Motivation for Port selective passivation

FSCPs usually provide so-called qualifier bits associated to the signal process data, which enable selectively passivating that particular signal channel and the associated safety function.

Safety of machinery usually requires an operator acknowledgment after repair of a defect signal channel to prevent from automatic restart of a machine. It is not necessary to provide the acknowledgment for each signal channel and it can be one for all channels.

11.11.2 Port selective (FS-Master)

In 11.11.1, a use case is described where the signal channel corresponds directly to a particular FS-Device. The qualifier and acknowledgment mechanism shall be implemented within the FS-Master in accordance with the specifications of the particular FSCP.

It can be helpful for the user to provide an indication in each FS-Device that an operator acknowledgment is required prior to a restart of a safety function. CB0 (ChFAckReq) within the Control&MCnt octet (see Table 31) shall be used for that purpose. It is not safety related.

Optionally, in case of FS_PortMode "OSSDe" (see 10.4.2), the signal ChFAckReq can be connected separately to the corresponding FS-Device indication (see Clause H.1).

11.11.3 Signal selective (FS-Terminal)

Figure 13 shows the use case of an FS-Terminal where an FS-Device provides several signal channels to switching devices such as E-Stop buttons.

For those FS-Devices the design rules in 11.4.9.3 apply. The acknowledgment mechanisms shall be implemented within the safety Process Data.

11.11.4 Qualifier settings in case of communication

Figure 63 illustrates the embedding of the qualifier handler in case of FS_PortModes "SafetyCom" and "MixedSafetyCom" (see 10.4.2). The services/signals "FAULT_S", "SDset_S", "ChFAckReq_S", and "ChFAck_C" are specified in 11.3.2 and 11.5.2.

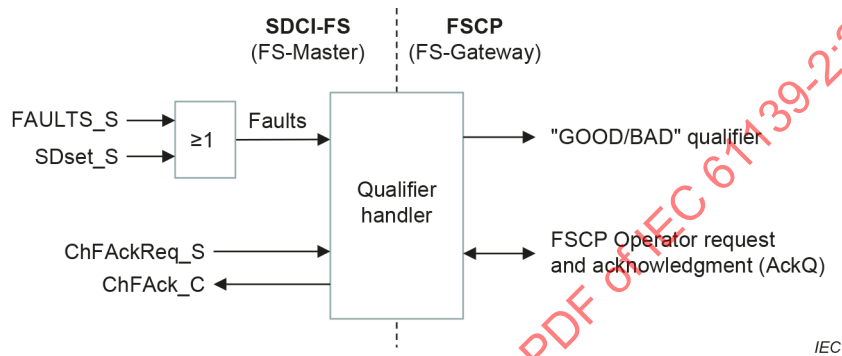


Figure 63 – Qualifier handler (communication)

The qualifier bits "GOOD/BAD" shall be set according to the definitions in Table 42 during the FSCP mapping procedure.

Table 42 – Qualifier bits "GOOD/BAD"

Faults	Qualifier	Signal state
0	GOOD	1
1	BAD	0

11.11.5 Qualifier handling in case of OSSDe

Figure 64 illustrates the embedding of the qualifier handler in case of FS_PortModes "OSSDe" (see 10.4.2). Definitions of Table 42 apply.

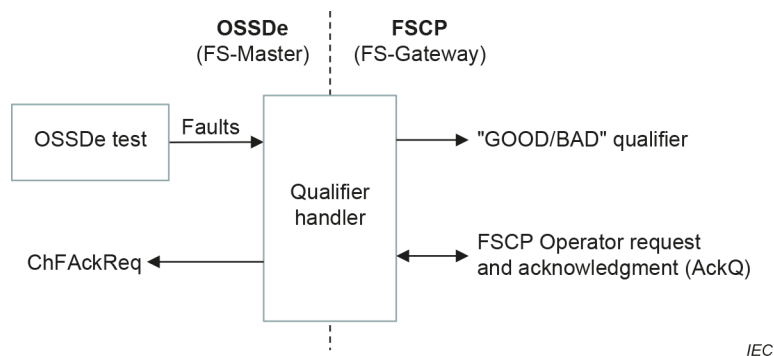


Figure 64 – Qualifier handler (OSSDe)

Figure 65 shows the state machine for the behavior of the qualifier handler (OSSDe).

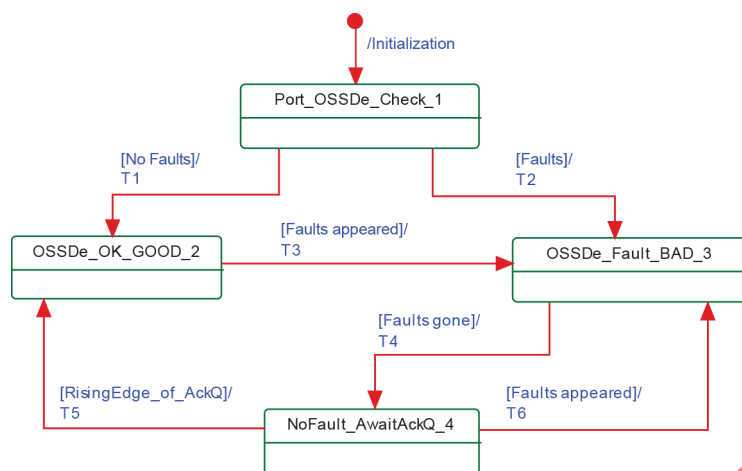


Figure 65 – Qualifier behavior per FS-Master port

Table 43 shows the state and transition table for the qualifier behavior.

Table 43 – State transition Table for the qualifier behavior

STATE NAME	STATE DESCRIPTION		
Initialization	Use SD, Qualifier = BAD, ChFAckReq =0		
1 Port_OSSDe_Check	Perform Port diagnosis to detect Faults		
2 OSSDe_OK_GOOD	Perform Port diagnosis cyclically to detect Faults		
3 OSSDe_Fault_BAD	Perform Port diagnosis cyclically to detect Faults		
4 NoFault_AwaitAckQ	Wait on rising edge of AckQ		

TRAN-SITION	SOURCE STATE	TARGET STATE	ACTION
T1	1	2	Use PD, Qualifier = GOOD, AckQ = 0, ChFAckReq =0
T2	1	3	Use SD, Qualifier = BAD, AckQ = 0, ChFAckReq =0
T3	2	3	Use SD, Qualifier = BAD, AckQ = 0, ChFAckReq =0
T4	3	4	Use SD, Qualifier = BAD, AckQ = 0, ChFAckReq =1
T5	4	2	Use PD, Qualifier = GOOD, AckQ = 1, ChFAckReq =0
T6	4	3	Use SD, Qualifier = BAD, AckQ = 0, ChFAckReq =0

INTERNAL ITEMS	TYPE	DEFINITION
RisingEdge_of_AckQ	Flag	Means to prevent from permanently actuating the AckQ signal.
AckQ	Flag	Flag depending on the individual upper-level FSCP system and its mapping.
Faults	Flag	Any detected fault such as a wire break, short circuit.
ChFAckReq	Flag	Signal set by qualifier handler (see 11.11.2 and Clause H.1)

11.12 SCL diagnosis

The Safety Communication Layer can create its own EventCodes such as CRC error, counter error, or timeout as listed in Clause B.1.

12 Functional safe processing (FS-P)

12.1 Recommendations for efficient I/O mappings

Figure 66 shows how efficiency can be increased when packing I/O data from the connected safety devices into one FSCP SPDU instead of several individual FSCP SPDUs. On the left, the bits of safety devices (OSSD) are packed into one FSCP SPDU by the FS-DI module. On the right, the FS-Devices use each an FSCP SPDU through the FS-Master/Gateway (FS-M/G) to transmit a bit. In the middle, an FS-M/G packs several bits into one FSCP SPDU similar to an FS-DI.

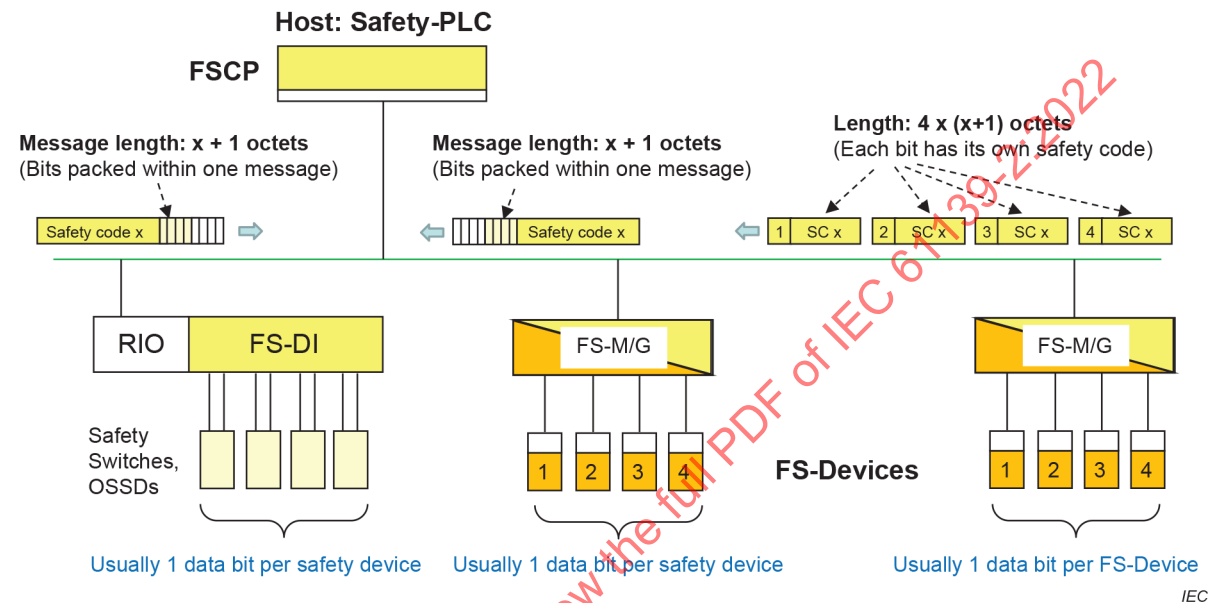


Figure 66 – Mapping efficiency issues

The FS I/O data structure shall be created as a multiple of 16 bits.

12.2 Embedded FS controller

Specification and implementation of an embedded FS controller as described in 4.2 and Figure 1 to provide "local" safety functions on SDCI-FS level are manufacturer's responsibility and not standardized.

Annex A (normative)

Extensions to parameters

A.1 Indices and parameters for SDCI-FS

The Index range reserved for SDCI-FS includes 255 Indices from 0x4200 to 0x42FF.

Table A.1 shows the specified Indices for SDCI profiles, the protocol parameters (FSP) of SDCI-FS, comprising authenticity, protocol, I/O data structure records, and auxiliary parameters as well as the total reserved range for SDCI-FS, and the second range of Indices for SDCI profiles.

Table A.1 – Indices for SDCI-FS

Index (dec)	Sub index	Object name	Access	Length	Data type	M/O/C	Purpose/reference
...							
0x4000 to 0x41FF		Profile specific Indices					For example: Smart sensors
Authenticity (11 octets)							
0x4200 (16896)	1	FSP_Authenticity_1	R/W	4 octets	UIntegerT	M	"A-Code" from the upper-level FSCP system; see A.2.1
	2	FSP_Authenticity_2	R/W	4 octets	UIntegerT	M	Extended "A-Code" from the upper-level FSCP system
	3	FSP_Port	R/W	1 octet	UIntegerT	M	PortNumber identifying the particular FS-Device; see A.2.2
	4	FSP_AuthentCRC	R/W	2 octets	UIntegerT	M	CRC-16 across authenticity parameters; see A.2.3
Protocol (12 octets)							
0x4201 (16897)	1	FSP_ProtVersion	R/W	1 octet	UIntegerT	M	Protocol version: 0x01; see A.2.4
	2	FSP_ProtMode	R/W	1 octet	UIntegerT	M	Protocol modes, e.g. 16/32 bit CRC; see A.2.5
	3	FSP_Watchdog	R/W	2 octets	UIntegerT	M	Monitoring of I/O update; 1 to 65 535 ms; see A.2.6
	4	FSP_IO_StructCRC	R/W	2 octets	UIntegerT	M	CRC-16 signature across I/O structure description block; see A.2.7
	5	FSP_TechParCRC	R/W	4 octets	UIntegerT	M	Securing code across FST (technology specific parameter); see A.2.8
	6	FSP_ProtParCRC	R/W	2 octets	UIntegerT	M	CRC-16 across protocol parameters; see A.2.9
Verification Record (23 octets)							
0x4202 (16898)		FSP_VerifyRecord	W	23 octets	RecordT	M	FS-Master sends this verification record consisting of authenticity and protocol parameters at PREOPERATE. This Index is hidden to the user; see A.2.10

Index (dec)	Sub index	Object name	Access	Length	Data type	M/O/C	Purpose/reference
Auxiliary parameters							
0x4210 (16912)		FSP_TimeToReady	R	2 octets	UIntegerT	M	Time to Ready pulse; 1 to 32 767 (in 10 ms), see A.2.11
0x4211 (16913)		FSP_MinShutDown Time	R	2 octets	UIntegerT	M	Minimum time for the FS-Device to shut down after Port power off; 100 to 1 000 (in 10 ms), see A.2.12
0x4212 (16914)		FSP_ParamDescCRC	R	4 octets	UIntegerT	M	CRC-32 signature securing authenticity, protocol, and FS I/O structure description within IODD; see A.2.15
0x4213 (16915)		FSP_WCDT	R	2 octets	UIntegerT	M	Worst-case delay time; 1 to 32 767 (in ms), see A.2.13 and H.6
0x4214 (16916)		FSP_OFDT	R	2 octets	UIntegerT	M	One fault delay time; 1 to 32 767 (in ms), see A.2.14 and H.6
0x4215 (16917) to 0x42FF (17151)		Reserved for SDCI-FS					
0x4300 to 0x4FFF		Profile specific Indices					For example: BLOB and Firmware update
...							
Key							
M = mandatory; O = optional; C = conditional							

A.2 Parameters in detail

A.2.1 FSP_Authenticity

During off-line commissioning of an SDCI-FS project, the default value of this parameter is "0". During on-line commissioning, the user acquires the FSCP authenticity ("A-Code") from the FS-Master via SMI service and propagates it to the FS-Device within an entire record as described in 10.4.3.1. The FS-Master Tool shall only transfer entire authenticity blocks to the FS-Device with correct CRC signature values such that the FS-Device can check plausibility and correctness (see A.2.3).

In case the system is armed (FSP_TechParCRC ≠ "0") the FS-Device compares at each start-up (DS_Change or PortPowerOffOn) its locally stored values with the values of the FSP_VerifyRecord to detect any misconnection (incorrect Port or FS-Master), see Annex G.

Some FSCPs provide extended authenticity. In those cases, the extended code shall be included in this parameter.

Padding bits and octets shall be filled with "0".

A.2.2 FSP_Port

The FS-Master Tool identifies the FS-Master PortNumber of the attached FS-Device and stores it in this parameter. Storage and checking of the parameter by the FS-Device corresponds to A.2.1 and A.2.3. Numbering starts at "1". Thus, the FS-Device shall not accept a "0".

Default PortNumber in IODD is "0" and means PortNumber of a particular Device has not been assigned yet.

A.2.3 FSP_AuthentCRC

The FS-Master Tool shall only transfer entire authenticity blocks to the FS-Device including FSP_Authenticity and FSP_Port (see Table A.1).

For the CRC signature calculation of the entire authenticity block, the CRC-16 in Table D.1 shall be used. This CRC polynomial has a Hamming distance of ≥ 6 for lengths ≤ 16 octets. A seed value "0" shall be used (see D.3.6).

A.2.4 FSP_ProtVersion

Table A.2 shows the coding of FSP_ProtVersion.

Table A.2 – Coding of protocol version

Value	Definition
0x00	Not permitted
0x01	This protocol version
0x02 to 0xFF	Reserved

A.2.5 FSP_ProtMode

Table A.3 shows the coding of FSP_ProtMode. The "test mirrors" are used by testers.

Table A.3 – Coding of protocol mode

Value	Definition
0x00	Not permitted
0x01	0 to 3 octets of FS I/O Process Data; CRC-16
0x02	0 to 25 octets of FS I/O Process Data; CRC-32
0x03 to 0xF8	Reserved
0xF9	Test mirror in case of CRC-32 (reserved for test)
0xFA	Test mirror in case of CRC-16 (reserved for test)
0xFB to 0xFF	Reserved

A.2.6 FSP_Watchdog

The FS-Device designer determines the I/O update time and uses it as default value of this parameter within the IODD. The I/O update time is the time period between two safety PDUs with subsequent counter values (I/O samples) including possible repetitions within the SDCI communication layer (black channel; see 11.5.5).

With the help of the parameter default value (I/O update time), the transmission times of the safety PDUs, and FS-Master processing times, the FS-Master Tool can estimate the total time and suggest the value of the "FSP_Watchdog" parameter.

The value range is 1 to 65 535 (measured in ms). A value of "0" is not permitted. The SCL of the FS-Device is responsible to check the validity at start-up and to create an error in case (see Table B.1).

A.2.7 FSP_IO_StructCRC

An IODD-based non-safety viewer can be used to calculate this 16-bit CRC signature across the FS I/O structure description within the IODD during the development phase. The algorithm for the calculation is shown in Annex D. A seed value "0" shall be used (see D.3.6).

The safety-related interpreter of the FS-Master Tool transfers the entire instance data together with the CRC signature to the FS I/O data mapper as shown in 10.4.3.1.

Table A.4 shows Version "1" of the generic FS I/O data structure description for FS-Devices. With the help of this table, individual instances of FS-Device I/O Process Data can be created via IODD and, amongst others, used for an automatic mapping of SDCI-FS data to FSCP safety data.

Table A.4 – Generic FS I/O data structure description

Item name	Item length	Definition
IO_DescVersion	1 octet	Version of this generic structure description: 0x01
InputDataRange	1 octet	Length in octets of the entire FS input Process Data including the 4 or 6 octets respectively for the safety code (Control/Status, PortNumber, and CRC-16/32)
TotalOfInBits	1 octet	Number of the entire set of input BooleanT (bits)
TotalOfInOctets	1 octet	Number of octets with input BooleanT (including unfilled octets)
TotalOfInInt16	1 octet	Number of input IntegerT(16)
TotalOfInInt32	1 octet	Number of input IntegerT(32)
OutputDataRange	1 octet	Length in octets of the entire FS output Process Data including the 4 or 6 octets respectively for the safety code (Control/Status, PortNumber, and CRC-16/32)
TotalOfOutBits	1 octet	Number of the entire set of output BooleanT (bits)
TotalOfOutOctets	1 octet	Number of octets with output BooleanT (including unfilled octets)
TotalOfOutInt16	1 octet	Number of output IntegerT(16)
TotalOfOutInt32	1 octet	Number of output IntegerT(32)
FSP_IO_StructCRC	2 octets	CRC-16 signature value across all items (see Clause D.1)

Figure A.1 shows the instance of the FS I/O data description of the example in Figure A.2.

IO_DESCVERSION	01	0x01
INPUT_DATA_RANGE	10	0x0A
TOTAL_OF_INBITS	13	0x0D
TOTAL_OF_INOCTETS	02	0x02
TOTAL_OF_ININT16	01	0x01
TOTAL_OF_ININT32	00	0x00
OUTPUT_DATA_RANGE	06	0x06
TOTAL_OF_OUTBITS	00	0x00
TOTAL_OF_OUTOCTETS	00	0x00
TOTAL_OF_OUTINT16	00	0x00
TOTAL_OF_OUTINT32	00	0x00
FSP_IO_STRUCTCRC	39464	0x9A28

IEC

Figure A.1 – Instance of an FS I/O data description

Figure A.2 shows an example with FS input Process Data and no FS output Process Data.

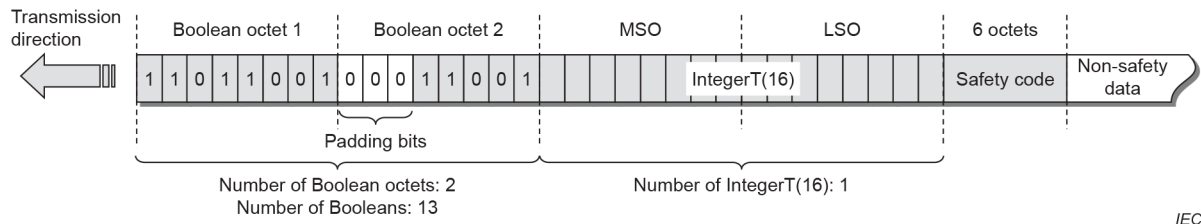


Figure A.2 – Example FS I/O data structure with non-safety data

A.2.8 FSP_TechParCRC

This document specifies two basic methods for the assignment of technology specific parameters (FST). The FS-Device designer is responsible for the selection of the securing method.

The method in 11.7.8 is based on IODD and suggests using one of the CRC generator polynomials in Table D.1. If calculation of the CRC signature value results in "0", a "1" shall be used.

The method in 11.7.9 depends on an existing FS-Device Tool (Dedicated Tool). Whatever method is used, the tool shall display a securing code after verification and validation that can be copied and pasted into the FSP_TechParCRC parameter entry field.

During commissioning a value of "0" can be entered to allow for certain behavior at start-ups of the FS-Device (see 10.4.3.1). During production, this value shall be $\neq$ "0".

For technology specific parameter block transfers > 232 octets, the SMI_PortCmd service CMD = "0" (DeviceParBatch) specified in IEC 61131-9 can be used.

A.2.9 FSP_ProtParCRC

The FS-Master Tool shall only transfer entire protocol blocks to the FS-Device including all protocol parameters (see Table A.1). For the CRC signature calculation of the entire protocol block, the CRC-16 in Table D.1 shall be used. This CRC polynomial has a Hamming distance of ≥ 6 for lengths ≤ 16 octets. A seed value "0" shall be used (see D.3.6).

A.2.10 FSP_VerifyRecord

A record consisting of the authenticity and protocol parameters is transferred via the service "SMI_PortConfiguration" (see 10.2.1 and 10.3.3) and stored within the Configuration Manager of an FS-Master. At start-up during PREOPERATE, the FS-Master forwards this verification record in write only manner to a "hidden" Index in the FS-Device (see 11.7.4). The FS-Device uses this diversely handled record for verification of authenticity, protocol, I/O structure, and technology parameters. This takes place during PREOPERATE after a "DS_Change" (see Figure 35 and IEC 61131-9) whenever an FS-Device has been replaced and parameter have been restored through Data Storage mechanisms. It also takes place after Port power OFF/ON during commissioning through SMI_PortPowerOffOn (see IEC 61131-9:—, 11.2.14 and Clause E.9).

The record shall be transferred as an entity. Subindex access is not permitted. Index 0x4202 (16898) shall be "hidden" to the user; that is, it shall not be described within the IODD.

A.2.11 FSP_TimeToReady

The FS-Device designer measures/determines the time from power-on to the appearance of the Ready pulse (see 5.3.3 and t_{2R} in Table 7) and assigns the value to FSP_TimeToReady in the IODD of the FS-Device (see E.5.8).

NOTE The value is related to the parameter "Time delay before availability" in [9] or [13].

Values greater than 5 s leads to restricted FS-Master behavior, for example no automatic Device detection.

A.2.12 FSP_MinShutDownTime

The FS-Device designer measures/determines the minimum time required to shut down after Port power off prior to a restart and assigns the value to FSP_MinShutDownTime in the IODD of the FS-Device (see E.5.8).

A.2.13 FSP_WCDT

The FS-Device designer measures/determines the "Worst-Case Delay Time" values as defined in H.6. Several different values are possible since FS-Devices can be configured via technology parameters, for example high resolution may lead to longer and low resolution to shorter times. When reading this parameter, the FS-Device will provide a value corresponding to its current parameterization.

A.2.14 FSP_OFDT

The FS-Device designer measures/determines the "One Fault Delay Time" values as defined in Clause H.6. Several different values are possible. When reading this parameter, the FS-Device will provide a value corresponding to its current parameterization.

A.2.15 FSP_ParamDescCRC

The purpose of this parameter is to secure the relevant descriptions of safety parameters within the IODD (see E.5.6) against data falsification as shown in Figure A.3.

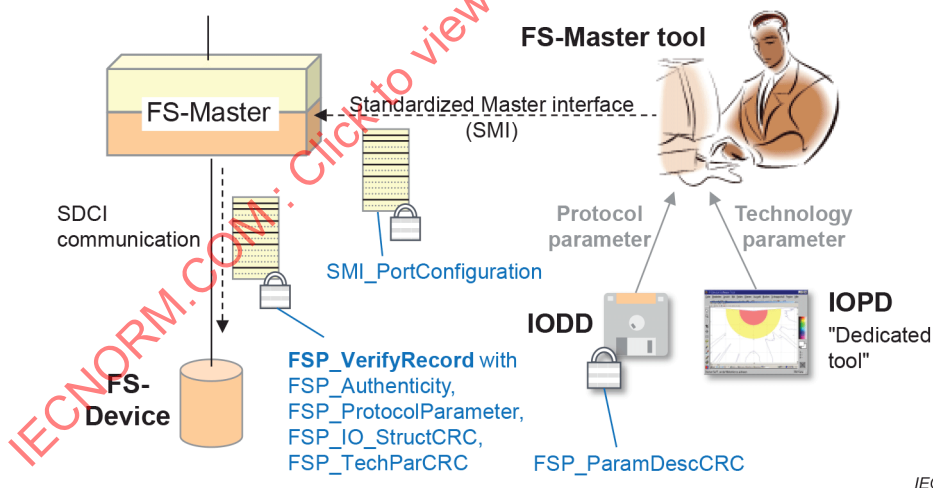


Figure A.3 – Securing of safety parameters

Annex B (normative)

Extensions to EventCodes

B.1 Additional FS-Device EventCodes

The Safety Communication Layer (SCL) within an FS-Device can create its own EventCodes as shown in Table B.1. They are conveyed by the SMI_DeviceEvent service.

Table B.1 – FS-Device SCL specific EventCodes

EventCode	Definition and recommended maintenance action	FS-Device status value	TYPE
0xB000	Transmission error (CRC signature)	2	Notification
0xB001	Transmission error (Counter)	2	Notification
0xB002	Transmission error (Timeout)	3	Error
0xB003	Unexpected authentication code	3	Error
0xB004	Unexpected authentication port	3	Error
0xB005	Incorrect FSP_AuthentCRC	3	Error
0xB006	Incorrect FSP_ProtParCRC	3	Error
0xB007	Incorrect FSP_TechParCRC	3	Error
0xB008	Incorrect FSP_IO_StructCRC	3	Error
0xB009	Watchdog time out of specification (e.g. "0")	3	Error
0xB00A	No FSP_VerifyRecord received (triggered after transition to OPERATE)	3	Error
0xB00B to 0xB0FF	Reserved: do not use number; do not evaluate number	-	-

Usually, "CRC signature" and/or "Counter" transmission errors are caused by seriously falsified SDCI messages with SPDUs due to heavy interferences. There is nothing to repair and an operator acknowledgment is sufficient. This very unlikely warning should inform the operator and the responsible production manager about possible changes within a machine requiring an inspection according to the safety manual (see Clause H.6).

B.2 Additional Port EventCodes

The Safety Communication Layer (SCL) within an FS-Master can create its own EventCodes as shown in Table B.2. They are conveyed by the SMI_PortEvent service (see IEC 61131-9).

Table B.2 – FS-Master SCL specific EventCodes

EventCode	Definition and recommended maintenance action	Status value	TYPE
0x2000	Transmission error (CRC signature)	2	Notification
0x2001	Transmission error (Counter)	2	Notification
0x2002	Transmission error (Timeout)	3	Error
0x2003	Unexpected authentication code	3	Error
0x2004	Reserved		
0x2005	Incorrect FSP_AuthentCRC	3	Error
0x2006	Incorrect FSP_ProtParCRC	3	Error
0x2007 to 0x2008	Reserved		
0x2009	Watchdog time out of specification (e.g. "0")	3	Error
0x200A to 0x20EF	Reserved: do not use number; do not evaluate number	-	-
0x20F0	OSSD locked – signals beyond specified discrepancy times	3	Error
0x20F1 to 0x20FF	Reserved: do not use number; do not evaluate number	-	-

Annex C (normative)

Extensions to Data Types

C.1 Data types for SDCI-FS

Table C.1 shows the available data types in SDCI-FS for cyclic exchange of Process Data for safety functions (see 11.4.9.2).

Table C.1 – Data types for SDCI-FS

Data type	Coding	Length	See IEC 61131-9:—	Device example
BooleanT/bit	BooleanT ("packed form" for efficiency, no WORD structures); assignment of signal names to bits is possible.	1 bit	F.2.2, Table F.22 and Figure F.9	Proximity switch
IntegerT(16)	IntegerT (enumerated or signed)	2 octets	F.2.4, Table F.4, Table F.7 and Figure F.3	Protection fields of laser scanner
IntegerT(32)	IntegerT (enumerated or signed)	4 octets	F.2.4, Table F.4, Table F.6, and Figure F.3	Encoder or length measurement ($\approx \pm 2$ km, resolution 1 μ m)

C.2 BooleanT (bit)

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 C.2.

Table C.2 – BooleanT for SDCI-FS

Data type name	Value range	Resolution	Length
BooleanT	TRUE / FALSE	-	1 bit

SDCI-FS uses solely the so-called packed form via RecordT as shown in Table C.3.

Table C.3 – Example of BooleanT within a RecordT

Subindex	Offset	Data items	Data Type	Name/symbol
1	0	TRUE	BooleanT	Proximity_1
2	1	FALSE	BooleanT	Proximity_2
3	2	FALSE	BooleanT	EmergencyStop_1
4	3	TRUE	BooleanT	EmergencyStop_2
5	4	TRUE	BooleanT	EmergencyStop_3

Figure C.1 demonstrates an example of a BooleanT data structure. Padding bits are "0".

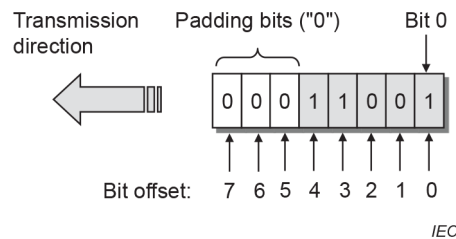


Figure C.1 – Example of a BooleanT data structure

Only RecordT data structures of 8-bit length are permitted. Longer data structures shall use multiple RecordT data structures (see Clause C.5).

NOTE Data structures longer than 8 bit can cause mapping problems with upper-level FSCP systems (see 3.5.2).

C.3 IntegerT (16)

An IntegerT(16) is representing a signed number depicted by 16 bits. The number is accommodated within the octet container 2 and right aligned and extended correctly signed to the chosen number of bits. The data type is specified in Table C.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 C.4 – IntegerT(16)

Data type name	Value range	Resolution	Length
IntegerT(16)	-2^{15} to $2^{15}-1$	1	2 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 C.5).			

The coding of IntegerT(16) is shown in Table C.5.

Table C.5 – IntegerT(16) coding

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	

C.4 IntegerT (32)

An IntegerT(32) is representing a signed number depicted by 32 bits. The number is accommodated within the octet container 4 and right aligned and extended correctly signed to the chosen number of bits. The data type is specified in Table C.6 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 C.6 – IntegerT(32)

Data type name	Value range	Resolution	Length
IntegerT(32)	-2^{31} to $2^{31} - 1$	1	4 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 C.7).			

The coding of IntegerT(32) is shown in Table C.7.

Table C.7 – IntegerT(32) coding

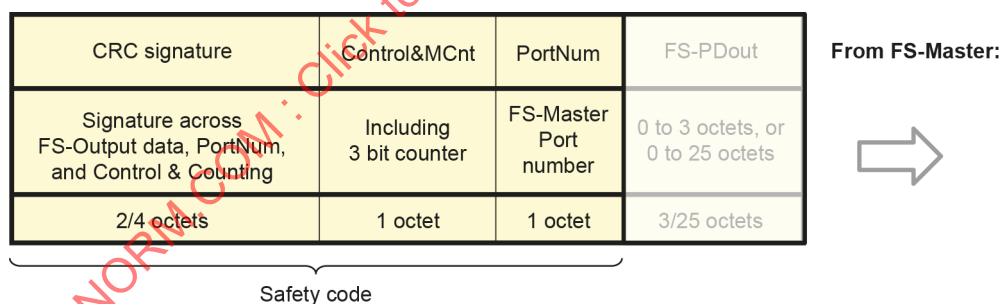
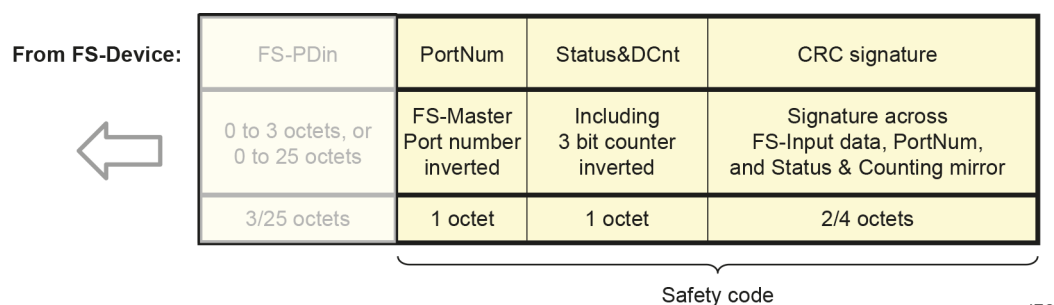
Bit	7	6	5	4	3	2	1	0	Container
Octet 1	SN	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	4 octets
Octet 2	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
Octet 3	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	
Octet 4	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	

C.5 Safety Code

Size of the Safety Code as shown in Figure C.2 and Figure C.3 can be identified by the

- Parameter "FSP_ProtMode" (see Table A.1), and
- FS I/O structure description (see Table A.1).

Thus, the overall I/O data structure can be identified even if there are non-safety related I/O data associated with the SPDU.

**Figure C.2 – Safety Code of an output message****Figure C.3 – Safety Code of an input message**

Annex D (normative)

CRC generator polynomials

D.1 Overview of CRC generator polynomials

Hamming distance and properness for all required data lengths are important characteristics to select a particular generator polynomial.

If the generator polynomial $g(x) = p(x) \cdot (1 + x)$ is used, where $p(x)$ is a primitive polynomial of degree $(r - 1)$, then the maximum total block length is $2^{(r-1)} - 1$, and the code is able to detect single, double, triple and any odd number of errors (see [14]).

If properness is approved, the residual error probability for the approved data length is 2^{-r} .

It shall be prohibited that the CRC generator polynomial used in the underlying transmission systems, for example SDCI, matches the CRC generator polynomial used for SDCI-FS.

Table D.1 shows the CRC-16 and CRC-32 generator polynomials in use for SDCI-FS:

Table D.1 – CRC generator polynomials for SDCI-FS

CRC-16/32 polynomial ("Normal" representation)	Data length (bits)	Hamming distance	Properness	Reference	Remark
0x4EAB	≤ 128	≥ 6	≤ 7 octets	[15]	Suitable for functional safety
0xF4ACFB13	≤ 32768	≥ 6	≤128 octets	[15]	
	≤ 65534	≥ 4			
NOTE Representation: "Normal": high order bit omitted					

The CRC-16 can be used

- to secure cyclic Process Data exchange with a total safety PDU length of up to 7 octets, i.e. 3 octets for safety Process Data and 4 octets of safety code, and
- to secure the transfer of up to 16 octets of FSP parameters at start-up or restart.

The CRC-32 can be used

- to secure cyclic Process Data exchange with a total safety PDU length of up to 31 octets, i.e. 25 octets for safety Process Data (see 11.4.4) and 6 octets of safety code, and
- to secure the transfer and data integrity of the entire FST parameter set.

Additional parameters and assumptions for the calculation of residual error probabilities/rates can be found in 11.4.7.

D.2 Residual error probabilities

Figure D.1 shows the results of residual error probability (REP) calculations over bit error probabilities (BEP) for safety PDU lengths from 3 to 7 octets.

The REP is less than $0,9 \times 10^{-9}$ for BEPs less than the required 10^{-2} at a length of 7 octets.

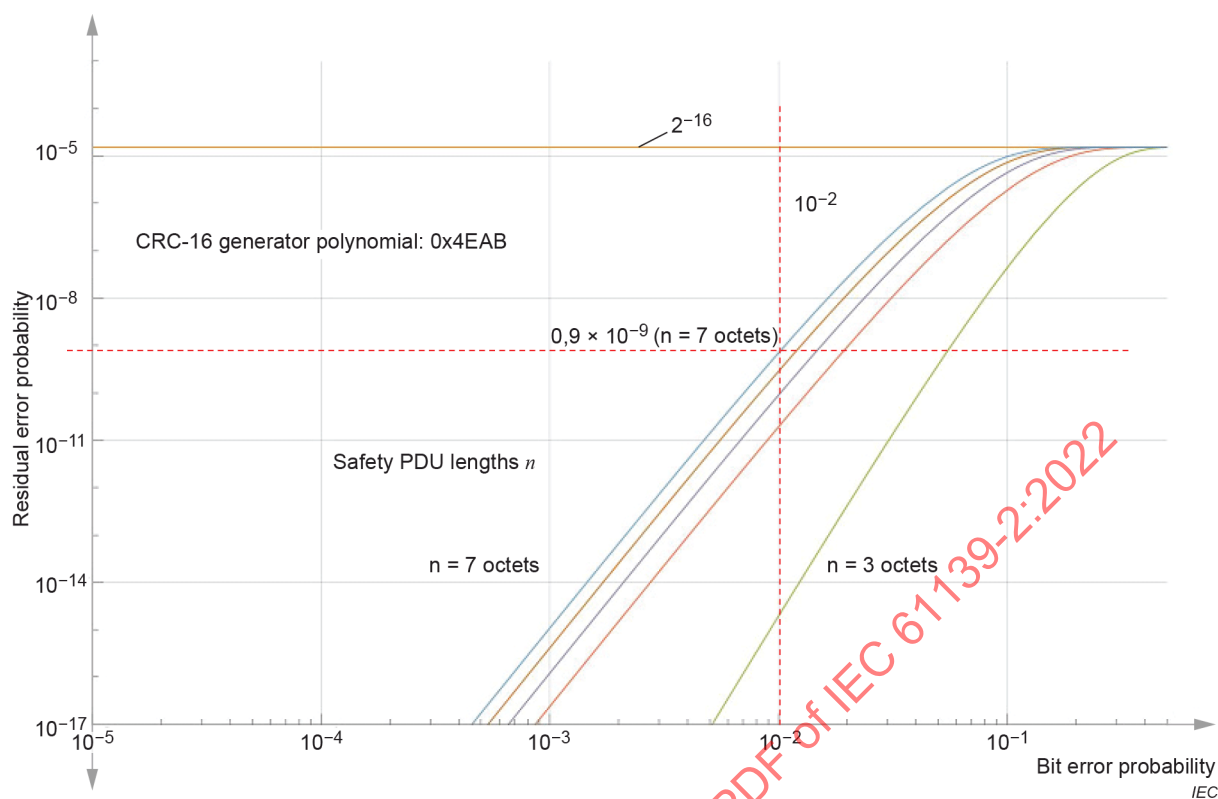


Figure D.1 – CRC-16 generator polynomial

Figure D.2 shows the results of residual error probability (REP) calculations over bit error probabilities (BEP) for safety PDU lengths from 5 to 128 octets.

The REP is less than $0,5 \times 10^{-10}$ for BEPs less than the required 10^{-2} at a length of 26 octets.

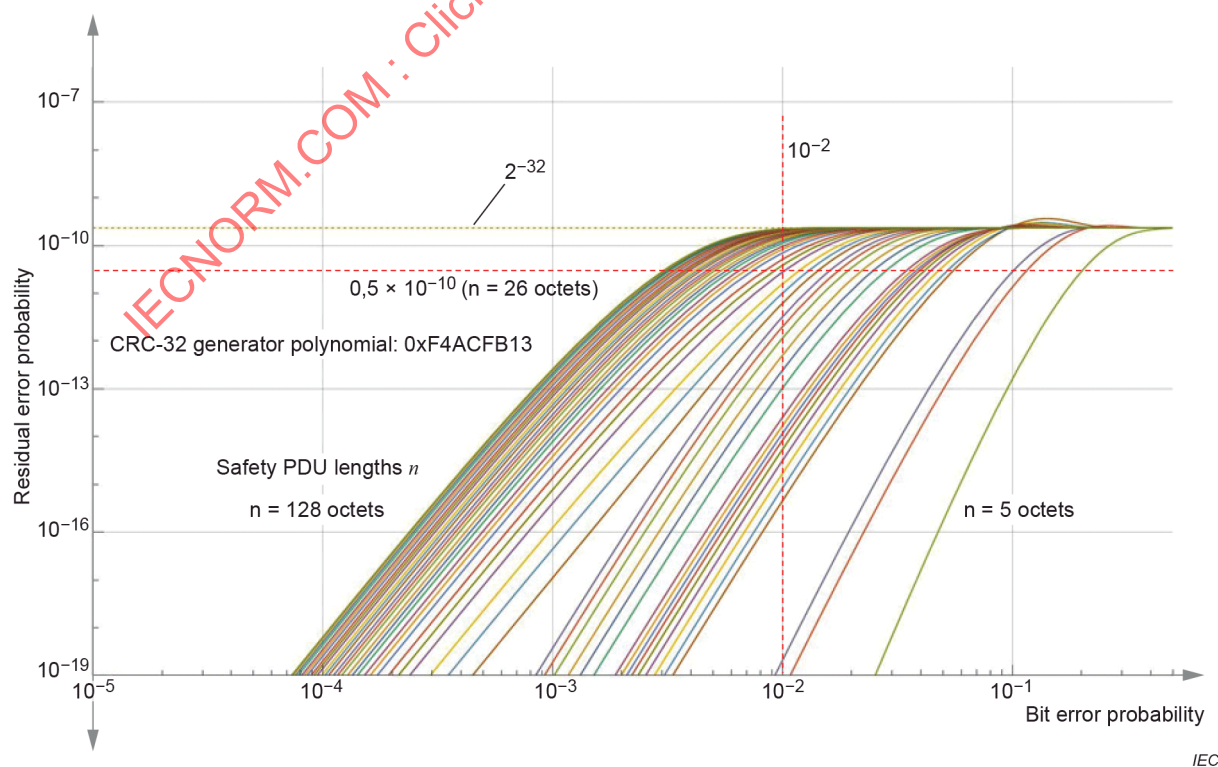


Figure D.2 – CRC-32 generator polynomial

D.3 Implementation considerations

D.3.1 Overview

The designer has two choices to implement the CRC signature calculation. One is based on an algorithm using XOR and shift operations while the other is faster using octet shifts and lookup tables.

D.3.2 Bit shift algorithm (16 bit)

For the 16-bit CRC signature, the value 0x4EAB is used as the generator polynomial. The number of data bits may be odd or even. The value generated after the last octet corresponds to the CRC signature to be transferred.

Figure D.3 shows the algorithm for the innermost loop in "C" programming language.

```
void crc16_calc(unsigned char x, unsigned long *r)
{
    int i;
    for (i = 1; i <= 8; i++)
        if ((bool)(*r & 0x8000) != (bool)(x & 0x80))
            /* XOR = 1 → shift and process polynomial */
            *r = (*r << 1) ^ 0x4EAB;
        else
            /* XOR = 0 → shift only */
            *r = *r << 1;
        x = x << 1;
    /* for */
}
```

IEC

Figure D.3 – Bit shift algorithm in "C" language (16 bit)

The variables used in Figure D.3 are specified in Table D.2.

Table D.2 – Definition of variables used in Figure D.3

Variable	Definition
x	Data bits including 16-bit CRC signature with "0"
*r	Dereferenced pointer to CRC signature
i	Bitcount 1 to 8

D.3.3 Lookup table (16 bit)

The corresponding function in "C" language is shown in Figure D.4. This function is faster. However, the lookup table requires memory space.

```
r = crctab16 [((r >> 8) ^ *q++) & 0xff] ^ (r << 8)
```

IEC

Figure D.4 – CRC-16 signature calculation using a lookup table

The variables used in Figure D.4 are specified in Table D.3.

Table D.3 – Definition of variables used in Figure D.4

Variable	Definition
r	CRC signature
q	q represents the pointer to the actual octet value requiring CRC calculation. After reading the value this pointer shall be incremented for the next octet via q++.

The function in Figure D.4 uses the lookup in Table D.4.

Table D.4 – Lookup Table for CRC-16 signature calculation

CRC-16 lookup table (0 to 255)							
0000	4EAB	9D56	D3FD	7407	3AAC	E951	A7FA
E80E	A6A5	7558	3BF3	9C09	D2A2	015F	4FF4
9EB7	D01C	03E1	4D4A	EAB0	A41B	77E6	394D
76B9	3812	EBEF	A544	02BE	4C15	9FE8	D143
73C5	3D6E	EE93	A038	07C2	4969	9A94	D43F
9BCB	D560	069D	4836	EFCC	A167	729A	3C31
ED72	A3D9	7024	3E8F	9975	D7DE	0423	4A88
057C	4BD7	982A	D681	717B	3FD0	EC2D	A286
E78A	A921	7ADC	3477	938D	DD26	0EDB	4070
0F84	412F	92D2	DC79	7B83	3528	E6D5	A87E
793D	3796	E46B	AAC0	0D3A	4391	906C	DEC7
9133	DF98	0C65	42CE	E534	AB9F	7862	36C9
944F	DAE4	0919	47B2	E048	AEE3	7D1E	33B5
7C41	32EA	E117	AFBC	0846	46ED	9510	DBBB
0AF8	4453	97AE	D905	7EFF	3054	E3A9	AD02
E2F6	AC5D	7FA0	310B	96F1	D85A	0BA7	450C
81BF	CF14	1CE9	5242	F5B8	BB13	68EE	2645
69B1	271A	F4E7	BA4C	1DB6	531D	80E0	CE4B
1F08	51A3	825E	CCF5	6B0F	25A4	F659	B8F2
F706	B9AD	6A50	24FB	8301	CDAA	1E57	50FC
F27A	BCD1	6F2C	2187	867D	C8D6	1B2B	5580
1A74	54DF	8722	C989	6E73	20D8	F325	BD8E
6CCD	2266	F19B	BF30	18CA	5661	859C	CB37
84C3	CA68	1995	573E	F0C4	BE6F	6D92	2339
6635	289E	FB63	B5C8	1232	5C99	8F64	C1CF
8E3B	C090	136D	5DC6	FA3C	B497	676A	29C1
F882	B629	65D4	2B7F	8C85	C22E	11D3	5F78
108C	5E27	8DDA	C371	648B	2A20	F9DD	B776
15F0	5B5B	88A6	C60D	61F7	2F5C	FCA1	B20A
FDFE	B355	60A8	2E03	89F9	C752	14AF	5A04
8B47	C5EC	1611	58BA	FF40	B1EB	6216	2CBD
6349	2DE2	FE1F	B0B4	174E	59E5	8A18	C4B3

NOTE This table contains 16-bit values in hexadecimal representation for each value (0 to 255) of the argument a in the function crctab16 [a]. The table should be used in ascending order from top left (0) to bottom right (255).

D.3.4 Bit shift algorithm (32 bit)

For the 32-bit CRC signature, the value 0xF4ACFB13 is used as the generator polynomial. The number of data bits may be odd or even. The value generated after the last octet corresponds to the CRC signature to be transferred.

Figure D.5 shows the algorithm for the innermost loop in "C" programming language.

```
void crc32_calc(unsigned char x, unsigned long *r)
{
    int i;
    for (i = 1; i <= 8; i++)
        if ((bool)(*r & 0x80000000) != (bool)(x & 0x80))
            /* XOR = 1 → shift and process polynomial */
            *r = (*r << 1) ^ 0xF4ACFB13;
        else
            /* XOR = 0 → shift only */
            *r = *r << 1;
        x = x << 1;
    /* for */
}
```

IEC

Figure D.5 – Bit shift algorithm in "C" language (32 bit)

The variables used in Figure D.5 are specified in Table D.5.

Table D.5 – Definition of variables used in Figure D.5

Variable	Definition
x	Data bits including 32-bit CRC signature with "0"
*r	Dereferenced pointer to CRC signature
i	Bit count 1 to 8

D.3.5 Lookup table (32 bit)

The corresponding function in "C" language is shown in Figure D.6. This function is faster. However, the lookup table requires memory space.

```
r = crctab32 [((r >> 24) ^ *q++) & 0xff] ^ (r << 8)
```

IEC

Figure D.6 – CRC-32 signature calculation using a lookup table

The variables used in Figure D.6 are specified in Table D.6.

Table D.6 – Definition of variables used in Figure D.4

Variable	Definition
r	CRC signature
q	q represents the pointer to the actual octet value requiring CRC calculation. After reading the value this pointer shall be incremented for the next octet via q++.

The function in Figure D.6 uses the lookup table in Table D.7.

Table D.7 – Lookup Table for CRC-32 signature calculation

CRC-32 lookup table (0 to 255)							
00000000	F4ACFB13	1DF50D35	E959F626	3BEA1A6A	CF46E179	261F175F	D2B3EC4C
77D434D4	8378CFC7	6A2139E1	9E8DC2F2	4C3E2EBE	B892D5AD	51CB238B	A567D898
EFA869A8	1B0492BB	F25D649D	06F19F8E	D44273C2	20EE88D1	C9B77EF7	3D1B85E4
987C5D7C	6CD0A66F	85895049	7125AB5A	A3964716	573ABC05	BE634A23	4ACFB130
2BFC2843	DF50D350	36092576	C2A5DE65	10163229	E4BAC93A	0DE33F1C	F94FC40F
5C281C97	A884E784	41DD11A2	B571EAB1	67C206FD	936EFDEE	7A370BC8	8E9BF0DB
C45441EB	30F8BAF8	D9A14CDE	2D0DB7CD	FFBE5B81	0B12A092	E24B56B4	16E7ADA7
B380753F	472C8E2C	AE75780A	5AD98319	886A6F55	7CC69446	959F6260	61339973
57F85086	A354AB95	4A0D5DB3	BEA1A6A0	6C124AEC	98BEB1FF	71E747D9	854BBCCA
202C6452	D4809F41	3DD96967	C9759274	1BC67E38	EF6A852B	0633730D	F29F881E
B850392E	4CFCC23D	A5A5341B	5109CF08	83BA2344	7716D857	9E4F2E71	6AE3D562
CF840DFA	3B28F6E9	D27100CF	26DDFBDC	F46E1790	00C2EC83	E99B1AA5	1D37E1B6
7C0478C5	88A883D6	61F175F0	955D8EE3	47EE62AF	B34299BC	5A1B6F9A	AEB79489
0BD04C11	FF7CB702	16254124	E289BA37	303A567B	C496AD68	2DCF5B4E	D963A05D
93AC116D	6700EA7E	8E591C58	7AF5E74B	A8460B07	5CEAF014	B5B30632	411FFD21
E47825B9	10D4DEAA	F98D288C	0D21D39F	DF923FD3	2B3EC4C0	C26732E6	36CBC9F5
AFF0A10C	5B5C5A1F	B205AC39	46A9572A	941ABB66	60B64075	89EFB653	7D434D40
D82495D8	2C886ECB	C5D198ED	317D63FE	E3CE8FB2	176274A1	FE3B8287	0A977994
4058C8A4	B4F433B7	5DADC591	A9013E82	7BB2D2CE	8F1E29DD	6647DFFB	92EB24E8
378CFC70	C3200763	2A79F145	DED50A56	0C66E61A	F8CA1D09	1193EB2F	E53F103C
840C894F	70A0725C	99F9847A	6D557F69	BFE69325	4B4A6836	A2139E10	56BF6503
F3D8BD9B	07744688	EE2DB0AE	1A814BBB	C832A7F1	3C9E5CE2	D5C7AAC4	216B51D7
6BA4E0E7	9F081BF4	7651EDD2	82FD16C1	504EFA8D	A4E2019E	4DBBF7B8	B9170CAB
1C70D433	E8DC2F20	0185D906	F5292215	279ACE59	D336354A	3A6FC36C	CEC3387F
F808F18A	0CA40A99	E5FDFCBF	115107AC	C3E2EBE0	374E10F3	DE17E6D5	2ABB1DC6
8FDCC55E	7B703E4D	9229C86B	66853378	B436DF34	409A2427	A9C3D201	5D6F2912
17A09822	E30C6331	0A559517	FEF96E04	2C4A8248	D8E6795B	31BF8F7D	C513746E
6074ACF6	94D857E5	7D81A1C3	892D5AD0	5B9EB69C	AF324D8F	466BBBA9	B2C740BA
D3F4D9C9	275822DA	CE01D4FC	3AAD2FEF	E81EC3A3	1CB238B0	F5EBCE96	01473585
A420ED1D	508C160E	B9D5E028	4D791B3B	9FCAF777	6B660C64	823FFA42	76930151
3C5CB061	C8F04B72	21A9BD54	D5054647	07B6AA0B	F31A5118	1A43A73E	EEEE5C2D
4B8884B5	BF247FA6	567D8980	A2D17293	70629EDF	84CE65CC	6D9793EA	993B68F9

NOTE This table contains 32-bit values in hexadecimal representation for each value (0 to 255) of the argument a in the function `crctab32 [a]`. The table should be used in ascending order from top left (0) to bottom right (255).

D.3.6 Seed values

The algorithm for example in Figure D.3 does not mention explicitly any initial value for the CRC signature variable in "`*r`". It is implicitly assumed to be "0" by default. This initial value is sometimes called "seed value" in literature.

In 11.4.7, a seed value of "1" is required for the cyclic data exchange of safety PDUs. The reason for that is the possibility for the FS-PDout or FS-PDin data to become completely "0". Since it is a property of CRC-signatures for leading zeros in data strings not to be secured by CRC signatures whenever the seed value is "0", the requirement in 11.4.6 is justified. Any value instead of "0" could be used. However, a "1" is sufficient and faster since all the operations then are shifting and only the last one consists of shifting and XOR processing.

In A.2.3, A.2.9, A.2.7, E.5.1 and E.5.6, the seed value can be "0" since there are no leading zeros within the data strings.

Publicly available CRC signature calculators can be found in [16].

D.3.7 Octet order for CRC calculation

The order of octets for the CRC calculation of the SPDU shall start with the seed value followed by all other octets in the transmission order (see 3.5.2, Figure C.2, and Figure C.3).

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

Annex E (normative)

IODD extensions

E.1 General

The IODD of FS-Devices requires extensions for particular FSP parameters and a securing mechanism to protect the content of IODD files from being falsified as mentioned in 11.7.1.

In addition, some of the parameters specified in IEC 61131-9 shall be mandatory instead of optional for this SDCI extension/profile (see Clause E.3).

E.2 Schema

There are no extensions required to the basic IODD schema.

NOTE The IODD schema is outside the scope of IEC 61131-9. It is described in [17].

E.3 IODD constraints

E.3.1 General rules

Basis for the IODD in SDCI-FS are the definitions of the Common Profile in [18].

All parameters refer to IEC 61131-9. As a general rule, all parameters with Read/Write (R/W) access shall provide a default value within the IODD (for FSP parameters see E.5.2).

E.3.2 Description of the IODD structure

The structure of the IODD of an FS-Device is defined within the XML snippet file "IOL-SafetyProfile-Snippets1.1.3.xml". The following XML code shows the content of this file:

```
<?xml version="1.0" encoding="UTF-8"?>
<IODDProfileDefinitions xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.io-link.com/IODD/2010/10">
  <DocumentInfo version="V1.1.3" releaseDate="2022-01-21" copyright="Copyright 2022, IO-Link
  Community"/>
  <SupportedProfiles profileCharacteristic="16385" profileClassName="Safety"
  profilePrefixes="FSP, FST" reservedIndexRange="16896..16898, 16912, 16914..16916"
  reservedEvents="45056..45066" requiredProfile="16384">
    <ProfileVariant id="PR_FSP" profileId="16385" name="IO-Link Safety"/>
  </SupportedProfiles>
  <!--
    IODD snippet content for the IO-Link Safety Profile Extension.
    The IODD Checker will verify the IODD according to these rules.
  -->
  <DatatypeCollection>
    <Datatype id="D_FSP_Authenticity" xsi:type="RecordT" bitLength="88"
    subindexAccessSupported="false">
      <RecordItem subindex="1" bitOffset="56">
        <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
        <Name textId="TN_V_FSP_Authenticity_1"/>
        <Description textId="TD_V_FSP_Authenticity_1"/>
      </RecordItem>
      <RecordItem subindex="2" bitOffset="24">
        <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
        <Name textId="TN_V_FSP_Authenticity_2"/>
        <Description textId="TD_V_FSP_Authenticity_2"/>
      </RecordItem>
      <RecordItem subindex="3" bitOffset="16">
        <SimpleDatatype xsi:type="UIntegerT" bitLength="8"/>
        <Name textId="TN_V_FSP_Port"/>
      </RecordItem>
    </Datatype>
  </DatatypeCollection>
```

```

    <Description textId="TD_V_FSP_Port"/>
  </RecordItem>
  <RecordItem subindex="4" bitOffset="0">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="16"/>
    <Name textId="TN_V_FSP_AuthentCRC"/>
    <Description textId="TD_V_FSP_AuthentCRC"/>
  </RecordItem>
</Datatype>
<Datatype id="D_FSP_Protocol" xsi:type="RecordT" bitLength="96"
subindexAccessSupported="false">
  <RecordItem subindex="1" bitOffset="88">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="8">
      <SingleValue value="1"/>
    </SimpleDatatype>
    <Name textId="TN_V_FSP_ProtVersion"/>
    <Description textId="TD_V_FSP_ProtVersion"/>
  </RecordItem>
  <RecordItem subindex="2" bitOffset="80">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="8">
      <!-- Which of these SingleValues is supported is device specific. Only one shall be
referenced. -->
      <SingleValue value="1" checkElement="minOccurs 0">
        <Name textId="TN_SV_FSP_ProtMode_crc16"/>
      </SingleValue>
      <SingleValue value="2" checkElement="minOccurs 0">
        <Name textId="TN_SV_FSP_ProtMode_crc32"/>
      </SingleValue>
    </SimpleDatatype>
    <Name textId="TN_V_FSP_ProtMode"/>
    <Description textId="TD_V_FSP_ProtMode"/>
  </RecordItem>
  <RecordItem subindex="3" bitOffset="64">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="16">
      <!-- Which ValueRange is supported is device specific (but the lowerValue must be >0) -->
      <ValueRange lowerValue="#tbd 1..65535 #" upperValue="#tbd 2..65535 #"/>
    </SimpleDatatype>
    <Name textId="TN_V_FSP_Watchdog"/>
    <Description textId="TD_V_FSP_Watchdog"/>
  </RecordItem>
  <RecordItem subindex="4" bitOffset="48">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="16"/>
    <Name textId="TN_V_FSP_IO_StructCRC"/>
    <Description textId="TD_V_FSP_IO_StructCRC"/>
  </RecordItem>
  <RecordItem subindex="5" bitOffset="16">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
    <Name textId="TN_V_FSP_TechParCRC"/>
    <Description textId="TD_V_FSP_TechParCRC"/>
  </RecordItem>
  <RecordItem subindex="6" bitOffset="0">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="16"/>
    <Name textId="TN_V_FSP_ProtParCRC"/>
    <Description textId="TD_V_FSP_ProtParCRC"/>
  </RecordItem>
</Datatype>
<Datatype id="D_FSP_PDin" xsi:type="RecordT" bitLength="#tbd#"
subindexAccessSupported="false">
  <!-- Technology specific safety process data have subindex 1..126 -->
  <!-- Datatype order for subindices is: boolean, integer16, integer32 -->
  <!-- Boolean array -->
  <!-- There may be no gaps between the booleans, but the last octet may contain less than
eight booleans. -->
  <RecordItem subindex="#tbd 1..126 #" bitOffset="#tbd#" checkElement="minOccurs 0">
    <SimpleDatatype xsi:type="BooleanT">
      <SingleValue value="false" checkElement="minOccurs 0">
        <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
      </SingleValue>
      <SingleValue value="true" checkElement="minOccurs 0">
        <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
      </SingleValue>
    </SimpleDatatype>
    <Name textId="TN_V_FST_" checkAttributes="startsWith textId"/>
    <Description textId="TD_V_FST_" checkAttributes="startsWith textId"/>
  </RecordItem>
  <!-- Integer16 values -->
  <!-- Integer values always are aligned to byte borders. -->
  <RecordItem subindex="#tbd 1..126 #" bitOffset="#tbd#" checkElement="minOccurs 0">
    <SimpleDatatype xsi:type="IntegerT" bitLength="16">

```

```

    <SingleValue value="#tbd#" checkElement="minOccurs 0">
      <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
    </SingleValue>
    <ValueRange lowerValue="#tbd#" upperValue="#tbd#" checkElement="minOccurs 0"/>
  </SimpleDatatype>
  <Name textId="TN_V_FST_" checkAttributes="startsWith textId"/>
  <Description textId="TD_V_FST_" checkAttributes="startsWith textId"/>
</RecordItem>
<!-- Integer32 values -->
<!-- Integer values always are aligned to byte borders. -->
<RecordItem subindex="#tbd 1..126 #" bitOffset="#tbd#" checkElement="minOccurs 0">
  <SimpleDatatype xsi:type="IntegerT" bitLength="32">
    <SingleValue value="#tbd#" checkElement="minOccurs 0">
      <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
    </SingleValue>
    <ValueRange lowerValue="#tbd#" upperValue="#tbd#" checkElement="minOccurs 0"/>
  </SimpleDatatype>
  <Name textId="TN_V_FST_" checkAttributes="startsWith textId"/>
  <Description textId="TD_V_FST_" checkAttributes="startsWith textId"/>
</RecordItem>
<!-- Safety Code has fixed subindex 127 -->
<RecordItem subindex="127" bitOffset="#tbd#">
  <SimpleDatatype xsi:type="OctetStringT" fixedLength="#tbd 4, 6 #"/>
  <Name textId="TN_V_FSP_SafetyCode"/>
  <Description textId="TD_V_FSP_SafetyCode"/>
</RecordItem>
<!-- Non-safety process data has subindex 128..255 -->
<RecordItem subindex="#tbd 128..255 #" bitOffset="#tbd#" checkElement="minOccurs 0">
  <DatatypeRef datatypeId="#tbd#" checkElement="minOccurs 0"/>
  <SimpleDatatype xsi:type="tbd" checkElement="minOccurs 0" checkAttributes="option bitLength,
option fixedLength">
    <SingleValue value="false" checkElement="minOccurs 0">
      <Name textId="TN_SV_" checkAttributes="startsWith textId"/>
    </SingleValue>
    <ValueRange lowerValue="#tbd#" upperValue="#tbd#" checkElement="minOccurs 0"/>
  </SimpleDatatype>
  <Name textId="#tbd#" />
  <Description textId="#tbd#" />
</RecordItem>
</Datatype>
<Datatype id="D_FSP_PDout" xsi:type="RecordT" bitLength="#tbd#"
subindexAccessSupported="false">
  <!-- Technology specific safety process data have subindex 1..126 -->
  <!-- Datatype order for subindexes is: boolean, integer16, integer32 -->
  <!-- Boolean array -->
  <!-- There may be no gaps between the booleans, but the last octet may contain less than
eight booleans. -->
  <RecordItem subindex="#tbd 1..126 #" bitOffset="#tbd#" checkElement="minOccurs 0">
    <SimpleDatatype xsi:type="BooleanT">
      <SingleValue value="false" checkElement="minOccurs 0">
        <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
      </SingleValue>
      <SingleValue value="true" checkElement="minOccurs 0">
        <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
      </SingleValue>
    </SimpleDatatype>
    <Name textId="TN_V_FST_" checkAttributes="startsWith textId"/>
    <Description textId="TD_V_FST_" checkAttributes="startsWith textId"/>
  </RecordItem>
  <!-- Integer16 values -->
  <!-- Integer values always are aligned to byte borders. -->
  <RecordItem subindex="#tbd 1..126 #" bitOffset="#tbd#" checkElement="minOccurs 0">
    <SimpleDatatype xsi:type="IntegerT" bitLength="16">
      <SingleValue value="#tbd#" checkElement="minOccurs 0">
        <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
      </SingleValue>
      <ValueRange lowerValue="#tbd#" upperValue="#tbd#" checkElement="minOccurs 0"/>
    </SimpleDatatype>
    <Name textId="TN_V_FST_" checkAttributes="startsWith textId"/>
    <Description textId="TD_V_FST_" checkAttributes="startsWith textId"/>
  </RecordItem>
  <!-- Integer32 values -->
  <!-- Integer values always are aligned to byte borders. -->
  <RecordItem subindex="#tbd 1..126 #" bitOffset="#tbd#" checkElement="minOccurs 0">
    <SimpleDatatype xsi:type="IntegerT" bitLength="32">
      <SingleValue value="#tbd#" checkElement="minOccurs 0">
        <Name textId="TN_SV_FST_" checkAttributes="startsWith textId"/>
      </SingleValue>

```

```

    <ValueRange lowerValue="#tbd#" upperValue="#tbd#" checkElement="minOccurs 0"/>
  </SimpleDatatype>
  <Name textId="TN_V_FST_" checkAttributes="startsWith textId"/>
  <Description textId="TD_V_FST_" checkAttributes="startsWith textId"/>
</RecordItem>
<!-- Safety Code has fixed subindex 127 -->
<RecordItem subindex="127" bitOffset="#tbd#" >
  <SimpleDatatype xsi:type="OctetStringT" fixedLength="6"/>
  <Name textId="TN_V_FSP_SafetyCode"/>
  <Description textId="TD_V_FSP_SafetyCode"/>
</RecordItem>
<!-- Non-safety process data has subindex 128..255 -->
<RecordItem subindex="#tbd 128..255 #" bitOffset="#tbd#" checkElement="minOccurs 0">
  <DatatypeRef datatypeId="#tbd#" checkElement="minOccurs 0"/>
  <SimpleDatatype xsi:type="tbd" checkElement="minOccurs 0" checkAttributes="option bitLength,
option fixedLength">
    <SingleValue value="false" checkElement="minOccurs 0">
      <Name textId="TN_SV_" checkAttributes="startsWith textId"/>
    </SingleValue>
    <ValueRange lowerValue="#tbd#" upperValue="#tbd#" checkElement="minOccurs 0"/>
  </SimpleDatatype>
  <Name textId="#tbd#" />
  <Description textId="#tbd#" />
</RecordItem>
</Datatype>
</DatatypeCollection>
<VariableCollection checkElement="atLeast">
  <StdVariableRef id="V_SystemCommand" checkElement="minOccurs 1">
    <StdSingleValueRef value="128" checkElement="maxOccurs 0"/>
    <StdSingleValueRef value="130" checkElement="maxOccurs 0"/>
  </StdVariableRef>
  <StdVariableRef id="V_ProcessDataInput" checkElement="minOccurs 1"/>
  <StdVariableRef id="V_ProcessDataOutput" checkElement="minOccurs 1"/>
  <Variable index="16896" id="V_FSP_Authenticity" accessRights="rw" profileConstraints="PR_FSP">
    <DatatypeRef datatypeId="D_FSP_Authenticity"/>
    <RecordItemInfo subindex="1" defaultValue="0"/>
    <RecordItemInfo subindex="2" defaultValue="0"/>
    <RecordItemInfo subindex="3" defaultValue="0"/>
    <RecordItemInfo subindex="4" defaultValue="0"/>
    <Name textId="TN_V_FSP_Authenticity"/>
    <Description textId="TD_V_FSP_Authenticity"/>
  </Variable>
  <Variable index="16897" id="V_FSP_Protocol" accessRights="rw" profileConstraints="PR_FSP">
    <DatatypeRef datatypeId="D_FSP_Protocol"/>
    <!-- FSP_ProtVersion: 1= valid -->
    <RecordItemInfo subindex="1" defaultValue="1"/>
    <!-- FSP_ProtMode: 1 (16 bit CRC) or 2 (32 bit CRC)= valid -->
    <RecordItemInfo subindex="2" defaultValue="#tbd 1 2 #"/>
    <!-- FSP_Watchdog: 1 .. 65535 = valid -->
    <RecordItemInfo subindex="3" defaultValue="#tbd 1..65535 #"/>
    <!-- FSP_IO_StructCRC: = valid -->
    <RecordItemInfo subindex="4" defaultValue="#tbd 1..65535 #"/>
    <!-- FSP_TechParCRC: 0= invalid -->
    <RecordItemInfo subindex="5" defaultValue="0"/>
    <!-- FSP_ParamParCRC: 0= invalid -->
    <RecordItemInfo subindex="6" defaultValue="0"/>
    <Name textId="TN_V_FSP_Protocol"/>
    <Description textId="TD_V_FSP_Protocol"/>
  </Variable>
  <!-- Note: Variable FSP_VerifyRecord (index="16898") shall not be described in the IODD. -->
  <Variable id="V_FSP_TimeToReady" index="16912" accessRights="ro" defaultValue="#tbd 1..32767
# " profileConstraints="PR_FSP">
    <!-- resolution 10 ms -->
    <Datatype xsi:type="UIntegerT" bitLength="16"/>
    <Name textId="TN_V_FSP_TimeToReady"/>
    <Description textId="TD_V_FSP_TimeToReady"/>
  </Variable>
  <Variable id="V_FSP_MinShutDownTime" index="16913" accessRights="ro" defaultValue="#tbd
100..1000 # " profileConstraints="PR_FSP">
    <!-- resolution 10 ms -->
    <Datatype xsi:type="UIntegerT" bitLength="16"/>
    <Name textId="TN_V_FSP_MinShutDownTime"/>
    <Description textId="TD_V_FSP_MinShutDownTime"/>
  </Variable>
  <Variable id="V_FSP_ParamDescCRC" index="16914" accessRights="ro" defaultValue="#tbd#"
profileConstraints="PR_FSP">
    <Datatype xsi:type="UIntegerT" bitLength="32"/>

```



```
<Name textId="TN_V_FSP_ParamDescCRC"/>  
<Description textId="TD_V_FSP_ParamDescCRC"/>  
</Variable>  
<Variable id="V_FSP_WCDT" index="16915" accessRights="ro" defaultValue="#tbd 1..32767 #" profileConstraints="PR_FSP">  
  <!-- in ms -->  
  <Datatype xsi:type="UIntegerT" bitLength="16"/>  
  <Name textId="TN_V_FSP_WCDT"/>  
  <Description textId="TD_V_FSP_WCDT"/>  
</Variable>  
<Variable id="V_FSP_OFDT" index="16916" accessRights="ro" defaultValue="#tbd 1..32767 #" profileConstraints="PR_FSP">  
  <!-- in ms -->  
  <Datatype xsi:type="UIntegerT" bitLength="16"/>  
  <Name textId="TN_V_FSP_OFDT"/>  
  <Description textId="TD_V_FSP_OFDT"/>  
</Variable>  
</ValueCollection>  
<ProcessDataCollection>  
  <ProcessData id="P_" checkElement="minOccurs 1" checkAttributes="startsWith id">  
    <Condition variableId="#tbd#" subindex="#tbd#" value="#tbd#" checkElement="minOccurs 0" checkAttributes="option subindex"/>  
    <ProcessDataIn id="PI_PDin" bitLength="#tbd 32..256 #" checkAttributes="startsWith id">  
      <DatatypeRef datatypeId="D_FSP_PDin" checkAttributes="startsWith datatypeid"/>  
      <Name textId="TN_PI_PDin" checkAttributes="startsWith textid"/>  
    </ProcessDataIn>  
    <ProcessDataOut id="PO_PDout" bitLength="#tbd 32..256 #" checkAttributes="startsWith id">  
      <DatatypeRef datatypeId="D_FSP_PDout" checkAttributes="startsWith datatypeid"/>  
      <Name textId="TN_PO_PDout" checkAttributes="startsWith textid"/>  
    </ProcessDataOut>  
  </ProcessData>  
</ProcessDataCollection>  
<EventCollection checkElement="atLeast">  
  <Event code="45056" type="Warning">  
    <Name textId="TN_EV_FSP_TransmissionError_CRCSTransmissionSignature"/>  
    <Description textId="TD_EV_FSP_TransmissionError_CRCSTransmissionSignature"/>  
  </Event>  
  <Event code="45057" type="Warning">  
    <Name textId="TN_EV_FSP_TransmissionError_CounterCounter"/>  
    <Description textId="TD_EV_FSP_TransmissionError_CounterCounter"/>  
  </Event>  
  <Event code="45058" type="Error">  
    <Name textId="TN_EV_FSP_TransmissionError_TimeoutTimeout"/>  
    <Description textId="TD_EV_FSP_TransmissionError_TimeoutTimeout"/>  
  </Event>  
  <Event code="45059" type="Error">  
    <Name textId="TN_EV_FSP_UnexpectedAuthenticationCodeCode"/>  
    <Description textId="TD_EV_FSP_UnexpectedAuthenticationCodeCode"/>  
  </Event>  
  <Event code="45060" type="Error">  
    <Name textId="TN_EV_FSP_UnexpectedAuthenticationPortPort"/>  
    <Description textId="TD_EV_FSP_UnexpectedAuthenticationPortPort"/>  
  </Event>  
  <Event code="45061" type="Error">  
    <Name textId="TN_EV_FSP_Incorrect_AuthentCRCAuthenticCRC"/>  
    <Description textId="TD_EV_FSP_Incorrect_AuthentCRCAuthenticCRC"/>  
  </Event>  
  <Event code="45062" type="Error">  
    <Name textId="TN_EV_FSP_Incorrect_ProtParCRCProtParamCRC"/>  
    <Description textId="TD_EV_FSP_Incorrect_ProtParCRCProtParamCRC"/>  
  </Event>  
  <Event code="45063" type="Error">  
    <Name textId="TN_EV_FSP_Incorrect_TechParCRCTechnicalParamCRC"/>  
    <Description textId="TD_EV_FSP_Incorrect_TechParCRCTechnicalParamCRC"/>  
  </Event>  
  <Event code="45064" type="Error">  
    <Name textId="TN_EV_FSP_Incorrect_IO_StructCRCIOStructCRC"/>  
    <Description textId="TD_EV_FSP_Incorrect_IO_StructCRCIOStructCRC"/>  
  </Event>  
  <Event code="45065" type="Error">  
    <Name textId="TN_EV_FSP_WatchdogTimeOutOfSpecWatchdogTimeOutOfSpecification"/>  
    <Description textId="TD_EV_FSP_WatchdogTimeOutOfSpecWatchdogTimeOutOfSpecification"/>  
  </Event>  
  <Event code="45066" type="Error">  
    <Name textId="TN_EV_FSP_NoFSVerifyRecordNoFileSystemVerificationRecord"/>  
    <Description textId="TD_EV_FSP_NoFSVerifyRecordNoFileSystemVerificationRecord"/>  
  </Event>  
</EventCollection>
```



```

<UserInterface checkElement="atLeast">
  <ProcessDataRefCollection checkElement="atLeast">
    <ProcessDataRef processDataId="PI_PDin" checkElement="atLeastSequence"
checkAttributes="startsWith processDataId">
      <!-- Space for technology specific functional safety PD Input Data-->
      <ProcessDataRecordItemInfo subindex="#tbd 1..127 #" checkElement="minOccurs 0"
checkAttributes="option displayFormat, option offset, option gradient, option unitCode"/>
      <!-- Safety Code -->
      <ProcessDataRecordItemInfo subindex="127" displayFormat="Hex"/>
      <!-- Space for technology specific non-safety PD Input Data-->
      <ProcessDataRecordItemInfo subindex="#tbd 128..255 #" checkElement="minOccurs 0"
checkAttributes="option displayFormat, option offset, option gradient, option unitCode"/>
    </ProcessDataRef>
    <ProcessDataRef processDataId="PO_PDout" checkElement="atLeastSequence"
checkAttributes="startsWith processDataId">
      <!-- Space for technology specific functional safety PD Input Data-->
      <ProcessDataRecordItemInfo subindex="#tbd 1..127 #" checkElement="minOccurs 0"
checkAttributes="option displayFormat, option offset, option gradient, option unitCode"/>
      <!-- Safety Code -->
      <ProcessDataRecordItemInfo subindex="127" displayFormat="Hex"/>
      <!-- Space for technology specific non-safety PD Input Data-->
      <ProcessDataRecordItemInfo subindex="#tbd 128..255 #" checkElement="minOccurs 0"
checkAttributes="option displayFormat, option offset, option gradient, option unitCode"/>
    </ProcessDataRef>
  </ProcessDataRefCollection>
  <MenuCollection checkElement="atLeast">
    <Menu id="M_OR_Param" checkElement="atLeastSequence">
      <MenuRef menuId="M_OR_Param_" checkElement="minOccurs 0" checkAttributes="startsWith
menuId"/>
      <MenuRef menuId="M_OR_FST_Param" checkElement="minOccurs 0"/>
      <MenuRef menuId="M_OR_FSP_Param"/>
      <MenuRef menuId="M_OMSR_FSP_Param_Aux"/>
    </Menu>
    <Menu id="M_MSR_Param" checkElement="atLeastSequence">
      <MenuRef menuId="M_MSR_Param_" checkElement="minOccurs 0" checkAttributes="startsWith
menuId"/>
      <MenuRef menuId="M_MSR_FST_Param" checkElement="minOccurs 0"/>
      <MenuRef menuId="M_MSR_FSP_Param"/>
      <MenuRef menuId="M_OMSR_FSP_Param_Aux"/>
    </Menu>
    <Menu id="M_OR_FST_Param" contextConstraints="ParameterMenu" checkElement="minOccurs 0">
      <Name textId="TN_M_FST_Param"/>
      <MenuRef menuId="M_OR_FST_Param" checkElement="minOccurs 0" checkAttributes="startsWith
menuId"/>
      <VariableRef variableId="V_FST_" accessRightRestriction="ro" checkElement="minOccurs 0"
checkAttributes="startsWith variableId"/>
      <RecordItemRef variableId="V_FST_" subindex="#tbd#" accessRightRestriction="ro"
checkElement="minOccurs 0" checkAttributes="startsWith variableId"/>
    </Menu>
    <Menu id="M_MSR_FST_Param" contextConstraints="ParameterMenu" checkElement="minOccurs 0">
      <Name textId="TN_M_FST_Param"/>
      <MenuRef menuId="M_MSR_FST_Param" checkElement="minOccurs 0" checkAttributes="startsWith
menuId"/>
      <VariableRef variableId="V_FST_" checkElement="minOccurs 0" checkAttributes="startsWith
variableId, option accessRightRestriction"/>
      <RecordItemRef variableId="V_FST_" subindex="#tbd#" checkElement="minOccurs 0"
checkAttributes="startsWith variableId, option accessRightRestriction"/>
    </Menu>
    <Menu id="M_OR_FSP_Param" contextConstraints="ParameterMenu" checkElement="atLeastSequence">
      <Name textId="TN_M_FSP_Param"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="1" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="2" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="3" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="4" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Protocol" subindex="1" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Protocol" subindex="2" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Protocol" subindex="3" accessRightRestriction="ro"
displayFormat="Dec.0" offset="0.0" gradient="1.0" unitCode="1056"/>
      <RecordItemRef variableId="V_FSP_Protocol" subindex="4" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Protocol" subindex="5" accessRightRestriction="ro"/>
      <RecordItemRef variableId="V_FSP_Protocol" subindex="6" accessRightRestriction="ro"/>
    </Menu>
    <Menu id="M_MSR_FSP_Param" contextConstraints="ParameterMenu">
      <Name textId="TN_M_FSP_Param"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="1"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="2"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="3"/>
      <RecordItemRef variableId="V_FSP_Authenticity" subindex="4"/>

```

```

    <RecordItemRef variableId="V_FSP_Protocol" subindex="1"/>
    <RecordItemRef variableId="V_FSP_Protocol" subindex="2"/>
    <RecordItemRef variableId="V_FSP_Protocol" subindex="3" displayFormat="Dec.0" offset="0.0"
gradient="1.0" unitCode="1056"/>
    <RecordItemRef variableId="V_FSP_Protocol" subindex="4"/>
    <RecordItemRef variableId="V_FSP_Protocol" subindex="5"/>
    <RecordItemRef variableId="V_FSP_Protocol" subindex="6"/>
  </Menu>
  <Menu id="M_OMSR_FSP_Param_Aux" contextConstraints="ParameterMenu">
    <Name textId="TN_M_FSP_Param_Aux"/>
    <VariableRef variableId="V_FSP_TimeToReady" displayFormat="Dec.0" offset="0.0"
gradient="10.0" unitCode="1056"/>
    <VariableRef variableId="V_FSP_MinShutDownTime" displayFormat="Dec.0" offset="0.0"
gradient="10.0" unitCode="1056"/>
    <VariableRef variableId="V_FSP_WCDT" displayFormat="Dec.0" offset="0.0" gradient="1.0"
unitCode="1056"/>
    <VariableRef variableId="V_FSP_OFDT" displayFormat="Dec.0" offset="0.0" gradient="1.0"
unitCode="1056"/>
  </Menu>
  <Menu id="M_OMSR_Observe">
    <VariableRef variableId="V_ProcessDataInput" contextConstraints="ObservationMenu"
checkElement="minOccurs 0"/>
    <VariableRef variableId="V_ProcessDataOutput" contextConstraints="ObservationMenu"
checkElement="minOccurs 0"/>
    <RecordItemRef variableId="V_ProcessDataInput" subindex="#tbd#"
contextConstraints="ObservationMenu" checkElement="minOccurs 0" checkAttributes="option
displayFormat, option offset, option gradient, option unitCode"/>
    <RecordItemRef variableId="V_ProcessDataOutput" subindex="#tbd#"
contextConstraints="ObservationMenu" checkElement="minOccurs 0" checkAttributes="option
displayFormat, option offset, option gradient, option unitCode"/>
  </Menu>
</MenuCollection>
<ObserverRoleMenuSet>
  <IdentificationMenu menuId="M_OR_Ident" checkElement="minOccurs 1"/>
  <ParameterMenu menuId="M_OR_Param"/>
  <ObservationMenu menuId="M_OMSR_Observe" checkElement="minOccurs 1"/>
  <DiagnosisMenu menuId="M_OR_Diag" checkElement="minOccurs 1"/>
</ObserverRoleMenuSet>
<MaintenanceRoleMenuSet>
  <IdentificationMenu menuId="M_MSR_Ident" checkElement="minOccurs 1"/>
  <ParameterMenu menuId="M_MSR_Param"/>
  <ObservationMenu menuId="M_OMSR_Observe" checkElement="minOccurs 1"/>
  <DiagnosisMenu menuId="M_MSR_Diag" checkElement="minOccurs 1"/>
</MaintenanceRoleMenuSet>
<SpecialistRoleMenuSet>
  <IdentificationMenu menuId="M_MSR_Ident" checkElement="minOccurs 1"/>
  <ParameterMenu menuId="M_MSR_Param"/>
  <ObservationMenu menuId="M_OMSR_Observe" checkElement="minOccurs 1"/>
  <DiagnosisMenu menuId="M_MSR_Diag" checkElement="minOccurs 1"/>
</SpecialistRoleMenuSet>
</UserInterface>
<ExternalTextCollection>
  <PrimaryLanguage xml:lang="en" checkElement="minOccurs 1">
    <Text id="TN_P_PDin" value="PD Input"/>
    <Text id="TN_P_PDout" value="PD Output"/>
    <Text id="TN_V_FSP_SafetyCode" value="FS Safety Code"/>
    <Text id="TD_V_FSP_SafetyCode" value="Control/Status octet and CRC."/>
    <Text id="TN_V_FSP_Authenticity" value="Authenticity"/>
    <Text id="TD_V_FSP_Authenticity" value="Authenticity parameters."/>
    <Text id="TN_V_FSP_Authenticity_1" value="FSP_Authenticity_1"/>
    <Text id="TD_V_FSP_Authenticity_1" value=""A-Code" from the upper level FSCP
system."/>
    <Text id="TN_V_FSP_Authenticity_2" value="FSP_Authenticity_2"/>
    <Text id="TD_V_FSP_Authenticity_2" value="Extended "A-Code" from the upper level
FSCP system."/>
    <Text id="TN_V_FSP_Port" value="FSP_Port"/>
    <Text id="TD_V_FSP_Port" value="PortNumber identifying the particular FS-Device."/>
    <Text id="TN_V_FSP_AuthentCRC" value="FSP_AuthentCRC"/>
    <Text id="TD_V_FSP_AuthentCRC" value="CRC-16 across authenticity parameters."/>
    <Text id="TN_V_FSP_Protocol" value="Protocol"/>
    <Text id="TD_V_FSP_Protocol" value="Protocol parameters."/>
    <Text id="TN_V_FSP_ProtVersion" value="FSP_ProtVersion"/>
    <Text id="TD_V_FSP_ProtVersion" value="Protocol version (1=current version)."/>
    <Text id="TN_V_FSP_ProtMode" value="FSP_ProtMode"/>
    <Text id="TD_V_FSP_ProtMode" value="Protocol mode (1=16 bit CRC, 2=32 bit CRC)/>
    <Text id="TN_SV_FSP_ProtMode_crc16" value="16 bit CRC" checkElement="minOccurs 0"/>
    <Text id="TN_SV_FSP_ProtMode_crc32" value="32 bit CRC" checkElement="minOccurs 0"/>
    <Text id="TN_V_FSP_Watchdog" value="FSP_Watchdog"/>
  </PrimaryLanguage>

```

E.3.3 Behavior of "Reset" SystemCommands in SDCI-FS

Table E.1 shows the specific behavior of the "reset" SystemCommands in FS-Devices (see IEC 61131-9). None of these SystemCommands is accepted by the FS-Device in "armed" mode.

Table E.1 – Specific behavior of FS-Device "Reset" SystemCommands

Command (hex)	Command (dec)	Command name	H/M/O	Definition
...				
0x80	128	Device reset	–	Not permitted for implementation in FS-Device
0x81	129	Application reset	M	Permitted in commissioning mode. Authenticity and Protocol parameters shall not be changed.
0x82	130	Restore factory settings	–	Not permitted for implementation in FS-Device
0x83	131	Back-to-box	M	This command shall only be effective whenever the parameter value of FSP_TechParCRC is "0" (commissioning phase)
...				
Key				
H = highly recommended; M = mandatory; O = optional				

E.3.4 Profile Characteristic

The identifier for the common profile SDCI-FS is 16 385 or 0x4001 (see E.5.8). The function class 0x8020 is reserved for future use.

E.3.5 ProcessDataInput and ProcessDataOutput

These variables shall be implemented. The sample IODD in E.5.8 shows details.

E.4 IODD conventions

E.4.1 Naming

While this document and IEC 61131-9 use "parameter" for any data object of a Device and FS-Device, IODDs in [17] use "variable" instead and thus all those data objects are indicated via the prefix "V_". The following rules apply:

- 1) Naming of non-safety parameters shall be "V_xxx". Prefixes "V_FSP", "V_FST" shall be omitted for FS-Devices.
- 2) Naming of FST technology safety parameters shall be "V_FST_xxx".
- 3) Naming of FSP safety parameters shall be "V_FSP_xxx".

These naming conventions shall only be used for SDCI-FS.

E.4.2 Process Data (PD)

The following rules apply for Process Data:

- 1) PDin and PDout shall be described as record.
- 2) Subindices shall be used within the records to differentiate between safety PD and non-safety PD.
- 3) Subindices 1 to 126 shall be used to describe safety PD starting with the highest bit offset.
- 4) Safety Code (see C.5) shall not be described in detail within the IODD. However, Subindex 127 shall be used to describe the Safety Code by means of an OctetStringT (4 or 6 octets) as a dummy to indicate the length of the Safety Code.
- 5) Subindices 128 to 255 shall be used to describe non-safety PD.

- 6) Multiple PD structure definitions selected by conditions are not permitted. This does not impact switching of the user interface to display scaling and units, e.g. °C and °F via conditions.

E.4.3 IODD conventions for user interface

The following rules apply for user interface:

- 1) The IODD shall contain different headlines (menu IDs) for the parameter block types "FS Technology", "FS Protocol", and "FS Auxiliary" in this order.
- 2) FS Technology parameters shall only be referenced in menus marked with the menu ID prefix M_xxx_FST.
- 3) FS Protocol parameters shall only be referenced in menus marked with the menu ID prefix M_xxx_FSP.
- 4) NSR parameters shall not be referenced in menus containing FS parameters.
- 5) The prefixes "FSP" and "FST" shall only be used for FS variables. Corresponding menus shall be colored in yellow.
- 6) User roles are "Observer", "Maintenance", "Specialist". The menus are organized for the "Observer" role (prefix OR) and the combined maintenance and specialist role (prefix MSR). Menus covering all user roles are marked with the prefix OMSR. The menu IDs shall be structured and named as follows:

```
M_OR_FST_Param
M_MSR_FST_Param
M_OR_FSP_Param
M_MSR_FSP_Param
M_OMSR_FSP_Param_Aux
```

E.4.4 Master Tool features

The following rules on how to present the IODD to the user are highly recommended:

- 1) IODD interpreter in Master Tools should show headlines not only for PDin and PDout but also for SR and NSR PD. These headlines should use yellow colors.
- 2) In case of PD observation via ISDU access the variable names should be the same as with cyclic PDs.

E.5 Securing

E.5.1 General

An IODD-based non-safety viewer calculates the 32-bit CRC signature across the FSP parameter description within the IODD. The algorithm for the calculation is shown in this Annex. The safety-related interpreter of the FS-Master Tool checks the correctness of the imported IODD data. Parameter names associated to Index/Subindex are known in the interpreter and can be checked in a safe manner.

An IODD checker is not safety-related and thus not sufficient.

Only one IODD per DeviceID is permitted. A particular FS-Device (hardware) can have two DeviceIDs for example a current DeviceID and a DeviceID of a previous software version.

Figure E.1 shows the algorithm to build the FSP_ParamDescCRC signature. The algorithm shall be used across the Authenticity and the Protocol block (see Table A.1). A seed value "0" shall be used (see D.3.6).

1. General rule: All numerical values are serialized in **big-endian octet order** (most significant octet first).
2. Serialize the **Index** (16-bit unsigned integer) of the FS parameter.
3. Serialize the **bitLength** (16-bit unsigned integer) of the RecordT.
4. Sort the **RecordItems** in ascending order by Subindex.
5. For each **RecordItem** (including the last one) serialize:
 - a) The **Subindex** (8-bit unsigned integer)
 - b) The **bitOffset** (16-bit unsigned integer)
 - c) The **Datatype** (8-bit unsigned integer): 1 = UIntegerT(8), 2 = UIntegerT(16), 3 = UIntegerT(32)
 - d) If and only if a **DefaultValue** is given in the IODD: The DefaultValue (8/16/32 bit unsigned integer according to data type).
 - e) If and only if **SingleValues** or **ValueRange** is given in the IODD: The allowed values. A list of SingleValues is serialized as a sequence of these values, in ascending order. A ValueRange is serialized as a sequence of the minimum and maximum value. Whether SingleValues and/or a ValueRange are allowed depends on the specific RecordItem. See Table E.4
6. Calculate the 4-octet CRC across the octet stream using the CRC polynomial 0xF4ACFB13.

IEC

Figure E.1 – Algorithm to build the FSP parameter CRC signatures**E.5.2 DefaultValues for FSP**

The DefaultValues for FSP_Authenticity1/2, FSP_Port, FSP_AuthentCRC, FSP_TechParCRC, and FSP_ProtParCRC shall be "0". Table E.2 demonstrates the user actions to replace the default values by actual values.

Table E.2 – User actions to replace DefaultValues

Parameter	User actions
FSP_Authenticity1/2	During commissioning, the Authenticity values can be acquired from the gateway and displayed by the Master Tool. SCL will not start with the default value.
FSP_Port	The user shall replace the default "0" by an allowed number with the help of the Master Tool during commissioning. SCL will not start with the default value.
FSP_AuthentCRC	Master Tool calculates this CRC signature
FSP_TechParCRC	The user parameterizes the FS-Device during commissioning or maintenance and uses a Dedicated Tool to calculate the actual value (see 11.7.8 and 11.7.9)
FSP_ProtParCRC	Master Tool calculates this CRC signature

E.5.3 FSP_Authenticity

The values of the authenticity parameters cannot be defined within the IODD. They are maintained by the FS-Master Tool.

E.5.4 FSP_Protocol

The limited variability of the protocol parameters requires the securing mechanism specified in E.5.1.

Table E.3 lists the RecordItems of FSP_Protocol to be serialized.

Table E.3 – RecordItems of FSP_Protocol where allowed values shall be serialized

RecordItem	Serialized as
FSP_ProtVersion	List of 8-bit unsigned integer containing the allowed values, in ascending order
FSP_ProtMode	List of 8-bit unsigned integer containing the allowed values, in ascending order
FSP_Watchdog	Minimum value and maximum value of the contiguous range of allowed values
Any other	All values according to the data type are allowed, therefore nothing is serialized

E.5.5 FSP_IO_Description

The FSP_IO_Description parameters do not require a particular securing mechanism since these instance values are straight forward. The IODD designer can calculate the CRC signature already and place it into the IODD (see A.2.7).

E.5.6 Sample serialization for FSP_ParamDescCRC

Table E.4 shows a sample serialization for the calculation of the FSP_ParamDescCRC signature in E.5.7. A seed value "0" shall be used since there are no leading zeros (see D.3.6).

Table E.4 – Sample serialization for FSP_ParamDescCRC

Offset	Serialization	IODD items	Expected values
0000	42 00	index	42 00 ($\neq 0$)
0002	00 58	bitLength of index	00 58
0004	01	subindex	01 (Authenticity 1)
0005	00 38	bitOffset	00 38
0007	03	xsi:type=UIntegerT, bitLength=32	03
0008	00 00 00 00	RecordItemInfo/@defaultValue	00 00 00 00
000C	02	subindex	02 (Authenticity 2)
000D	00 18	bitOffset	00 18
000F	03	xsi:type=UIntegerT, bitLength=32	03
0010	00 00 00 00	RecordItemInfo/@defaultValue	00 00 00 00
0014	03	subindex	03 (Port)
0015	00 10	bitOffset	00 10
0017	01	xsi:type=UIntegerT, bitLength=8	01
0018	00	RecordItemInfo/@defaultValue	00
0019	04	subindex	04 (AuthentCRC)
001A	00 00	bitOffset	00 00
001C	02	xsi:type=UIntegerT, bitLength=16	02
001D	00 00	RecordItemInfo/@defaultValue	00 00 (dummy CRC)
001F	42 01	index	42 01
0021	00 60	bitLength of index	00 60
0023	01	subindex	01 (ProtVersion)
0024	00 58	bitOffset	00 58
0026	01	xsi:type=UIntegerT, bitLength=8	01
0027	01	RecordItemInfo/@defaultValue	01
0028	01	SingleValue/@value	01 (this document)
0029	02	subindex	02 (ProtMode)

Offset	Serialization	IODD items	Expected values
002A	00 50	bitOffset	00 50
002C	01	xsi:type=UIntegerT, bitLength=8	01
002D	02	RecordItemInfo/@defaultValue	(Vendor defined)
002E	02	SingleValue/@value	02 (example 32 bit)
002F	03	subindex	03 (Watchdog)
0030	00 40	bitOffset	00 40
0032	02	xsi:type=UIntegerT, bitLength=16	02
0033	00 64	RecordItemInfo/@defaultValue	(Vendor defined)
0035	00 64	ValueRange/@lowerValue	00 64 (example: 100)
0037	13 88	ValueRange/@upperValue	13 88 (example: 5000)
0039	04	subindex	04 (IO_StructCRC)
003A	00 30	bitOffset	00 30
003C	02	xsi:type=UIntegerT, bitLength=16	02
003D	9A 28	RecordItemInfo/@defaultValue (see A.2.7)	(Vendor defined)
003F	05	subindex	05 (TechParCRC)
0040	00 10	bitOffset	00 10
0042	03	xsi:type=UIntegerT, bitLength=32	03
0043	00 00 00 00	RecordItemInfo/@defaultValue	00 00 00 00 (dummy CRC)
0047	06	subindex	06 (ProtParCRC)
0048	00 00	bitOffset	00 00
004A	02	xsi:type=UIntegerT, bitLength=16	02
004B	00 00	RecordItemInfo/@defaultValue	00 00 (dummy CRC)
Calculated 32-bit FSP_ParamDescCRC signature value is: 1860635738			See E.5.7

The sample serialization in Table E.4 shows 77 octets to be secured via the CRC-32 polynomial listed in Table D.1. FS-Master Tool shall check the signature after import of the IODD. Only a few values are variable and "Vendor defined" (see offsets: 002D, 002E, 0033 to 0037, and 003D). FS-Master Tool can compare the remaining values with preset values as an additional safety measure.

The "dummy CRC" are placeholders to be replaced by the FS-Master Tool once the user assigned the actual parameter values.

E.5.7 FST and FSP parameters and Data Storage

FST parameters shall be described within the IODD. A "packed" parameter transfer via one ISDU that is not described within the IODD is possible for Data Storage as long as the result in the Device/FS-Device is the same as with discrete ISDUs (see 11.7.6). A manufacturer/vendor is responsible to guarantee this behavior.

FSP parameters (authenticity and protocol) shall be described within the IODD also and are part of Data Storage.

E.5.8 Sample IODD of an FS-Device

The following XML code represents the sample code of an FS-Device IODD. It refers to the Process Data example in Figure A.2. A complete IODD file with name *IO-Link-Safety-01-20220121-IODD1.1.xml* can be downloaded from the website referenced in Annex I.

This sample IODD contains already calculated CRC signature values:

```
<?xml version="1.0" encoding="utf-8"?>
<IODevice xmlns="http://www.io-link.com/IODD/2010/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.io-link.com/IODD/2010/10 IODD1.1.xsd">
  <DocumentInfo version="V1.00" releaseDate="2022-01-21" copyright="Copyright 2022 IO-Link Community"/>
  <ProfileHeader>
    <ProfileIdentification>IO Device Profile</ProfileIdentification>
    <ProfileRevision>1.1</ProfileRevision>
    <ProfileName>Device Profile for IO Devices</ProfileName>
    <ProfileSource>IO-Link Consortium</ProfileSource>
    <ProfileClassID>Device</ProfileClassID>
    <ISO15745Reference>
      <ISO15745Part>1</ISO15745Part>
      <ISO15745Edition>1</ISO15745Edition>
      <ProfileTechnology>IODD</ProfileTechnology>
    </ISO15745Reference>
  </ProfileHeader>
  <ProfileBody>
    <DeviceIdentity vendorId="65535" vendorName="IO-Link Community" deviceId="4608">
      <VendorText textId="T_VendorText"/>
      <VendorUrl textId="T_VendorUrl"/>
      <DeviceName textId="T_DeviceName"/>
      <DeviceFamily textId="T_DeviceFamily"/>
      <DeviceVariantCollection>
        <DeviceVariant productId="Safety-01" deviceSymbol="IO-Link-Safety-pic.png" deviceIcon="IO-Link-Safety-icon.png">
          <Name textId="TN_ProductName"/>
          <Description textId="TD_ProductDescr"/>
        </DeviceVariant>
      </DeviceVariantCollection>
    </DeviceIdentity>
    <DeviceFunction>
      <Features blockParameter="true" dataStorage="true" profileCharacteristic="16384 16385">
        <SupportedAccessLocks parameter="false" dataStorage="false" localParameterization="false" localUserInterface="false"/>
      </Features>
      <DatatypeCollection>
        <Datatype id="D_FSP_Authenticity" xsi:type="RecordT" bitLength="88" subindexAccessSupported="false">
          <RecordItem subindex="1" bitOffset="56">
            <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
            <Name textId="TN_V_FSP_Authenticity_1"/>
            <Description textId="TD_V_FSP_Authenticity_1"/>
          </RecordItem>
          <RecordItem subindex="2" bitOffset="24">
            <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
            <Name textId="TN_V_FSP_Authenticity_2"/>
            <Description textId="TD_V_FSP_Authenticity_2"/>
          </RecordItem>
          <RecordItem subindex="3" bitOffset="16">
            <SimpleDatatype xsi:type="UIntegerT" bitLength="8"/>
            <Name textId="TN_V_FSP_Port"/>
            <Description textId="TD_V_FSP_Port"/>
          </RecordItem>
          <RecordItem subindex="4" bitOffset="0">
            <SimpleDatatype xsi:type="UIntegerT" bitLength="16"/>
            <Name textId="TN_V_FSP_AuthentCRC"/>
            <Description textId="TD_V_FSP_AuthentCRC"/>
          </RecordItem>
        </Datatype>
        <Datatype id="D_FSP_Protocol" xsi:type="RecordT" bitLength="96" subindexAccessSupported="false">
          <RecordItem subindex="1" bitOffset="88">
            <SimpleDatatype xsi:type="UIntegerT" bitLength="8">
              <SingleValue value="1"/>
            </SimpleDatatype>
            <Name textId="TN_V_FSP_ProtVersion"/>
            <Description textId="TD_V_FSP_ProtVersion"/>
          </RecordItem>
          <RecordItem subindex="2" bitOffset="80">
            <SimpleDatatype xsi:type="UIntegerT" bitLength="8">
              <!-- Which of these SingleValues is supported is device specific. Only one shall be referenced. -->
            </SimpleDatatype>
          </RecordItem>
        </Datatype>
      </DatatypeCollection>
    </DeviceFunction>
  </ProfileBody>
</IODevice>
```

```

<SingleValue value="1" checkElement="minOccurs 0">
  <Name textId="TN_SV_FSP_ProtMode_crc16"/>
</SingleValue>
-->
<SingleValue value="2">
  <Name textId="TN_SV_FSP_ProtMode_crc32"/>
</SingleValue>
</SimpleDatatype>
<Name textId="TN_V_FSP_ProtMode"/>
<Description textId="TD_V_FSP_ProtMode"/>
</RecordItem>
<RecordItem subindex="3" bitOffset="64">
  <SimpleDatatype xsi:type="UIntegerT" bitLength="16">
    <!-- Which ValueRange is supported is device specific (but the lowerValue must be >0) -->
    <ValueRange lowerValue="100" upperValue="5000"/>
  </SimpleDatatype>
  <Name textId="TN_V_FSP_Watchdog"/>
  <Description textId="TD_V_FSP_Watchdog"/>
</RecordItem>
<RecordItem subindex="4" bitOffset="48">
  <SimpleDatatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FSP_IO_StructCRC"/>
  <Description textId="TD_V_FSP_IO_StructCRC"/>
</RecordItem>
<RecordItem subindex="5" bitOffset="16">
  <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
  <Name textId="TN_V_FSP_TechParCRC"/>
  <Description textId="TD_V_FSP_TechParCRC"/>
</RecordItem>
<RecordItem subindex="6" bitOffset="0">
  <SimpleDatatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FSP_ProtParCRC"/>
  <Description textId="TD_V_FSP_ProtParCRC"/>
</RecordItem>
</Datatype>
<Datatype id="D_FST_State" xsi:type="BooleanT">
  <SingleValue value="false">
    <Name textId="TN_SV_FST_State_inactive"/>
  </SingleValue>
  <SingleValue value="true">
    <Name textId="TN_SV_FST_State_active"/>
  </SingleValue>
</Datatype>
<Datatype id="D_FSP_PDin" xsi:type="RecordT" bitLength="112"
subindexAccessSupported="false">
  <!-- Technology specific safety process data have subindex 1..126 -->
  <!-- Datatype order for subindices is: boolean, integer16, integer32 -->
  <!-- Boolean array -->
  <!-- There may be no gaps between the booleans, but the last octet may contain less than
eight booleans. -->
  <RecordItem subindex="1" bitOffset="104">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_1"/>
    <Description textId="TD_V_FST_State"/>
  </RecordItem>
  <RecordItem subindex="2" bitOffset="105">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_2"/>
    <Description textId="TD_V_FST_State"/>
  </RecordItem>
  <RecordItem subindex="3" bitOffset="106">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_3"/>
    <Description textId="TD_V_FST_State"/>
  </RecordItem>
  <RecordItem subindex="4" bitOffset="107">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_4"/>
    <Description textId="TD_V_FST_State"/>
  </RecordItem>
  <RecordItem subindex="5" bitOffset="108">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_5"/>
    <Description textId="TD_V_FST_State"/>
  </RecordItem>
  <RecordItem subindex="6" bitOffset="109">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_6"/>

```

```

    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="7" bitOffset="110">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_7"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="8" bitOffset="111">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_8"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="9" bitOffset="96">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_9"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="10" bitOffset="97">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_10"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="11" bitOffset="98">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_11"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="12" bitOffset="99">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_12"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<RecordItem subindex="13" bitOffset="100">
    <DatatypeRef datatypeId="D_FST_State"/>
    <Name textId="TN_V_FST_State_13"/>
    <Description textId="TD_V_FST_State"/>
</RecordItem>
<!-- Integer16 values -->
<!-- Integer values always are aligned to byte borders. -->
<RecordItem subindex="14" bitOffset="80">
    <SimpleDatatype xsi:type="IntegerT" bitLength="16"/>
    <Name textId="TN_V_FST_Integer16Value"/>
    <Description textId="TD_V_FST_Integer16Value"/>
</RecordItem>
<!-- Safety Code has fixed subindex 127 -->
<RecordItem subindex="127" bitOffset="32">
    <SimpleDatatype xsi:type="OctetStringT" fixedLength="6"/>
    <Name textId="TN_V_FSP_SafetyCode"/>
    <Description textId="TD_V_FSP_SafetyCode"/>
</RecordItem>
<!-- Non-safety process data has subindex 128..255 -->
<RecordItem subindex="128" bitOffset="0">
    <SimpleDatatype xsi:type="UIntegerT" bitLength="32"/>
    <Name textId="TN_V_Revolutions"/>
    <Description textId="TD_V_Revolutions"/>
</RecordItem>
</Datatype>
<Datatype id="D_FSP_PDout" xsi:type="RecordT" bitLength="48"
subindexAccessSupported="false">
    <!-- Technology specific safety process data have subindex 1..126 -->
    <!-- Datatype order for subindices is: boolean, integer16, integer32 -->
    <!-- Safety Code has fixed subindex 127 -->
    <RecordItem subindex="127" bitOffset="0">
        <SimpleDatatype xsi:type="OctetStringT" fixedLength="6"/>
        <Name textId="TN_V_FSP_SafetyCode"/>
        <Description textId="TD_V_FSP_SafetyCode"/>
    </RecordItem>
    <!-- Non-safety process data has subindex 128..255 -->
</Datatype>
</DatatypeCollection>
<VariableCollection>
    <StdVariableRef id="V_DirectParameters_1"/>
    <StdVariableRef id="V_SystemCommand">
        <StdSingleValueRef value="129"/>
        <StdSingleValueRef value="131"/>
    </StdVariableRef>
    <StdVariableRef id="V_VendorName" defaultValue="IO-Link Community"/>
    <StdVariableRef id="V_VendorText" defaultValue="http://www.io-link.com"/>

```

```

<StdVariableRef id="V_ProductName" defaultValue="IO-Link Safety Device"/>
<StdVariableRef id="V_ProductID" defaultValue="Safety-01"/>
<StdVariableRef id="V_ProductText" defaultValue="Sample IODD for a device with IO-Link
Safety"/>
<StdVariableRef id="V_SerialNumber"/>
<StdVariableRef id="V_HardwareRevision"/>
<StdVariableRef id="V_FirmwareRevision"/>
<StdVariableRef id="V_ApplicationSpecificTag" excludedFromDataStorage="false"
defaultValue="***"/>
<StdVariableRef id="V_DeviceStatus" defaultValue="0"/>
<StdVariableRef id="V_DetailedDeviceStatus" fixedLengthRestriction="4"/>
<StdVariableRef id="V_ProcessDataInput"/>
<StdVariableRef id="V_ProcessDataOutput"/>
<Variable id="V_CP_FunctionTag" index="25" accessRights="rw" excludedFromDataStorage="false"
defaultValue="***">
  <Datatype xsi:type="StringT" fixedLength="32" encoding="UTF-8"/>
  <Name textId="TN_V_CP_FunctionTag"/>
  <Description textId="TD_V_CP_FunctionTag"/>
</Variable>
<Variable id="V_CP_LocationTag" index="26" accessRights="rw" excludedFromDataStorage="false"
defaultValue="***">
  <Datatype xsi:type="StringT" fixedLength="32" encoding="UTF-8"/>
  <Name textId="TN_V_CP_LocationTag"/>
  <Description textId="TD_V_CP_LocationTag"/>
</Variable>
<Variable index="16896" id="V_FSP_Authenticity" accessRights="rw">
  <DatatypeRef datatypeId="D_FSP_Authenticity"/>
  <RecordItemInfo subindex="1" defaultValue="0"/>
  <RecordItemInfo subindex="2" defaultValue="0"/>
  <RecordItemInfo subindex="3" defaultValue="0"/>
  <RecordItemInfo subindex="4" defaultValue="0"/>
  <Name textId="TN_V_FSP_Authenticity"/>
  <Description textId="TD_V_FSP_Authenticity"/>
</Variable>
<Variable index="16897" id="V_FSP_Protocol" accessRights="rw">
  <DatatypeRef datatypeId="D_FSP_Protocol"/>
  <!-- FSP_ProtVersion: 1= valid -->
  <RecordItemInfo subindex="1" defaultValue="1"/>
  <!-- FSP_ProtMode: 1 (16 bit CRC) or 2 (32 bit CRC)= valid -->
  <RecordItemInfo subindex="2" defaultValue="2"/>
  <!-- FSP_Watchdog: 1 .. 65535 = valid -->
  <RecordItemInfo subindex="3" defaultValue="100"/>
  <!-- FSP_IO_StructCRC: = valid -->
  <RecordItemInfo subindex="4" defaultValue="39464"/>
  <!-- FSP_TechParCRC: 0= invalid -->
  <RecordItemInfo subindex="5" defaultValue="0"/>
  <!-- FSP_ProtParCRC: 0= invalid -->
  <RecordItemInfo subindex="6" defaultValue="0"/>
  <Name textId="TN_V_FSP_Protocol"/>
  <Description textId="TD_V_FSP_Protocol"/>
</Variable>
<!-- Note: Variable FSP_VerifyRecord (index="16898") shall not be described in the IODD. -->
<Variable id="V_FSP_TimeToReady" index="16912" accessRights="ro" defaultValue="10">
  <!-- resolution 10 ms -->
  <Datatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FSP_TimeToReady"/>
  <Description textId="TD_V_FSP_TimeToReady"/>
</Variable>
<Variable id="V_FSP_MinShutDownTime" index="16913" accessRights="ro" defaultValue="100">
  <!-- resolution 10 ms -->
  <Datatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FSP_MinShutDownTime"/>
  <Description textId="TD_V_FSP_MinShutDownTime"/>
</Variable>
<Variable id="V_FSP_ParamDescCRC" index="16914" accessRights="ro" defaultValue="1860635738">
  <Datatype xsi:type="UIntegerT" bitLength="32"/>
  <Name textId="TN_V_FSP_ParamDescCRC"/>
  <Description textId="TD_V_FSP_ParamDescCRC"/>
</Variable>
<Variable id="V_FSP_WCDT" index="16915" accessRights="ro" defaultValue="10">
  <!-- in ms -->
  <Datatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FSP_WCDT"/>
  <Description textId="TD_V_FSP_WCDT"/>
</Variable>
<Variable id="V_FSP_OFDT" index="16916" accessRights="ro" defaultValue="10">
  <!-- in ms -->
  <Datatype xsi:type="UIntegerT" bitLength="16"/>

```

```
<Name textId="TN_V_FSP_OFDT"/>
<Description textId="TD_V_FSP_OFDT"/>
</Variable>
<!-- Non-Safety (Standard) Parameter example -->
<Variable index="64" id="V_NonSafetyParameter" accessRights="rw" defaultValue="123">
  <Datatype xsi:type="IntegerT" bitLength="16"/>
  <Name textId="TN_V_NonSafetyParameter"/>
  <Description textId="TD_V_NonSafetyParameter"/>
</Variable>
<!-- FS Technology Parameter example -->
<Variable index="65" id="V_FST_DiscrepancyTime" accessRights="rw" defaultValue="0">
  <Datatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FST_DiscrepancyTime"/>
  <Description textId="TD_V_FST_DiscrepancyTime"/>
</Variable>
<Variable index="66" id="V_FST_Filter" accessRights="rw" defaultValue="4">
  <Datatype xsi:type="UIntegerT" bitLength="16"/>
  <Name textId="TN_V_FST_Filter"/>
  <Description textId="TD_V_FST_Filter"/>
</Variable>
</ValueCollection>
<ProcessDataCollection>
<ProcessData id="P_ProcessData">
  <ProcessDataIn id="PI_PDin" bitLength="112">
    <DatatypeRef datatypeId="D_FSP_PDin"/>
    <Name textId="TN_PDin"/>
  </ProcessDataIn>
  <ProcessDataOut id="PO_PDout" bitLength="48">
    <DatatypeRef datatypeId="D_FSP_PDout"/>
    <Name textId="TN_PDout"/>
  </ProcessDataOut>
</ProcessData>
</ProcessDataCollection>
<EventCollection>
<Event code="45056" type="Warning">
  <Name textId="TN_EV_FSP_TransmissionError_CRCSSignature"/>
  <Description textId="TD_EV_FSP_TransmissionError_CRCSSignature"/>
</Event>
<Event code="45057" type="Warning">
  <Name textId="TN_EV_FSP_TransmissionError_Counter"/>
  <Description textId="TD_EV_FSP_TransmissionError_Counter"/>
</Event>
<Event code="45058" type="Error">
  <Name textId="TN_EV_FSP_TransmissionError_Timeout"/>
  <Description textId="TD_EV_FSP_TransmissionError_Timeout"/>
</Event>
<Event code="45059" type="Error">
  <Name textId="TN_EV_FSP_UnexpectedAuthenticationCode"/>
  <Description textId="TD_EV_FSP_UnexpectedAuthenticationCode"/>
</Event>
<Event code="45060" type="Error">
  <Name textId="TN_EV_FSP_UnexpectedAuthenticationPort"/>
  <Description textId="TD_EV_FSP_UnexpectedAuthenticationPort"/>
</Event>
<Event code="45061" type="Error">
  <Name textId="TN_EV_FSP_Incorrect_AuthentCRC"/>
  <Description textId="TD_EV_FSP_Incorrect_AuthentCRC"/>
</Event>
<Event code="45062" type="Error">
  <Name textId="TN_EV_FSP_Incorrect_ProtParCRC"/>
  <Description textId="TD_EV_FSP_Incorrect_ProtParCRC"/>
</Event>
<Event code="45063" type="Error">
  <Name textId="TN_EV_FSP_Incorrect_TechParCRC"/>
  <Description textId="TD_EV_FSP_Incorrect_TechParCRC"/>
</Event>
<Event code="45064" type="Error">
  <Name textId="TN_EV_FSP_Incorrect_IO_StructCRC"/>
  <Description textId="TD_EV_FSP_Incorrect_IO_StructCRC"/>
</Event>
<Event code="45065" type="Error">
  <Name textId="TN_EV_FSP_WatchdogTimeOutOfSpec"/>
  <Description textId="TD_EV_FSP_WatchdogTimeOutOfSpec"/>
</Event>
<Event code="45066" type="Error">
  <Name textId="TN_EV_FSP_NoFSVerifyRecord"/>
  <Description textId="TD_EV_FSP_NoFSVerifyRecord"/>
</Event>
```

```
</EventCollection>
<UserInterface>
  <ProcessDataRefCollection>
    <ProcessDataRef processDataId="PI_PDin">
      <!-- Space for technology specific functional safety PD Input Data-->
      <ProcessDataRecordItemInfo subindex="1"/>
      <ProcessDataRecordItemInfo subindex="2"/>
      <ProcessDataRecordItemInfo subindex="3"/>
      <ProcessDataRecordItemInfo subindex="4"/>
      <ProcessDataRecordItemInfo subindex="5"/>
      <ProcessDataRecordItemInfo subindex="6"/>
      <ProcessDataRecordItemInfo subindex="7"/>
      <ProcessDataRecordItemInfo subindex="8"/>
      <ProcessDataRecordItemInfo subindex="9"/>
      <ProcessDataRecordItemInfo subindex="10"/>
      <ProcessDataRecordItemInfo subindex="11"/>
      <ProcessDataRecordItemInfo subindex="12"/>
      <ProcessDataRecordItemInfo subindex="13"/>
      <ProcessDataRecordItemInfo subindex="14"/>
      <!-- Safety Code -->
      <ProcessDataRecordItemInfo subindex="127" displayFormat="Hex"/>
      <!-- Space for technology specific non-safety PD Input Data-->
      <ProcessDataRecordItemInfo subindex="128"/>
    </ProcessDataRef>
    <ProcessDataRef processDataId="PO_PDout">
      <!-- Space for technology specific functional safety PD Output Data-->
      <!-- Safety Code -->
      <ProcessDataRecordItemInfo subindex="127" displayFormat="Hex"/>
      <!-- Space for technology specific non-safety PD Output Data-->
    </ProcessDataRef>
  </ProcessDataRefCollection>
  <MenuCollection>
    <Menu id="M_OR_Ident">
      <VariableRef variableId="V_VendorName"/>
      <VariableRef variableId="V_VendorText"/>
      <VariableRef variableId="V_ProductName"/>
      <VariableRef variableId="V_ProductText"/>
      <VariableRef variableId="V_ProductID"/>
      <VariableRef variableId="V_SerialNumber"/>
      <VariableRef variableId="V_HardwareRevision"/>
      <VariableRef variableId="V_FirmwareRevision"/>
      <VariableRef variableId="V_ApplicationSpecificTag" accessRightRestriction="ro"/>
      <VariableRef variableId="V_CP_FunctionTag" accessRightRestriction="ro"/>
      <VariableRef variableId="V_CP_LocationTag" accessRightRestriction="ro"/>
    </Menu>
    <Menu id="M_MSR_Ident">
      <VariableRef variableId="V_VendorName"/>
      <VariableRef variableId="V_VendorText"/>
      <VariableRef variableId="V_ProductName"/>
      <VariableRef variableId="V_ProductText"/>
      <VariableRef variableId="V_ProductID"/>
      <VariableRef variableId="V_SerialNumber"/>
      <VariableRef variableId="V_HardwareRevision"/>
      <VariableRef variableId="V_FirmwareRevision"/>
      <VariableRef variableId="V_ApplicationSpecificTag"/>
      <VariableRef variableId="V_CP_FunctionTag"/>
      <VariableRef variableId="V_CP_LocationTag"/>
    </Menu>
    <Menu id="M_OR_Param">
      <MenuRef menuId="M_OR_Param_NonSafety"/>
      <MenuRef menuId="M_OR_FST_Param"/>
      <MenuRef menuId="M_OR_FSP_Param"/>
      <MenuRef menuId="M_OMSR_FSP_Param_Aux"/>
    </Menu>
    <Menu id="M_MSR_Param">
      <MenuRef menuId="M_MSR_Param_NonSafety"/>
      <MenuRef menuId="M_MSR_FST_Param"/>
      <MenuRef menuId="M_MSR_FSP_Param"/>
      <MenuRef menuId="M_OMSR_FSP_Param_Aux"/>
      <MenuRef menuId="M_MSR_Param_GeneralSettings"/>
    </Menu>
    <Menu id="M_OR_Param_NonSafety">
      <Name textId="TN_M_Param_NonSafety"/>
      <VariableRef variableId="V_NonSafetyParameter" accessRightRestriction="ro"/>
    </Menu>
    <Menu id="M_MSR_Param_NonSafety">
      <Name textId="TN_M_Param_NonSafety"/>
      <VariableRef variableId="V_NonSafetyParameter" unitCode="1056"/>
    </Menu>
  </MenuCollection>
</UserInterface>
```



```

</Menu>
<Menu id="M_OR_FST_Param">
  <Name textId="TN_M_FST_Param"/>
  <VariableRef variableId="V_FST_DiscrepancyTime" accessRightRestriction="ro"
unitCode="1056"/>
  <VariableRef variableId="V_FST_Filter" accessRightRestriction="ro" unitCode="1056"/>
</Menu>
<Menu id="M_MSR_FST_Param">
  <Name textId="TN_M_FST_Param"/>
  <VariableRef variableId="V_FST_DiscrepancyTime" unitCode="1056"/>
  <VariableRef variableId="V_FST_Filter" unitCode="1056"/>
</Menu>
<Menu id="M_OR_FSP_Param">
  <Name textId="TN_M_FSP_Param"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="1" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="2" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="3" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="4" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="1" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="2" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="3" accessRightRestriction="ro"
displayFormat="Dec.0" offset="0.0" gradient="1.0" unitCode="1056"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="4" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="5" accessRightRestriction="ro"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="6" accessRightRestriction="ro"/>
</Menu>
<Menu id="M_MSR_FSP_Param">
  <Name textId="TN_M_FSP_Param"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="1"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="2"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="3"/>
  <RecordItemRef variableId="V_FSP_Authenticity" subindex="4"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="1"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="2"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="3" displayFormat="Dec.0" offset="0.0"
gradient="1.0" unitCode="1056"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="4"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="5"/>
  <RecordItemRef variableId="V_FSP_Protocol" subindex="6"/>
</Menu>
<Menu id="M_OMSR_FSP_Param_Aux">
  <Name textId="TN_M_FSP_Param_Aux"/>
  <VariableRef variableId="V_FSP_TimeToReady" displayFormat="Dec.0" offset="0.0"
gradient="10.0" unitCode="1056"/>
  <VariableRef variableId="V_FSP_MinShutDownTime" displayFormat="Dec.0" offset="0.0"
gradient="10.0" unitCode="1056"/>
  <VariableRef variableId="V_FSP_WCDT" displayFormat="Dec.0" offset="0.0" gradient="1.0"
unitCode="1056"/>
  <VariableRef variableId="V_FSP_OFDT" displayFormat="Dec.0" offset="0.0" gradient="1.0"
unitCode="1056"/>
</Menu>
<Menu id="M_MSR_Param_GeneralSettings">
  <Name textId="TN_M_CP_Param_GeneralSettings"/>
  <VariableRef variableId="V_SystemCommand">
    <Button buttonValue="129">
      <Description textId="TD_STD_SystemCommand_ApplicationReset"/>
    </Button>
  </VariableRef>
</Menu>
<Menu id="M_OMSR_Observe">
  <VariableRef variableId="V_ProcessDataInput"/>
  <VariableRef variableId="V_ProcessDataOutput"/>
</Menu>
<Menu id="M_OR_Diag">
  <MenuRef menuId="M_OMSR_CP_Diag_DeviceStatusInfo"/>
</Menu>
<Menu id="M_MSR_Diag">
  <MenuRef menuId="M_OMSR_CP_Diag_DeviceStatusInfo"/>
  <MenuRef menuId="M_MSR_CP_Diag_ServiceFunctions"/>
</Menu>
<Menu id="M_OMSR_CP_Diag_DeviceStatusInfo">
  <Name textId="TN_M_CP_Diag_DeviceStatusInfo"/>
  <VariableRef variableId="V_DeviceStatus"/>
  <VariableRef variableId="V_DetailedDeviceStatus"/>
</Menu>
<Menu id="M_MSR_CP_Diag_ServiceFunctions">
  <Name textId="TN_M_CP_Diag_ServiceFunctions"/>
  <VariableRef variableId="V_SystemCommand">

```

```

        <Button buttonValue="131">
            <Description textId="TD_STD_SystemCommand_BackToBox"/>
        </Button>
    </VariableRef>
</Menu>
</MenuCollection>
<ObserverRoleMenuSet>
    <IdentificationMenu menuId="M_OR_Ident"/>
    <ParameterMenu menuId="M_OR_Param"/>
    <ObservationMenu menuId="M_OMSR_Observe"/>
    <DiagnosisMenu menuId="M_OR_Diag"/>
</ObserverRoleMenuSet>
<MaintenanceRoleMenuSet>
    <IdentificationMenu menuId="M_MSR_Ident"/>
    <ParameterMenu menuId="M_MSR_Param"/>
    <ObservationMenu menuId="M_OMSR_Observe"/>
    <DiagnosisMenu menuId="M_MSR_Diag"/>
</MaintenanceRoleMenuSet>
<SpecialistRoleMenuSet>
    <IdentificationMenu menuId="M_MSR_Ident"/>
    <ParameterMenu menuId="M_MSR_Param"/>
    <ObservationMenu menuId="M_OMSR_Observe"/>
    <DiagnosisMenu menuId="M_MSR_Diag"/>
</SpecialistRoleMenuSet>
</UserInterface>
</DeviceFunction>
</ProfileBody>
<CommNetworkProfile xsi:type="IOLinkCommNetworkProfileT" iolinkRevision="V1.1">
    <TransportLayers>
        <PhysicalLayer bitrate="COM3" minCycleTime="2000" sioSupported="true"
mSequenceCapability="43">
            <Connection xsi:type="M12-4ConnectionT" connectionSymbol="IO-Link-Safety-con-pic.png">
                <ProductRef productId="Safety-01"/>
                <Description textId="TD_Connection"/>
                <Wire1 function="L+" color="BN"/>
                <Wire2 function="Other" color="WH"/>
                <Wire3 function="L-" color="BU"/>
                <Wire4 function="C/Q" color="BK"/>
            </Connection>
        </PhysicalLayer>
    </TransportLayers>
    <Test>
        <Config1 index="64" testValue="0x55,0x99"/>
        <Config2 index="256" testValue="0x01,0x02,0x03,0x04"/>
        <Config3 index="24"
testValue="0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x20"/>
        <Config7 index="254">
            <EventTrigger appearValue="01" disappearValue="02"/>
            <EventTrigger appearValue="03" disappearValue="04"/>
        </Config7>
    </Test>
</CommNetworkProfile>
<ExternalTextCollection>
    <PrimaryLanguage Xml:lang="en">
        <Text id="T_VendorText" value="USE - Universal, Smart, Easy"/>
        <Text id="T_VendorUrl" value="http://www.io-link.com"/>
        <Text id="T_DeviceName" value="Safety Device"/>
        <Text id="T_DeviceFamily" value="Safety Device Family"/>
        <Text id="TN_ProductName" value="Safety Device 01"/>
        <Text id="TD_ProductDescr" value="Sample for a device with IO-Link Safety"/>
        <Text id="TD_Connection" value="Port type A with OSSDe. Pin 4 = C/Q/OSSD1e, Pin 2 = OSSD2e"/>
        <Text id="TN_PDin" value="PD Input"/>
        <Text id="TN_PDout" value="PD Output"/>
        <Text id="TN_V_CP_FunctionTag" value="Function Tag"/>
        <Text id="TD_V_CP_FunctionTag" value="Possibility to mark a device with function-specific
information."/>
        <Text id="TN_V_CP_LocationTag" value="Location Tag"/>
        <Text id="TD_V_CP_LocationTag" value="Possibility to mark a device with location-specific
information."/>
        <Text id="TD_STD_SystemCommand_ApplicationReset" value="The parameter of the technology-
specific application are set to default values. Identification parameter remain unchanged. An
upload to the data storage of the master will be executed, if activated in the port
configuration of the master."/>
        <Text id="TD_STD_SystemCommand_BackToBox" value="The parameter of the device are set to
factory default values and communication will be inhibited until the next power cycle. Note:
Directly detach the device from the master port!"/>
    <!-- IO-Link Non-Safety Parameter -->
    <Text id="TN_V_NonSafetyParameter" value="Sample Non-Safety Parameter"/>

```



```

    <Text id="TD_V_NonSafetyParameter" value="Configures any functionality of the device which is
not relevant for the safety function."/>
    <Text id="TN_V_Revolutions" value="Revolutions"/>
    <Text id="TD_V_Revolutions" value="Shows a non-safety measurement value."/>
    <!-- IO-Link Safety FS Technology Parameter -->
    <Text id="TN_V_FST_DiscrepancyTime" value="Discrepancy Time"/>
    <Text id="TD_V_FST_DiscrepancyTime" value="Defines the functional safety relevant value for
the technology specific discrepancy time."/>
    <Text id="TN_V_FST_Filter" value="Filter"/>
    <Text id="TD_V_FST_Filter" value="Defines the functional safety relevant value for the
technology specific signal filter setting."/>
    <Text id="TN_V_FST_State_1" value="FS-PD Boolean State 1"/>
    <Text id="TN_V_FST_State_2" value="FS-PD Boolean State 2"/>
    <Text id="TN_V_FST_State_3" value="FS-PD Boolean State 3"/>
    <Text id="TN_V_FST_State_4" value="FS-PD Boolean State 4"/>
    <Text id="TN_V_FST_State_5" value="FS-PD Boolean State 5"/>
    <Text id="TN_V_FST_State_6" value="FS-PD Boolean State 6"/>
    <Text id="TN_V_FST_State_7" value="FS-PD Boolean State 7"/>
    <Text id="TN_V_FST_State_8" value="FS-PD Boolean State 8"/>
    <Text id="TN_V_FST_State_9" value="FS-PD Boolean State 9"/>
    <Text id="TN_V_FST_State_10" value="FS-PD Boolean State 10"/>
    <Text id="TN_V_FST_State_11" value="FS-PD Boolean State 11"/>
    <Text id="TN_V_FST_State_12" value="FS-PD Boolean State 12"/>
    <Text id="TN_V_FST_State_13" value="FS-PD Boolean State 13"/>
    <Text id="TD_V_FST_State" value="Indicates the state of the boolean status data."/>
    <Text id="TN_SV_FST_State_inactive" value="Inactive"/>
    <Text id="TN_SV_FST_State_active" value="Active"/>
    <Text id="TN_V_FST_Integer16Value" value="FS-PD Integer 16 Value"/>
    <Text id="TD_V_FST_Integer16Value" value="Shows the FS-measurement value."/>
    <Text id="TN_V_FSP_SafetyCode" value="FS Safety Code"/>
    <Text id="TD_V_FSP_SafetyCode" value="Control/Status octet and CRC."/>
    <Text id="TN_V_FSP_Authenticity" value="Authenticity"/>
    <Text id="TD_V_FSP_Authenticity" value="Authenticity parameters."/>
    <Text id="TN_V_FSP_Authenticity_1" value="FSP Authenticity_1"/>
    <Text id="TD_V_FSP_Authenticity_1" value=""A-Code" from the upper level FSCP
system."/>
    <Text id="TN_V_FSP_Authenticity_2" value="FSP Authenticity_2"/>
    <Text id="TD_V_FSP_Authenticity_2" value="Extended "A-Code" from the upper level
FSCP system."/>
    <Text id="TN_V_FSP_Port" value="FSP Port"/>
    <Text id="TD_V_FSP_Port" value="PortNumber identifying the particular FS-Device."/>
    <Text id="TN_V_FSP_AuthentCRC" value="FSP AuthentCRC"/>
    <Text id="TD_V_FSP_AuthentCRC" value="CRC-16 across authenticity parameters."/>
    <Text id="TN_V_FSP_Protocol" value="Protocol"/>
    <Text id="TD_V_FSP_Protocol" value="Protocol parameters."/>
    <Text id="TN_V_FSP_ProtVersion" value="FSP ProtVersion"/>
    <Text id="TD_V_FSP_ProtVersion" value="Protocol version (1=current version)."/>
    <Text id="TN_V_FSP_ProtMode" value="FSP ProtMode"/>
    <Text id="TD_V_FSP_ProtMode" value="Protocol mode (1=16 bit CRC, 2=32 bit CRC)"/>
    <!--
    <Text id="TN_SV_FSP_ProtMode_crc16" value="16 bit CRC"/>
    -->
    <Text id="TN_SV_FSP_ProtMode_crc32" value="32 bit CRC"/>
    <Text id="TN_V_FSP_Watchdog" value="FSP Watchdog"/>
    <Text id="TD_V_FSP_Watchdog" value="Monitoring of IO update."/>
    <Text id="TN_V_FSP_IO_StructCRC" value="FSP IO_StructCRC"/>
    <Text id="TD_V_FSP_IO_StructCRC" value="CRC-16 across IO structure description block."/>
    <Text id="TN_V_FSP_TechParCRC" value="FSP TechParCRC"/>
    <Text id="TD_V_FSP_TechParCRC" value="Securing code across FST (technology specific
parameter)."/>
    <Text id="TN_V_FSP_ProtParCRC" value="FSP ProtParCRC"/>
    <Text id="TD_V_FSP_ProtParCRC" value="CRC-16 across protocol parameters"/>
    <Text id="TN_V_FSP_ParamDescCRC" value="FSP ParamDescCRC"/>
    <Text id="TD_V_FSP_ParamDescCRC" value="CRC-32 signature securing authenticity, protocol, and
FS I/O structure description within IOOD."/>
    <Text id="TN_V_FSP_TimeToReady" value="Time To Ready"/>
    <Text id="TD_V_FSP_TimeToReady" value="The time that is required by the device after power on
in order to be ready for communication."/>
    <Text id="TN_V_FSP_MinShutDownTime" value="Minimum shut down time"/>
    <Text id="TD_V_FSP_MinShutDownTime" value="Minimum time to wait after shut down of FS-
Device."/>
    <Text id="TN_V_FSP_WCDT" value="Worst-case Delay Time"/>
    <Text id="TD_V_FSP_WCDT" value="Time from triggering an FS-Device (sensor) until the output
shows a corresponding signal change or Process Data change. For an FS-Device (actuator) it is
the time from signal change or Process Data change to the actuator's safe state."/>
    <Text id="TN_V_FSP_OFDT" value="One Fault Delay Time"/>
    <Text id="TD_V_FSP_OFDT" value="Time from occurrence of a fault within the FS-Device to a
corresponding signal change or Process Data change"/>

```

```
<Text id="TN_EV_FSP_TransmissionError_CRCSTransmission error (CRC
signature)"/>
<Text id="TD_EV_FSP_TransmissionError_CRCSTransmission error (Counter)"/>
<Text id="TN_EV_FSP_TransmissionError_Counter" value="Transmission error (Counter)"/>
<Text id="TD_EV_FSP_TransmissionError_Counter" value=""/>
<Text id="TN_EV_FSP_TransmissionError_Timeout" value="Transmission error (Timeout)"/>
<Text id="TD_EV_FSP_TransmissionError_Timeout" value=""/>
<Text id="TN_EV_FSP_UnexpectedAuthenticationCode" value="Unexpected authentication code"/>
<Text id="TD_EV_FSP_UnexpectedAuthenticationCode" value=""/>
<Text id="TN_EV_FSP_UnexpectedAuthenticationPort" value="Unexpected authentication port"/>
<Text id="TD_EV_FSP_UnexpectedAuthenticationPort" value=""/>
<Text id="TN_EV_FSP_Incorrect_AuthentCRC" value="Incorrect FSP_AuthentCRC"/>
<Text id="TD_EV_FSP_Incorrect_AuthentCRC" value=""/>
<Text id="TN_EV_FSP_Incorrect_ProtParCRC" value="Incorrect FSP_ProtParCRC"/>
<Text id="TD_EV_FSP_Incorrect_ProtParCRC" value=""/>
<Text id="TN_EV_FSP_Incorrect_TechParCRC" value="Incorrect FSP_TechParCRC"/>
<Text id="TD_EV_FSP_Incorrect_TechParCRC" value=""/>
<Text id="TN_EV_FSP_Incorrect_IO_StructCRC" value="Incorrect FSP_IO_StructCRC"/>
<Text id="TD_EV_FSP_Incorrect_IO_StructCRC" value=""/>
<Text id="TN_EV_FSP_WatchdogTimeOutOfSpec" value="Watchdog time out of specification"/>
<Text id="TD_EV_FSP_WatchdogTimeOutOfSpec" value=""/>
<Text id="TN_EV_FSP_NoFSVerifyRecord" value="No FS_VerifyRecord received"/>
<Text id="TD_EV_FSP_NoFSVerifyRecord" value=""/>
<Text id="TN_M_Param_NonSafety" value="Non-Safety (Standard) Parameter"/>
<Text id="TN_M_FST_Param" value="FS Technology Parameter"/>
<Text id="TN_M_FSP_Param" value="FS Protocol Parameter"/>
<Text id="TN_M_FSP_Param_Aux" value="FS Auxiliary Parameter"/>
<Text id="TN_M_CP_Diag_DeviceStatusInfo" value="Device Status Information"/>
<Text id="TN_M_CP_Diag_ServiceFunctions" value="Service Functions"/>
<Text id="TN_M_CP_Param_GeneralSettings" value="General Settings"/>
</PrimaryLanguage>
</ExternalTextCollection>
<Stamp crc="2515840685"><Checker name="IODED-Checker V1.1.7" version="V1.1.7.0"/></Stamp>
</IODevice>
```

Annex F (normative)

Device Tool Interface (DTI) for SDCI

F.1 Purpose of DTI

For integration of SDCI Devices in a Master Tool, IODD files shall be used provided by the Device manufacturer. Syntax and semantics of these files are standardized (see [17]) such that the Devices can be integrated independently from the vendor/manufacturer.

However, some applications and standards such as functional safety require a so-called Dedicated Tool for e.g. parameter setting and validation, at least as a complement to the IODD method. This Dedicated Tool shall communicate with its Device and is responsible for the data integrity according to IEC 62061 and ISO 13849-1. In the following, the term "Device Tool" is used within this document. Without any additional standardized technology, such an SDCI system would force the user

- to know which Device Tool is required for a particular Device,
- to enter the communication parameters of the Device both in the Master Tool and in the Device Tool and to keep the parameters consistent,
- to store consistent configuration and parameterization data from both the Master Tool and the Device Tool at one single place to archive project data.

In addition, it would face the Device manufacturer

- with the necessity to implement the communication functionality for each supported field bus system, and
- with the problem of nested communication whenever the target Device is located in a different network and only a proprietary gateway interconnects the networks.

A solution is the Device Tool Interface (DTI) technology specified herein after. It can be used for safety (FS-Master/FS-Device) as well as for non-safety (Master/Device) SDCI devices.

F.2 Base model

The Device Tool Interface (DTI) comprises three main parts according to Figure 61:

- an invocation interface between Master Tool and Device Tool;
- a backward interface between Master Tool and Device Tool ("Backchannel");
- a communication interface between Device Tool and a Communication Server.

The combination of these three parts leads to the following user interaction.

A Master Tool is supposed to be already installed on a PC running Microsoft Windows operating system. A Device is configured with the help of the corresponding IODD file of the Device manufacturer. This step includes assignment of port addresses and adjustment of the Device parameters defined in the IODD.

Now, DTI allows for associating Device Tool identification with SDCI Device identification. The Master Tool uses DTI specific mechanisms to find the Device Tool for a given Device. It provides for example in the context menu of a selected Device an entry that can be used to invoke the Device Tool. As soon as the Device Tool is active, it identifies the selected Device. The user can instantly establish a communication with the Device without entering address information and alike and assign parameter values. Assigned values can be returned to the Master Tool using the Backchannel.

For the communication server part, DTI relies on technology specified in [19]. DTI comprises mechanisms to store and maintain Device data objects (project data).

F.3 Invocation interface

F.3.1 Overview

The invocation interface is used to transfer information from the representation of the Device in the Master Tool to the Device Tool. In order to achieve a high flexibility and to be able to identify different versions of the interface, both the description of the Device Tool capabilities and the invocation parameters are stored in XML based documents. For the assignment from Master Tool to Device Tool the system registry of the Microsoft Windows operating system is used.

Figure F.1 shows the principle of the DTI invocation interface part.

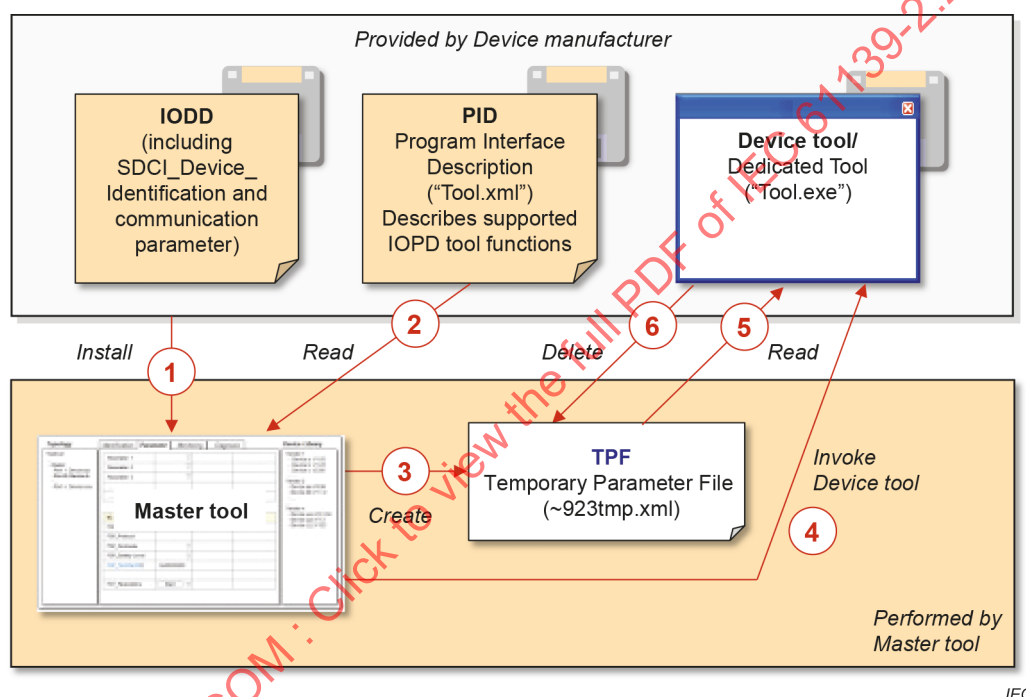


Figure F.1 – Principle of DTI invocation interface

Precondition for the mechanism is the availability of the Master Tool and all used Device Tools on one and the same PC.

For the Tool invocation the following steps are required:

- (1) As usual, the IODD file is imported into the Master Tool. The Device is configured, and communication settings are made. With the help of (SDCI) Device Identification data the Master Tool can find the installed Device Tool and the directory path to the "Program Interface Description" (PID) file. Subclause F.3.2 describes this procedure in detail.
- (2) The Master Tool reads the content of the PID file. This file contains information about the interface version and the supported Tool functions. The structure of the PID file is described in F.3.3.
- (3) Before launching the Device Tool, the Master Tool creates a new "Temporary Parameter File" (TPF) that contains all invocation parameters. See F.3.4 for details.
- (4) The Master Tool launches the Device Tool and passes the name of the TPF. See F.3.4.
- (5) The Device Tool reads and interprets the content of the TPF file.
- (6) The Device Tool deletes the TPF file after processing. See F.3.4.

F.3.2 Detection of Device Tool

F.3.2.1 Registry structure

For DTI to identify the type of an SDCI Device, a specific, unique, and unambiguous SDCI "Device_Identifier" is used in the PC system registry and within the Temporary Parameter File (TPF).

Figure F.2 shows the structure of the DTI part of the registry. Each class in the diagram represents a registry key. Each attribute in the diagram represents a string value of the registry key. The semantics of the attributes is defined in Table F.1 and Table F.2.

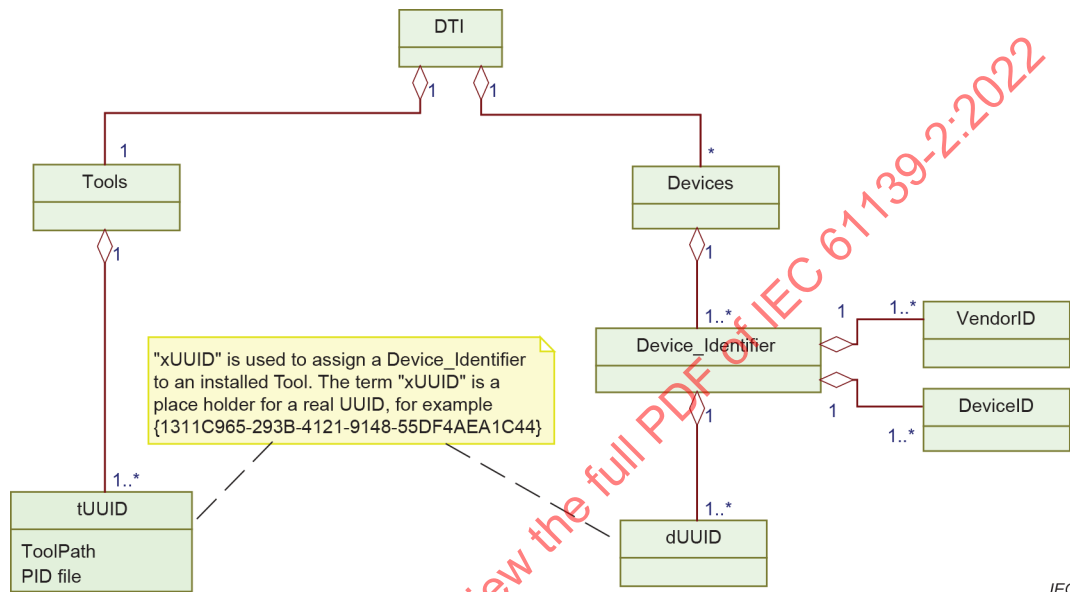


Figure F.2 – Structure of the registry

Since for an SDCI Device_Identifier an unlimited number of "UUID" elements can be inserted, the Master Tool shall handle all Tools of these "UUID" elements.

F.3.2.2 Device Tool specific registry entries

Each version of a Device Tool is represented by one UUID in the system registry.

The installation program of a Device Tool (32 bit or 64 bit) shall insert this UUID as key under its appropriate registry path:

[HKEY_LOCAL_MACHINE\SOFTWARE\IO-Link Community\DTI\Tools](#) or

[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IO-Link Community\DTI\Tools](#)

A Master Tool shall check both registry paths.

Within this key, two attributes with string values shall be used:

- "PID-file", containing the absolute path and name of the installed PID file, and
- "AppPath", containing the absolute path and name of the executable Device Tool file including its file extension (.exe).

Figure F.3 illustrates registry entries for Devices and Tools.

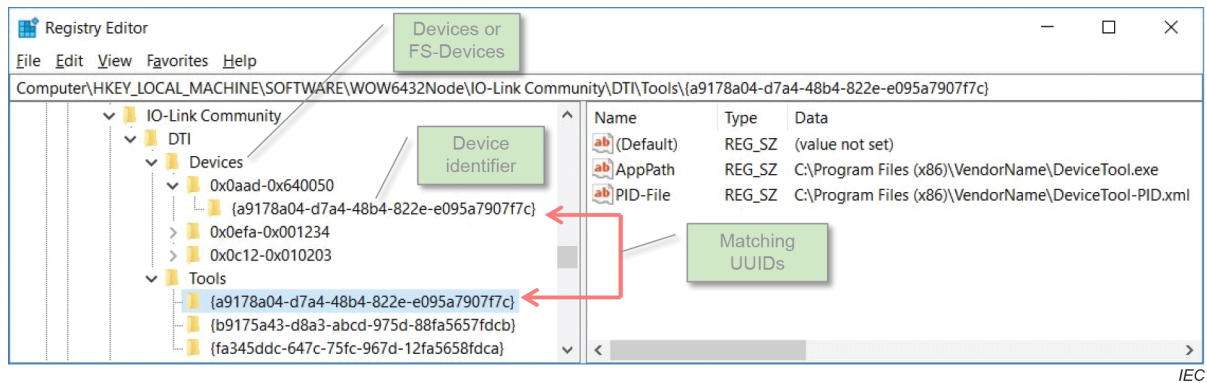


Figure F.3 – Example of a DTI registry

If different versions of a Device Tool for the same Device type exist (same Device_Identifier), each version requires a separate UUID in the registry. In the PID files of the Device Tools, different version information shall be provided in the attribute "ToolDescription" of the element "ToolDescription" (see Table F.1). This leads to multiple items in the context menu of the Master Tool, differing in the description text.

NOTE The advantage of a separate entry of the "ToolPath" keyword is a simpler installation procedure for the Device Tool. It can install the PID file without a need to modify this file.

The installation program of a Device Tool shall also insert each UUID as key under the registry path

HKEY_LOCAL_MACHINE\SOFTWARE\IO-Link Community\DTI\SDCI **Devices\<Device Identifier>**

Devices are identified unambiguously via the following items:

- VendorID (assigned by SDCI support organization, see Annex I);
- DeviceID (assigned by Device/FS-Device manufacturer).

This information is part of the IO Device Description (IODD), which allows the Master Tool to work with the Device (data, parameter) without establishing an online connection to the Device. The IDs can be found at the following locations within an IODD:

- //ISO15745Profile/ProfileBody/DeviceIdentity/@vendorId;
- //ISO15745Profile/ProfileBody/DeviceIdentity/@deviceId.

With the help of the registry, the Master Tool can read the required information about the Device Tool (in case of safety: Dedicated Tool). Location and structure for the entries shall be commonly agreed upon.

All entries shall be provided by the Device Tool under the following registry path:

HKEY_LOCAL_MACHINE\SOFTWARE\IO-Link Community\DTI\Devices

Within this path one or more keys can be inserted with the following field structure:

0xvvvvv-0xdddddd

The meaning of the fields is:

vvvv: Four-character VendorID in hexadecimal coding

dddddd: Six-character DeviceID in hexadecimal coding.

The question mark character "?" can be used in the DeviceID as wildcard to replace one single character. The number of question marks is only limited by the size of the field. If wildcards are used, the Device Tool is responsible for the check whether it supports the selected object.

The assignment to the Tool is made by a string value within this key. The UUID shall be used as name for the string value. The number of string values is not limited, which in turn means an unlimited number of Tools that can be assigned to the same Device.

Examples for valid keys (see Figure F.3):

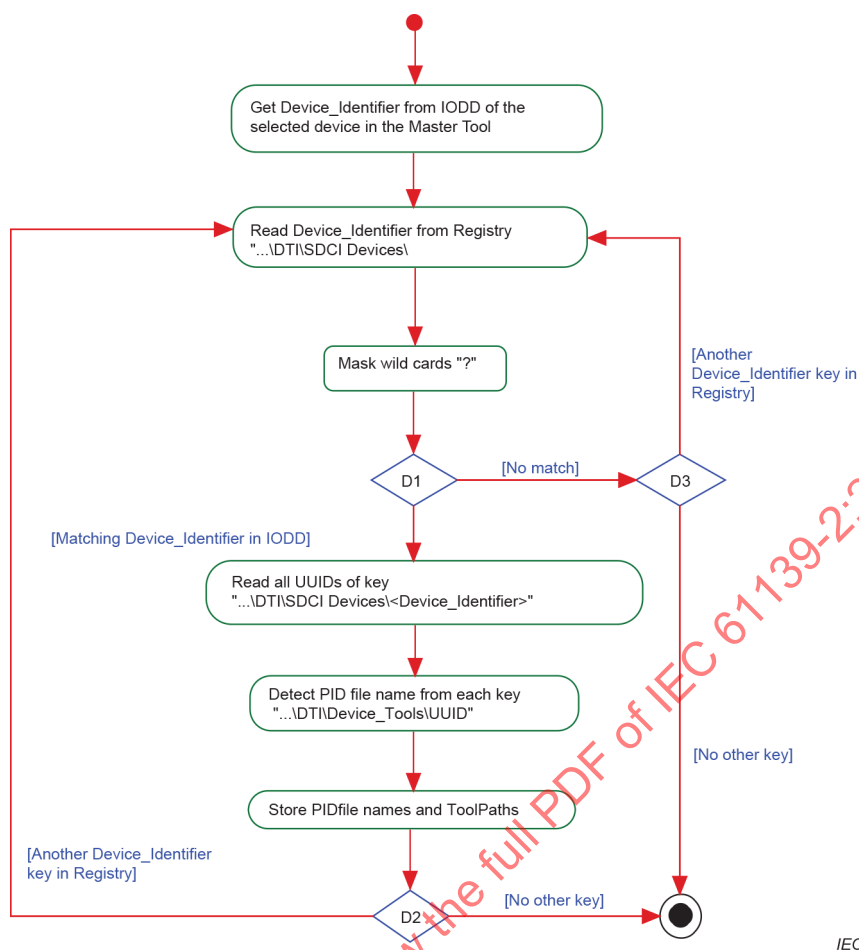
0x0A99-0x00880D	The Tool can be launched in the context of a Device with a DeviceID 0x00880D from the vendor with the VendorID 0x0A99.
0x0F3B-0x002B??	The Tool can be started in the context of Devices with a DeviceID in the range of 0x002B00 to 0x002BFF from the vendor with the VendorID 0x0F3B.

F.3.2.3 Processing of the Registry Data

The installation program of the Device Tool is responsible to insert the keys in the system registry as defined in F.3.2.2.

Figure F.4 shows an activity diagram illustrating the detection of a Device Tool in the registry via SDCI "Device_Identifier".

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022



IEC

Figure F.4 – Detection of a Device tool in registry

NOTE All registry keys in Figure F.4. are relative to the path HKEY_LOCAL_MACHINE\SOFTWARE\IO-Link Community.

In a first step, the Master Tool gets the Device_Identifier from the IODD of the selected object in the Master Tool. Then all sub keys in the system registry path ...DTI\SDCI Devices shall be compared with this Device_Identifier. If a sub key matches (excepting wildcards), the UUID sub key of this key is used to find the PID file name in the registry path DTI\Device Tools\<UUID>. Since the same PID file name can be found in different locations in the registry, the context menu of the Master Tool shall only show the Device Tools with different PID file names. As a last step, the information in the PID file is used to build the menu items of the Master Tool (Figure F.5).

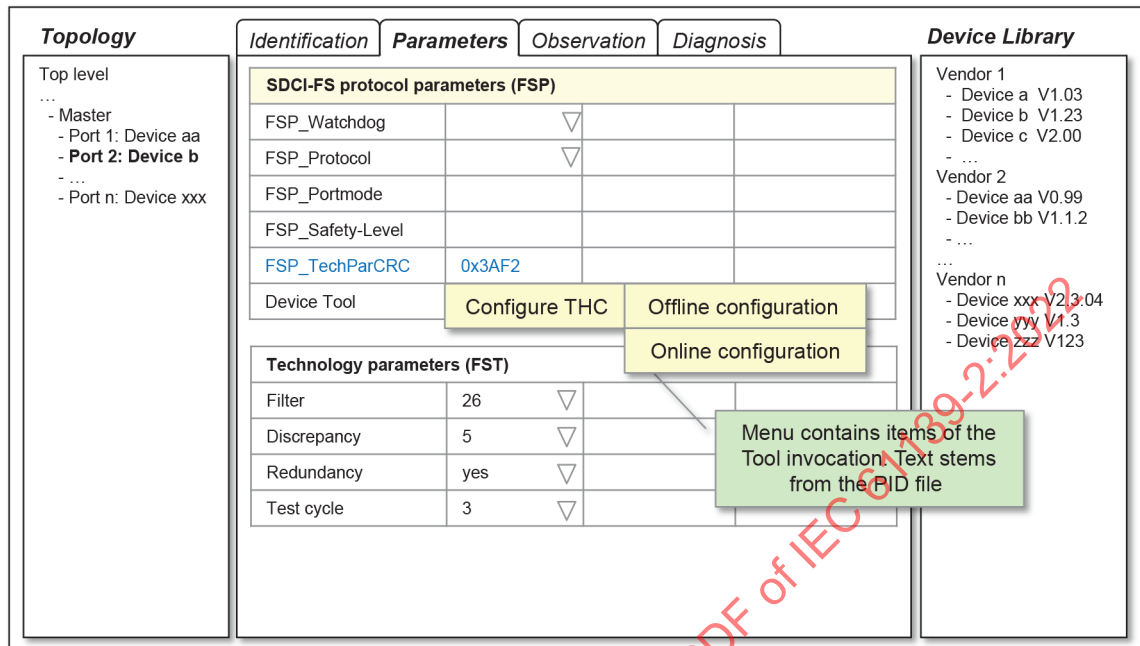
F.3.3 Program Interface Description – PID

F.3.3.1 General

The Program Interface Description (PID) file describes the properties of the Device Tool and contains data which are required by the Master Tool to build menu items in its graphical user interface (GUI). The PID file is an XML document. The corresponding XML schema is defined in F.9.2. UTF-8 shall be used for character encoding.

This PID file shall be provided by the manufacturer of a Device/Device Tool and installed by the installation program associated with the Device Tool. This installation program shall also insert the name and installation path in the system registry (see F.3.2).

The PID file allows the Master Tool to extend its GUI menu structure by the name of the Device Tool such that the user is able to launch the Device Tool for example from the context menu of a selected Device as illustrated exemplary in Figure F.5.



IEC

Figure F.5 – Menu for Device Tool invocation

F.3.3.2 Structure of the PID file

The PID file is an XML based document. Its XML schema is shown in Figure F.13. Namespace URI for this file is <http://www.io-link.com/DTI/2017/02/PID> (see F.9.2). Hint: The schema file for "primitives" shall be within the same file folder than the main schema file (see F.9.6).

The elements of Figure F.13 are specified in Table F.1. The column "SV" indicates the schema version a particular attribute has been introduced.

Table F.1 – Description of PID file elements

Element	Attribute	Type	M/O	SV	Description
ProgramInterface	–	–	M	1.0	Root element
DocumentInfo	–	–	M	1.0	Meta information about the PID file
	Version	xsd:string	M	1.0	Contains the schema version of PID interface definition. Also determines the newest TPF version supported by this tool. The value shall comply with the following regular expression: \\d+(\\.\\d+)* In this version, the string "1.1" shall be used.
General	–	–	M	1.0	General information about the Device Tool
	VendorName	xsd:string	M	1.0	Contains the name of the Device vendor

Element	Attribute	Type	M/O	SV	Description
ToolDescription	–	–	1..n	1.0	Is used to define language dependent text information for description of the Device Tool. A ToolDescription element in English language is mandatory.
	xml:lang	xsd:language	M	1.0	Defines the language of the text. The "2-letter coding" or the "3-letter coding" as defined in ISO 639-2 and ISO 639-3 shall be used.
	ToolName	xsd:string	M	1.0	Describes the function of the Device Tool. This text shall be used to extend the GUI menu items of the Master Tool. Default element in English language shall always be present.
	ToolDescription	xsd:string	O	1.0	Contains a short description of the Device Tool.
Invocation Prefix	–	–	–	1.0	With this element, the command line arguments of the called Device Tool can be modified. If a Device Tool is able to interpret different command line arguments, usually a prefix is used to define the semantic of an argument. If an InvocationPrefix is present in the PID file, the Master Tool shall insert a blank character as delimiter between the InvocationPrefix string and the file name of the TPF. To interpret the command line argument as a file name for a DTI call, a Device Tool shall be launched as follows: <i>DeviceTool.exe -i "c:\tmp\TPF01.xml"</i> In this case, the prefix "-i" shall be entered in the PID file.
	Name	xsd:string	O	1.0	Defines which command line prefix is used when the tool is launched. If this attribute is not present, only the file name of the TPF is used as command line argument. NOTE Since the datatype "string" is used, blank characters (ASCII 32 dec) are allowed. XML Entities are allowed and shall be converted by the Master Tool.
ConformanceClass	–	–	M	1.0	Describes the conformance class of the engineering system.
	Name	xsd:string	M	1.0	Contains the name of the conformance class (F.8.1). One of the following values is allowed: "C1", "C2", or "C3"
EntryPoints	–	–	M	1.0	This optional element shall be used, if a Device Tool has more than one entry point.

Element	Attribute	Type	M/O	SV	Description
EntryPoint	–	–	1 to n	1.0	This element represents an entry point of the Device Tool. Entry points are used to generate additional sub menu items in the "ToolDescription" context menu of the Master Tool. Using entry points a Device Tool can provide direct access to Tool specific views or functions.
	ID	xsd:string	M	1.0	The attribute "ID" identifies an EntryPoint. It shall be unique within a PID file.
InfoText	–	–	1 to n	1.0	The element "InfoText" is used to define language dependent text information for description of the entry point. This information can be used to extend the GUI menu items of the Master Tool. An InfoText element in English language shall always be present here.
	xml:lang	xsd:string	M	1.0	Defines the language of the text. The "2-letter coding" or the "3-letter coding" as defined in ISO 639-2 and ISO 639-3 shall be used.
	EntryPointName	xsd:string	M	1.0	Describes the function of the entry point. This text shall be used to extend the GUI menu items of the Master Tool.
	EntryPointDescription	xsd:string	O	1.0	Contains a short description of the entry point.

F.3.3.3 Example PID file

The following XML code shows an example content of a PID file with EntryPoints.

```
<?xml version="1.0" encoding="UTF-8"?>
<ProgramInterface xmlns="http://www.io-link.com/DTI/2017/02/PID"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:prim="http://www.io-link.com/DTI/2017/02/Primitives" xsi:schemaLocation="http://www.io-link.com/DTI/2017/02/PID
  iosafe_pid_schema_20170225.xsd">
  <DocumentInfo version="V1.0"/>
  <General vendorName="IO-Link Community">
    <ToolDescription name="Configure THC" description="IO-Link Safety Device" lang="en"/>
    <ToolDescription name="Konfiguriere THC" description="IO-Link Safety Device" lang="de"/>
    <InvocationPrefix name="/" />
  </General>
  <EntryPoints>
    <EntryPoint id="1">
      <InfoText name="Offline Configuration" description="Offline Configuration" lang="en"/>
      <InfoText name="Offline Konfiguration" description="Offline Konfiguration" lang="de"/>
    </EntryPoint>
    <EntryPoint id="2">
      <InfoText name="Online Configuration" description="Online Configuration" lang="en"/>
      <InfoText name="Online Konfiguration" description="Online Konfiguration" lang="de"/>
    </EntryPoint>
  </EntryPoints>
  <ConformanceClass name="C3"/>
</ProgramInterface>
```

F.3.4 Temporary Parameter File – TPF

F.3.4.1 General

Due to the large number of parameters to be transferred from the Master Tool to the Device Tool, a parameter transfer by command line arguments is not a good solution. The necessary syntax would become too complex to cover all aspects.

Instead, all required parameters are included into an XML file, called Temporary Parameter File (TPF) by the Master Tool and thus, the name of the XML file is passed as the only command line argument. If the Device Tool requires a command line switch, this information can be extracted from the PID file. See "InvocationPrefix" in Table F.1 for details.

The XML schema for the TPF is defined in F.9.3. For character encoding, UTF-8 shall be used. The Master Tool shall use the newest TPF schema version supported by both the Master Tool and the Device Tool.

After the TPF is interpreted, the Device Tool shall delete the TPF file.

F.3.4.2 Structure of a TPF

The structure of the TPF is defined by the XML schema shown in Figure F.14. This schema is built in a generic manner, which means, a new parameter does not require the schema itself to be updated. Thus, new parameters can be introduced without a new definition of the TPF structure.

Namespace URI for this file is <http://www.io-link.com/DTI/2017/02/TPF> (see F.9.3). The elements of the XML schema in Figure F.14 are specified in Table F.2. The column "SV" indicates the schema version a particular attribute has been introduced.

Table F.2 – Elements of a TPF

Element	Attribute	Type	M/O	SV	Description
InvocationInterface	–	–	M	1.0	Root element
General	–	–	M	1.0	General information about the TPF file
	schemaPath	xsd:string	M	1.0	This attribute defines the path where the schema files for FDT communication schemas and TPF/PID file are stored. • This schema files shall be installed on this path by the Master Tool • The path does not change during runtime of the Master Tool • The path can be used from a Device Tool to initialize the XML parser. NOTE Even if no schema validation is used, some XML parsers need the location of the schema files for initialization. In this case, a Device Tool does not need to install an own set of schema files – it should use the schema files in the path defined by this attribute.
	projectRelatedPath	xsd:string	M	1.0	The attribute "ProjectRelatedPath" contains information about a directory which is assigned to the project context of the Master Tool. A Device Tool should use this path for storage of its Device data. The format and structure of this data is defined by the Device Tool itself. Within this directory, additional subdirectories can be created. The Master Tool is responsible to keep all data in the directory tree in its project context. That means, if the project is copied or archived, also this data shall be copied or archived.

Element	Attribute	Type	M/O	SV	Description
					The attribute "ProjectRelatedPath" contains a unique path (directory) for each combination of Master project and DTI Device Tool. For example, different directories are used for the same tool, if two Master Tool projects are used. The file name in "ProjectRelatedPath" shall consist of the drive letter and an absolute path expression. Alternatively, the UNC notation can be used instead of the drive letter.
	portName	xsd:string	M	1.0	Name of used FS-Master port
	portId	xsd:string	M	1.0	ID of used FS-Master port 1 to n
	masterName	xsd:string	M	1.0	User defined name of FS-Master
	displayNameES	xsd:string	M	1.0	Display name of the Master Tool in the language specified in attribute "currentLanguage". The Device Tool can use this name in error messages or user dialogs to provide more understandable texts.
	currentLanguage	xsd:string	M	1.0	Defines which language shall be used by the Device Tool for TPF. The "2-letter coding" or the "3-letter coding" as defined in ISO 639-2 and ISO 639-3 can be used. If a Device Tool does not support the selected language, the tool shall use its default language.
	commServerProgID	xsd:string	O	1.0	This attribute contains the ProgID of the Communication Server provided by the Master Tool manufacturer. It allows the Device Tool to use the Communication Server functionality. See F.5.6 for details. If this attribute is not provided, the Master Tool does not support a Communication Server.
	busCategory	xsd:string	M	1.1	This attribute is used to specify the used communication protocol. It also can be used to find a corresponding Communication Server. Default value is "2C4CD8B8-D509-4ECB-94A7-019F12569C8B"
	selectedEntryPoint	xsd:string	O	1.0	Defines, which entry point of the Device Tool was selected in the Master Tool when the Device Tool was launched. This attribute shall contain only values defined in the attribute "ID" of any element "EntryPoint" of the corresponding PID file. This attribute allows the Device Tool to show an entry point specific GUI when it was launched. If the PID file does not contain any EntryPoint elements, this attribute shall not be used in the TPF.
	conformanceClass	xsd:string	M	1.0	Contains the name of the conformance class of the Master Tool. One of the following values is allowed: "C2" or "C3". See Table F.8.
DeviceItem	–	–	M	1.0	Used to describe the Device selected in the ES.

Element	Attribute	Type	M/O	SV	Description
	vendorId	xsd:string	M	1.0	See IEC 61131-9
	productId	xsd:string	M	1.0	See IEC 61131-9
	deviceId	xsd:string	M	1.0	See IEC 61131-9
	usedConfigFileCRC	xsd:string	M	1.0	IODD stamp
	usedConfigFile	xsd:string	M	1.0	The keyword usedConfigFile contains the file name of the used description file (e.g. IODD). The file name shall consist of the drive letter, an absolute path expression and the file extension. Alternatively, the UNC notation can be used instead of the drive letter. The Device Tool It is not allowed to modify the content of the description file.
	reference	xsd:string	M	1.0	Used to identify FS-Device within engineering project
	commReference	xsd:string	M	1.0	This attribute is used with the Communication Server (CS) to address a Device instance unambiguously within the PC. The unique nature of this attribute shall be ensured by the Master Tool. The structure of the attribute is only defined by the Master Tool. It is not allowed to interpret the syntax of this keyword in the Device Tool. LineFeed characters (ASCII 10 dec) are not allowed in the string. This attribute shall be provided for all Device instances of a TPF, if the Device Tool wants to use the CS interface (Conformance Class 3 (C3)) and the commReference is different from the DeviceReference.
VariableInstanceData	–	–	M	1.0	Element "VariableInstanceData" is a container for "Variable" elements (= parameter).
Variable	–	–	1 to n	1.0	Contains information about a variable.
	variableId	xsd:string	M	1.0	Contains the parameter ID
Item	–	–	1 to n	1.0	Contains information about a Subindex of a variable.
	subindex	xsd:string	M	1.0	See IEC 61131-9
	value	xsd:string	M	1.0	Contains the parameter value. In absence of a parameter-specific rule for the representation of the value: Numerical values shall use the decimal coding without left-hand zeros. Negative values shall have a hyphen (ASCII 45 dec) prefix. Separator for floating point values is a dot (ASCII 46 dec). Other separators are not permitted.
	state	xsd:string	M	1.0	Contains parameter status
	error	xsd:string	M	1.0	Contains parameter error

F.3.4.3 Example of a TPF

The following XML code shows the content of an exemplary TPF file.

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<InvocationInterface xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.io-link.com/DTI/2017/02/TPF" xsischemaLocation="http://www.io-link.com/DTI/2017/02/TPF
IOsafe_TPF_Schema_20170225.xsd" xmlns:prim="http://www.io-link.com/DTI/2017/02/Primitives">
  <General currentLanguage="en" commServerProgID="" />
  <VariableInstanceData>
    <Variable variableID="V_VendorName">
      <Item subindex="0" state="initial" error="0" value="IO-Link Community" />
    </Variable>
    <Variable variableID="V_VendorText">
      <Item subindex="0" state="initial" error="0" value="http://www.io-link.com" />
    </Variable>
    <Variable variableID="V_ProductName">
      <Item subindex="0" state="initial" error="0" value="IO-Link Safety Device" />
    </Variable>
    <Variable variableID="V_ProductText">
      <Item subindex="0" state="initial" error="0" value="Sample IODD for a device with
IO-Link Safety" />
    </Variable>
    <Variable variableID="V_ProductID">
      <Item subindex="0" state="initial" error="0" value="Safety-01" />
    </Variable>
    <Variable variableID="V_SerialNumber">
      <Item subindex="0" state="empty" error="0" value="" />
    </Variable>
    <Variable variableID="V_HardwareRevision">
      <Item subindex="0" state="empty" error="0" value="" />
    </Variable>
    <Variable variableID="V_FirmwareRevision">
      <Item subindex="0" state="empty" error="0" value="" />
    </Variable>
    <Variable variableID="V_ApplicationSpecificTag">
      <Item subindex="0" state="database" error="0" value="***" />
    </Variable>
    <Variable variableID="V_CP_FunctionTag">
      <Item subindex="0" state="database" error="0" value="***" />
    </Variable>
    <Variable variableID="V_CP_LocationTag">
      <Item subindex="0" state="database" error="0" value="***" />
    </Variable>
    <Variable variableID="V_ProcessDataInput">
      <Item subindex="1" state="empty" error="0" value="" />
      <Item subindex="2" state="empty" error="0" value="" />
      <Item subindex="3" state="empty" error="0" value="" />
      <Item subindex="4" state="empty" error="0" value="" />
      <Item subindex="5" state="empty" error="0" value="" />
      <Item subindex="6" state="empty" error="0" value="" />
      <Item subindex="7" state="empty" error="0" value="" />
      <Item subindex="8" state="empty" error="0" value="" />
      <Item subindex="9" state="empty" error="0" value="" />
      <Item subindex="10" state="empty" error="0" value="" />
      <Item subindex="11" state="empty" error="0" value="" />
      <Item subindex="12" state="empty" error="0" value="" />
      <Item subindex="13" state="empty" error="0" value="" />
      <Item subindex="14" state="empty" error="0" value="" />
      <Item subindex="127" state="empty" error="0" value="" />
      <Item subindex="128" state="empty" error="0" value="" />
    </Variable>
    <Variable variableID="V_ProcessDataOutput">
      <Item subindex="127" state="empty" error="0" value="" />
    </Variable>
    <Variable variableID="V_NonSafetyParameter">
      <Item subindex="0" state="initial" error="0" value="123" />
    </Variable>
    <Variable variableID="V_FST_DiscrepancyTime">
      <Item subindex="0" state="database" error="0" value="5" />
    </Variable>
    <Variable variableID="V_FST_Filter">
      <Item subindex="0" state="database" error="0" value="5" />
    </Variable>
    <Variable variableID="V_FSP_Authenticity">
      <Item subindex="1" state="database" error="0" value="123" />
      <Item subindex="2" state="database" error="0" value="0" />
      <Item subindex="3" state="database" error="0" value="1" />
      <Item subindex="4" state="database" error="0" value="0" />
    </Variable>
    <Variable variableID="V_FSP_Protocol">
      <Item subindex="1" state="database" error="0" value="1" />
      <Item subindex="2" state="database" error="0" value="2" />

```

```

        <Item subindex="3" state="database" error="0" value="100" />
        <Item subindex="4" state="database" error="0" value="39464" />
        <Item subindex="5" state="database" error="0" value="23746" />
        <Item subindex="6" state="database" error="0" value="0" />
    </Variable>
    <Variable variableID="V_FSP_TimeToReady">
        <Item subindex="0" state="initial" error="0" value="10" />
    </Variable>
    <Variable variableID="V_FSP_MinShutDownTime">
        <Item subindex="0" state="initial" error="0" value="100" />
    </Variable>
    <Variable variableID="V_FSP_WCDT">
        <Item subindex="0" state="initial" error="0" value="10" />
    </Variable>
    <Variable variableID="V_FSP_OFDT">
        <Item subindex="0" state="initial" error="0" value="10" />
    </Variable>
    <Variable variableID="V_DeviceStatus">
        <Item subindex="0" state="initial" error="0" value="0" />
    </Variable>
    <Variable variableID="V_DetailedDeviceStatus">
        <Item subindex="1" state="empty" error="0" value="" />
        <Item subindex="2" state="empty" error="0" value="" />
        <Item subindex="3" state="empty" error="0" value="" />
        <Item subindex="4" state="empty" error="0" value="" />
    </Variable>
</VariableInstanceData>
</InvocationInterface>

```

F.3.5 Temporary Backchannel File – TBF

F.3.5.1 General

The TBF should be transferred by a new transaction of the communication server. This transaction is initiated by the Device Tool and can be performed automatically or upon user request. Transaction acknowledgements (TAF) should be implemented indicating reception of the instance values by the Master Tool or indicating a transaction fault (see F.3.6).

F.3.5.2 Structure of the TBF

The structure of the TBF is defined by the XML schema Figure F.15. This schema is built in a generic manner, which means, a new parameter does not require the schema itself to be updated. Thus, new parameters can be introduced without a new definition of the TBF structure.

Namespace URI for this file is <http://www.io-link.com/DTI/2017/02/TBF> (see F.9.4). The elements of Figure F.15 are specified in Table F.3. The column "SV" indicates the schema version a particular attribute has been introduced.

Table F.3 – Elements of the TBF

Element	Attribute	Type	M/O	SV	Description
ReturnInterfaceRequest	–	–	M	1.0	Root element
VariableInstanceData	–	–	M	1.0	The element "VariableInstanceData" is a container for "Variable" elements (= parameter).
Variable	–	–	1 to n	1.0	Contains information about a variable.
	variableId	xsd:string	M	1.0	Contains the parameter ID
Item	–	–	1 to n	1.0	Contains information about a Subindex of a variable.
	subindex	xsd:string	M	1.0	See IEC 61131-9
	value	xsd:string	M	1.0	Contains the parameter value. In absence of a parameter-specific rule for the representation of the value: Numerical values shall use the decimal coding without left-hand zeros. Negative values shall have a hyphen (ASCII 45 dec) prefix. Separator for floating point values is a dot (ASCII 46 dec). Other separators are not permitted.
	state	xsd:string	M	1.0	Contains parameter status
	error	xsd:string	M	1.0	Contains parameter error

F.3.5.3 Example of a TBF

The following XML code shows the content of an exemplary TBF file.

```
<?xml version="1.0" encoding="UTF-8"?>
<ReturnInterfaceRequest xmlns="http://www.io-link.com/DTI/2017/02/TBF"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:prim="http://www.io-link.com/DTI/2017/02/Primitives"
  xsi:schemaLocation="http://www.io-link.com/DTI/2017/02/TBF IOsafe_TBF_Schema_20170225.xsd">
  <VariableInstanceData>
    <Variable variableID="V_ApplicationSpecificTag">
      <Item subindex="0" state="database" error="0" value="***" />
    </Variable>
    <Variable variableID="V_CP_FunctionTag">
      <Item subindex="0" state="database" error="0" value="***" />
    </Variable>
    <Variable variableID="V_CP_LocationTag">
      <Item subindex="0" state="database" error="0" value="***" />
    </Variable>
    <Variable variableID="V_NonSafetyParameter">
      <Item subindex="0" state="initial" error="0" value="123" />
    </Variable>
    <Variable variableID="V_FST_DiscrepancyTime">
      <Item subindex="0" state="database" error="0" value="6" />
    </Variable>
    <Variable variableID="V_FST_Filter">
      <Item subindex="0" state="database" error="0" value="6" />
    </Variable>
    <Variable variableID="V_FSP_Authenticity">
      <Item subindex="1" state="database" error="0" value="123" />
      <Item subindex="2" state="database" error="0" value="0" />
      <Item subindex="3" state="database" error="0" value="1" />
      <Item subindex="4" state="database" error="0" value="0" />
    </Variable>
    <Variable variableID="V_FSP_Protocol">
      <Item subindex="1" state="database" error="0" value="1" />
      <Item subindex="2" state="database" error="0" value="2" />
      <Item subindex="3" state="database" error="0" value="100" />
      <Item subindex="4" state="database" error="0" value="39464" />
      <Item subindex="5" state="database" error="0" value="592319" />
      <Item subindex="6" state="database" error="0" value="0" />
    </Variable>
  </VariableInstanceData>
</ReturnInterfaceRequest>
```

```

    </Variable>
  </VariableInstanceData>
</ReturnInterfaceRequest>

```

F.3.6 Temporary Acknowledgment File – TAF

F.3.6.1 General

Transaction acknowledgements should be implemented indicating reception of the instance values by the Master Tool or indicating a transaction fault. The same mechanism is used as with TBF (see F.3.5).

F.3.6.2 Structure of the TAF

The structure of the TAF is defined by the XML schema shown in Figure F.15. However, the root name has changed to "ReturnInterfaceResponse".

Namespace URI for this file is <http://www.io-link.com/DTI/2017/02/TBF> (see F.9.4). The elements of Figure F.15 are specified in Table F.4. The column "SV" indicates the schema version a particular attribute has been introduced.

Table F.4 – Elements of the TAF

Element	Attribute	Type	M/O	SV	Description
ReturnInterfaceResponse	–	–	M	1.0	Root element
Response	–	–	M	1.0	Contains the response of the ES.
	value	xsd:boolean	M	1.0	Contains the indication of success

F.3.6.3 Example of a TAF

The following XML code shows the content of an exemplary TAF file.

```

<?xml version="1.0" encoding="UTF-8"?>
<ReturnInterfaceResponse xmlns="http://www.io-link.com/DTI/2017/02/TBF"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:prim="http://www.io-
  link.com/DTI/2017/02/Primitives" xsi:schemaLocation="http://www.io-link.com/DTI/2017/02/TBF
  IOsafe TBF Schema 20170225.xsd">
  <Response value="true"/>
</ReturnInterfaceResponse>

```

F.3.7 Invocation behavior

F.3.7.1 Conventions on Device Tool invocation

Since the directory path of the TPF can contain "blank" characters, the Device Tool shall use the double quote character (") at the beginning and the end of the string when the ".exe" file is invoked.

It is not required for the invoking Master Tool to monitor the status of the launched Device Tools. Even in case an instance of a Device Tool is already running, the Master Tool will generate a new Device Tool invocation whenever the user launches the same tool again.

Therefore, it is the task of the Device Tool to handle multiple invocations. Table F.5 lists invocation cases and possible behaviors.

Table F.5 – Invocation cases and behaviors

Case	Behavior
Device Tool is launched once	No conflicts
Device Tool is already running and works on the same Device instance as in a prior session.	– The Tool should be brought to the foreground of the GUI desktop – Invocation of another instance of the Device Tool shall be avoided
Device Tool is already running and works on another Device instance as provided by the DTI call. The provided DeviceReference is <i>known</i> in the Device Tool.	The behavior depends on the design of the Device Tool: – Another Tool instance is launched and opens its Device data – The active GUI is brought to the foreground of the desktop in order to show the Device data of the selected Device
Device Tool is already running and works on another Device instance as provided by the DTI call. The provided DeviceReference is <i>not known</i> in the Device Tool.	The behavior depends on the design of the Device Tool: – Another Tool instance is launched and creates a new Device instance – The active GUI is brought to the foreground of the desktop in order to create a new Device instance of the selected Device

If a Device Tool is invoked via DTI, this Tool should not call another Device Tool because the Communication Server cannot interconnect (no nested communication defined for a DTI Communication Server).

F.3.7.2 Handling of the TPF

The name of the TPF will be provided to the Device Tool as a command line parameter. This name shall consist of a drive letter, an absolute path expression and the file extension. Alternatively, the UNC notation can be used instead of the drive letter. The Master Tool is responsible to create the file and unlock it before the Device Tool is invoked in such a manner that the Device Tool has full access to the file. The file name itself is only temporary and a new file name is generated with each Tool invocation.

After interpretation of the content of the TPF file, the Device Tool shall delete this file. Since the Master Tool can also delete this file when it is restarted, it is recommended for the Device Tool to make a "private" copy of the file when the Device Tool is launched.

F.4 Device data objects (DDO)

F.4.1 General

There are scenarios, where parameter instance data for a certain project are created off-site, for example via a "USB Master" tool enhanced for SDCI-FS, which can be mailed to a facility where they are imported into the engineering tool of the overall automation project. These data are called Device Data Objects (DDO). Figure F.6 shows such a scenario. In essence, DDO are used to export/import parameter instance data of Devices from one Master Tool to the other.

A TPF usually contains already those instance data. Thus, in principle, a DDO consists of TPF data and the associated IODD for consistency reasons.

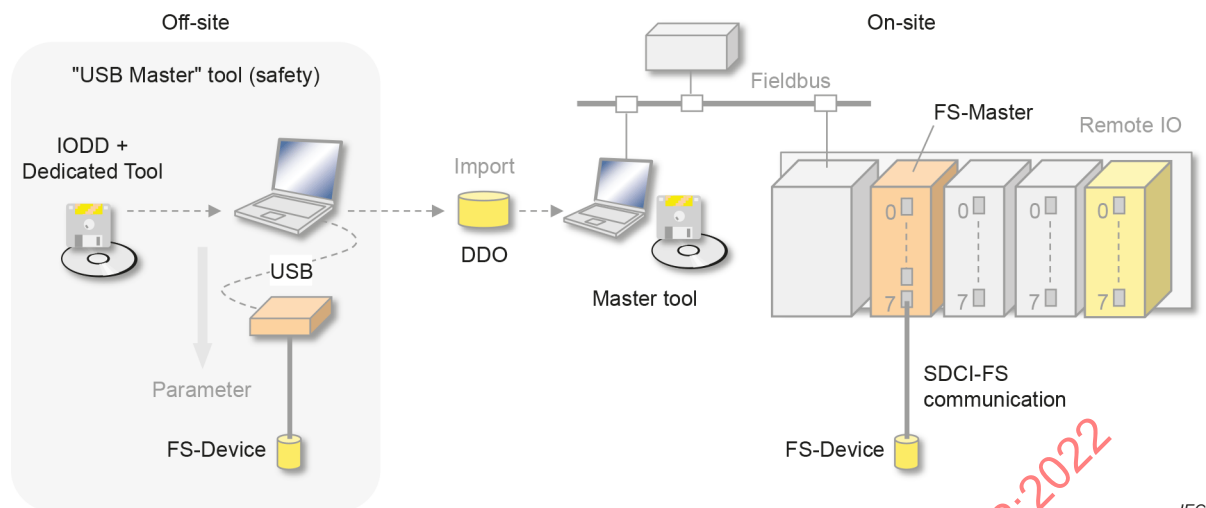


Figure F.6 – Purpose of Device data objects (DDO)

F.4.2 Structure of DDO package

A Master Tool creates a DDO package as zip archive consisting of the TPF as defined in F.3.4 and the corresponding IODD together with its associated files, such as icons, pictures, and alike (see [17]). The archive file name extension shall be "ioddo". The name of the archive file can be chosen freely.

F.5 Communication Interface

F.5.1 General

As already explained in Clause F.1, there is no seamless communication solution for stand-alone Device Tools such as "Dedicated Tools" for functional safety in SDCI so far. The only possibility in the past has been a separate point-to-point communication connection, for example RS232, USB, or alike, between a Device and a PC running the Device Tool software. Each of these connections requires appropriate driver software with different programming API for the Device and for the different PC communication interfaces.

This leads to the problem that a Device Tool either can work only with one particular communication interface or that the Device Tool has to implement different APIs for Device driver integration.

Another problem in a plant is that the network structure often requires communication across network boundaries (Routing). Due to the many fieldbuses and different communication protocols, it is very cumbersome to achieve an integrated network with routing functions for Device Tools down to the associated Device (see Figure F.7).

The second major part of DTI solves two problems:

- All Devices/FS-Devices and their Device Tools/Dedicated Tools can rely on one particular communication interface.
- The chosen communication technology is standardized in [19] and solves the routing problem across network boundaries.

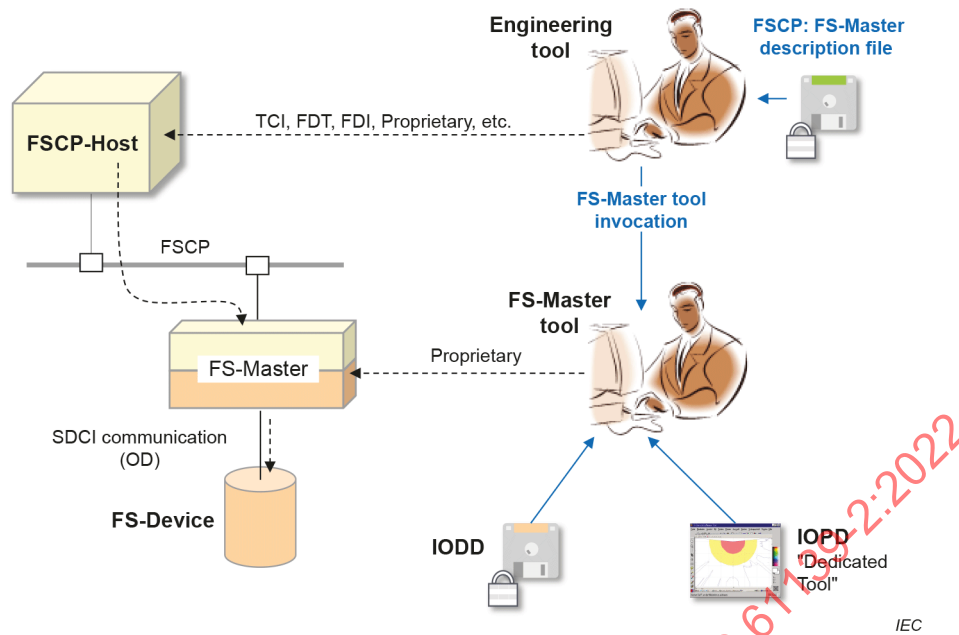


Figure F.7 – Communication routes between Device Tool and Device

F.5.2 Principle of DTI communications

The communication interface consists of a component which provides a unique interface (API) to the Device Tool. This component can provide communication functionality for different field busses and proprietary network protocols. The communication parameters which are necessary to establish a connection are entered in the Master Tool and passed to the Device Tool when it is launched.

Figure F.7 shows fieldbus or proprietary networks between the PC and the Device. Figure F.8 shows the mapping to software and Communication Servers.

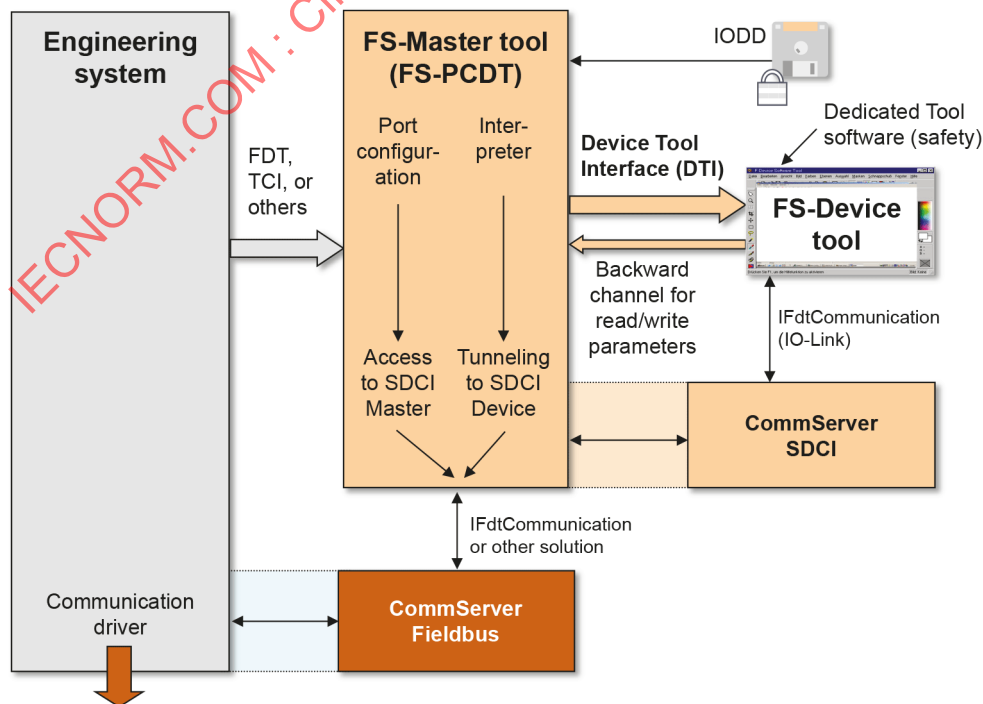


Figure F.8 – Routing across networks and SDCI

In this case, the Communication Server (Fieldbus) requires information about the network protocol. This routing information is generated by the Engineering System and transferred to the Communication Server (Fieldbus). Since manufacturer specific data is exchanged, the Communication Server and the Engineering System shall be provided by the same manufacturer.

The routing information for the second Communication Server (SDCI) is generated by the Master Tool and transferred to this CS. When the Device Tool is started, only a communication reference to the Device is passed. This reference is forwarded from the Device Tool to the Communication Server. With the help of the routing information from the Engineering System, the Communication Server can create physical network addresses and establish a connection to the Device. Figure F.9 shows the relationships between the components involved.

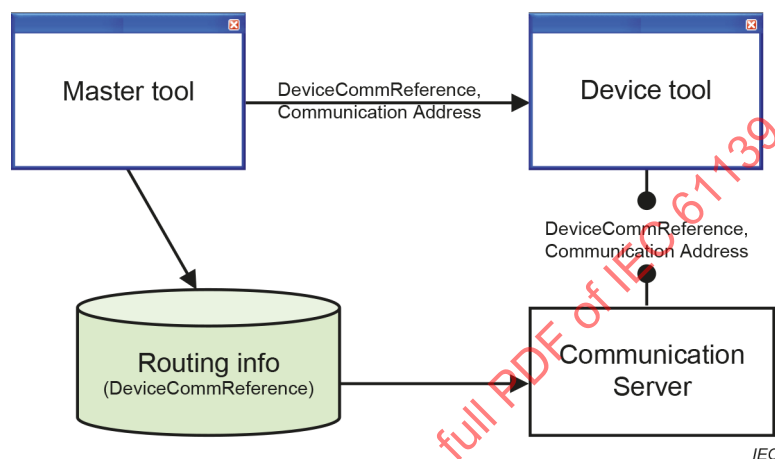


Figure F.9 – Communication Server

It is always possible for a Device Tool to use its native communication interfaces (for example serial RS232) as an alternative besides the Communication Server.

F.5.3 Gateways

A Communication Server allows a communication connection across network boundaries (see Figure F.8).

The Engineering System, all Device Tools and the Communication Server are located on the same PC which is connected e.g. via an Ethernet adapter to a network. The target Devices can be found behind a gateway which can work in different ways. From the Device Tool point of view, it is irrelevant where the Device is located because the network structure is handled by the Communication Server.

The Communication Server is potentially able to manage all gateway types which are supported by the Engineering System itself. The gateway functionalities itself are encapsulated by the Communication Server. Only gateway types known by the Communication Server can be supported (no nested communication).

If a Device can be reached through multiple paths in the network, it is up to the Engineering System to decide, which network path is used for communication.

F.5.4 Configuration of the Communication Server

In order to build the network communication addresses from the Device communication reference, the Communication Server requires configuration data from the Engineering System/Master Tool. The structure of configuration data itself and the way how the data is sent to the Communication Server is manufacturer specific and will not be standardized.

F.5.5 Definition of the Communication Interface

The Communication Server implements the interface "IFdtCommunication" and uses the "IFdtCommunication-Events" and "IFdtCommunicationEvents2" as described in [19]. All other DTM interfaces which are described in [19] are not relevant for the Communication Server. Due to this constraint, a Communication Server cannot be used in an FDT environment as communication DTM.

F.5.6 Sequence for establishing a communication relation

An interaction of Engineering System/Master Tool, Device Tool and Communication Server (CS) is required to establish a communication relation. The sequence is as follows:

At first, a Device is integrated into the Master Tool with the corresponding configuration file (IODD). Within the Engineering System, communication addresses and bus parameter are adjusted. Together with other network data, topology data for the network is the result.

Furthermore, the Master Tool shall build a unique Device communication reference. This reference is passed to the Device Tool when it is launched with the help of the TPF (keyword "DeviceCommReference"). The Device Tool is now able to establish a connection to the Device using the Communication Server and Device communication reference.

The Communication Server itself interprets the Device communication reference and converts it to network addresses. Therefore, it uses the configuration data from the Master Tool. Because it is up to the CS to decide if the Device communication reference or the communication address itself is used, the Device Tool shall always pass both attributes in the ConnectRequest XML document. If no routing functionality is required, the CS does not require the proprietary configuration. In order to connect, the CS can use the communication address itself from the Master Tool. Figure F.10 shows how a communication connection is established.

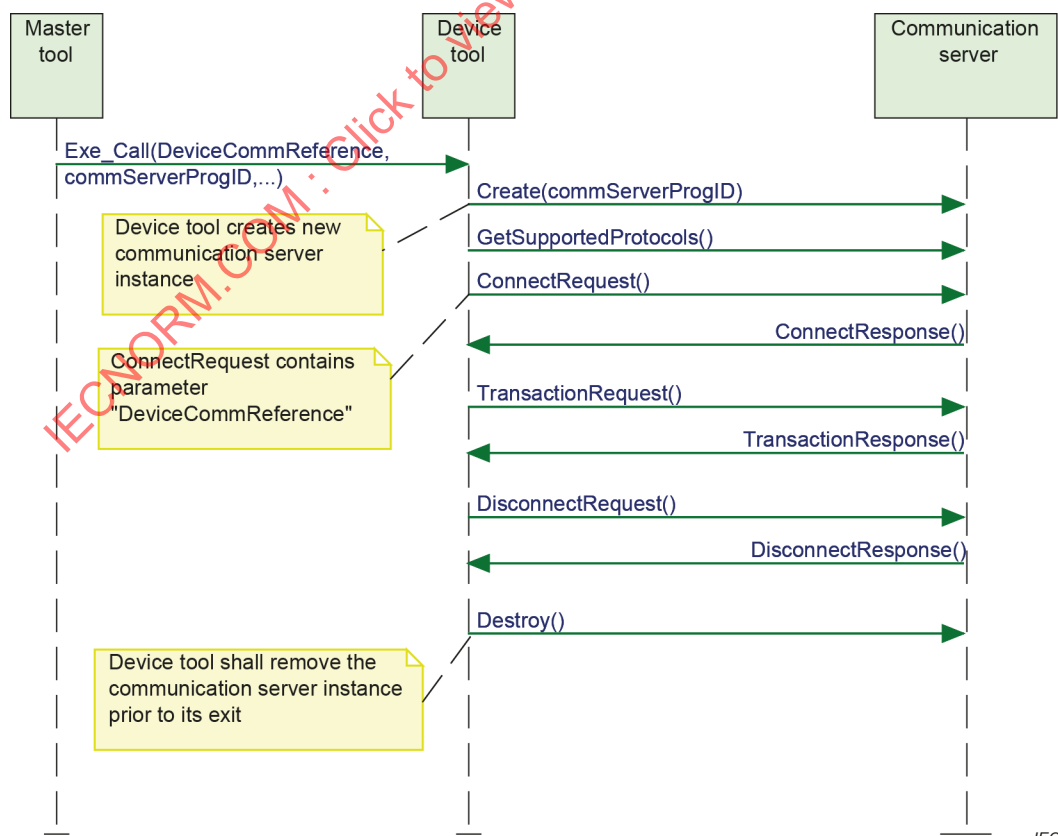
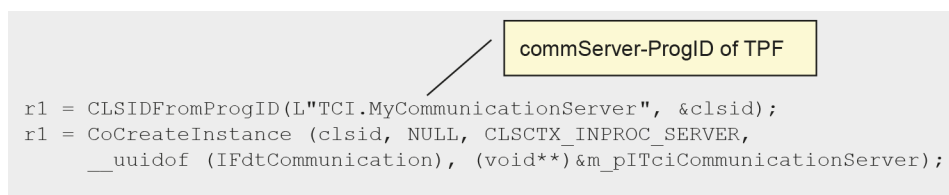


Figure F.10 – Sequence chart for establishing communication

The passed ProgID (Keyword commServer-ProgID) can be used to create a new instance of the Communication Server by the Device Tool. There is a 1:1 relationship between Device Tool and Communication Server instance. The Communication Server instance can connect to one or more Devices.

Figure F.11 shows a code fragment in C++ as an example on how to create a new instance.



```

r1 = CLSIDFromProgID(L"TCI.MyCommunicationServer", &clsid);
r1 = CoCreateInstance (clsid, NULL, CLSCTX_INPROC_SERVER,
    __uuidof (IFdtCommunication), (void**)&m_pITciCommunicationServer);
  
```

Figure F.11 – Create Communication Server instance

It is recommended to create the Communication Server instance as "in process server" (CLSCTX_INPROC_SERVER) due to performance issues.

After a new instance of the Communication Server is created, all methods of the interface "IFdtCommunication" can be called. At first a Device Tool shall call the "GetSupportedProtocols" method to find out if the required protocol is supported by the CS. If not, the Device Tool shall inform the user. A new connection is established with help of the function "ConnectRequest". Among others, as invocation parameter a pointer to the callback interface (Interface IFdtCommunicationEvents) is passed. This means that a Device Tool shall implement this interface.

The Device Tool is responsible to release the Communication Server instance when the Tool exits. If the Communication Server instance was created in the process of the Device Tool, as recommended before, this is done automatically since the instance is terminated with the process of the Device Tool.

F.5.7 Usage of the Communication Server in stand-alone mode

If a Device Tool is not called from a Master Tool with DTI, it shall find out the ProgID of the Communication Server by itself. In this case the "Component Categories" of the system registry can be used (HKEY_CLASSES_ROOT\Component Categories).

The following values are defined for the DTI Communication Server:

Symbolic Name of CatID: **CATID_DTI_CS**

UUID of CatID: **{7DDC60A6-1FD4-45a2-917F-0F8FC371BC57}**

A Device Tool can find out the ProgID of the Communication Server with the help of the Standard Component Categories Manager. If more than one component is assigned to this category, the user of the Device Tool shall select one of the Communication Servers.

If a Communication Server does not support the "Stand-Alone" mode (i.e. a Communication Server instance cannot be created by a Device Tool), a system registry entry should not be made.

A Device Tool that supports Conformance Class 3 and is intended for "Stand-Alone" mode shall store the DeviceCommReferences together with its DDOs. Whenever the DeviceComm-Reference is changed by the Master Tool while copying the entire project or while retrieving the project, the Device Tool shall check and – if changed – update the DeviceCommReference when called from the Master Tool with DTI. There are two general possibilities:

- 1) The Device Tool checks and updates the DeviceCommReference of a particular Device immediately before connection. Hint: After copy/retrieval of a Master Tool project, the user should call the Device Tool via DTI and connect to the particular Device(s) prior to the connection to this/these Device(s) later on in "Stand-Alone" mode.
- 2) The Device Tool checks and updates the DeviceCommReferences of all Devices immediately after being called by the Master Tool via DTI. Hint: After copy/retrieval of a Master Tool project, the user should call the Device Tool via DTI. Then, all Devices can be connected later in "Stand-Alone" mode.

F.5.8 SDCI specifics

The SDCI schema defined in [20] shall be used as communication schema.

Table F.6 shows the mapping between the TPF keywords and the attributes in the communication schema.

Table F.6 – Communication Schema mapping

Attribute of ConnectRequest element (FDTIOLinkCommunicationSchema.xml)	Parameter Keyword in TPF file	Remarks
fdt:nodeId	–	Unused
systemTag	"DeviceCommReference" attribute of element "Device".	

The communication parameters passed during the Device Tool invocation shall be used as input for the Connect Request XML document to be used in the connect method. Additionally, the device communication reference (Keyword "DeviceCommReference" in Table F.6) shall be entered in the Connect Request XML document as attribute "systemTag". Figure F.12 shows an example.

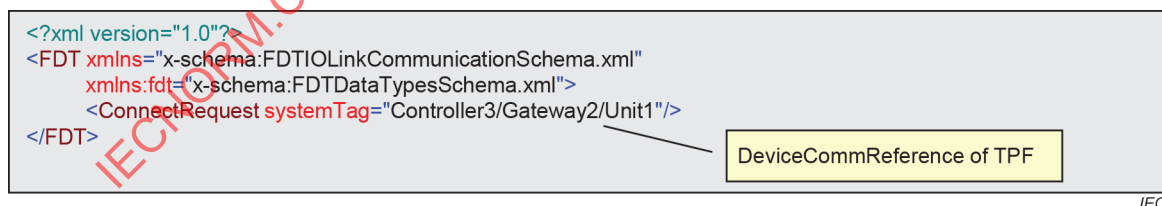


Figure F.12 – Example of a Connect Request XML document for SDCI

F.5.9 Changing communication settings

If it is necessary to change the communication address (Master, Port) in the Master Tool, the Device Tool needs information about the new communication address. This shall be done via relaunching the Device Tool by the user of the Master Tool. During relaunch, the new communication parameters are passed to the Device Tool. With these communication parameters a new communication relation can be established to the Device.

If the Device communication reference is used instead of the communication address between Device Tool and Communication Server, no relaunch of the Tool is required, because the Device communication reference does not change whenever the communications address changes. In this case, the Communication Server itself can reconnect to the Device with the new communication address (Master, Port).

For an existing connection, changed communication parameters in the Master Tool project shall not have any impact. Changed communication parameters shall be used when a connection is (re)established.

F.6 Reaction on incorrect Tool behavior

Table F.7 describes the system reaction if a Master Tool or Device Tool works incorrectly.

Table F.7 – Reaction on incorrect Tool behavior

Fault	Description	System reaction
XML structure of PID file not valid	The PID file of a Device Tool does not validate with the XML Schema in F.9.1	The Master Tool should only show an error message if required schema elements or attributes are missing. All unknown elements or attributes shall be ignored.
XML structure of TPF file not valid	The TPF file generated by the Master Tool does not validate with the XML Schema in F.9.3	The Device Tool should only show an error message if required schema elements or attributes are missing. All unknown elements or attributes shall be ignored.
Device Tool cannot be invoked	When the operation system is instructed to create a new process (tool invocation) the function returns an error code. Reason could be that the path of the exe file in the system registry is incorrect.	Master Tool shall show an error message (tool cannot be invoked) with the name and path of the exe file.
CommunicationServer object cannot be created. See F.5.6	The "CoCreateInstance" function returns an error code when an object with the ProgID of the TPF should be instantiated.	The Device Tool should show an error message.
TPF file not deleted by the Device Tool	The TPF file was not removed by the Device Tool as described in F.3.1	Master Tool should delete the TPF file when it is launched (garbage collection). If the file cannot be deleted, the Master Tool should not show an error message.
DeviceCommReference not valid (Communication channel cannot be established). See Clause F.5.	Device Tool is using a not existing DeviceCommReference in the Master Tool.	The Device Tool should show an error message.

F.7 Compatibility

F.7.1 Schema validation

XML documents can easily be validated with the help of standard parsers and schema files. If the structure of an XML document does not follow the rules defined in the corresponding schema, the XML parser rejects the document. This is not very practical if Tools with different versions of DTI files shall work together since a newer XML document cannot be processed by previous software.

To implement a robust model, the Master Tool and the Device Tools shall ignore any XML attributes or elements not recognizable in a valid XML document. This means that XML schema validation shall not be used. The schema files in Clause F.9 are for information purposes only.

The installation program of the Device Tool can always install the newest PID file version. The Master Tool shall ignore any unknown XML attributes or elements.

F.7.2 Version policy

If it is necessary to modify the structure definition of a TPF with the result that a new version of the invocation interface is defined, the Master Tool shall ensure that the right version of the TPF is created. That means it shall use an earlier version of the structure if the Device Tool is only able to support the earlier version.

The PID file version of the Device Tool determines the newest supported version of the corresponding Device Tool. See F.3.3 for details.

If a Device Tool supports a newer version than the Master Tool, the Master Tool uses its newest TPF version. In this case the Device Tool shall work with the old schema version.

F.8 Scalability

F.8.1 Scalability of a Device Tool

The manufacturer of a Device Tool can choose to support different function levels of DTI as shown in Table F.8.

Table F.8 – DTI conformance classes

Conformance Class	Description
C1 (Navigation)	Setup program creates system registry entries as described in F.3.2. This allows the user to invoke the Device Tool from the context of a selected Device in the Master Tool without any impact on an existing Device Tool itself.
C2 (Parameter transfer)	The Device Tool uses the information of the TPF. In this case, for example, the Tool can read FST parameter instances or use a communication address for its proprietary communication channel. This way, the user can be relieved from multiple entries. The implementation effort is limited to evaluation of the TPF file for internal initialization of the Device Tool.
C3 (DTI communication with optional backchannel)	The full functionality is available if the Device Tool uses the DTI Communication Server. This component enables the Tool to manage all network boundaries implemented by the Master Tool. In this case the Device Tool shall support the IFdtCommunication/IFdtCommunicationEvents/IFdtCommunicationEvents2 interface. In case of the backchannel option, the Master Tool uses the information of the TBF. In this case, for example, the Tool can read FST parameter instances or use the I/O Process Data description. This way, the user can be relieved from multiple entries. The implementation effort is limited to evaluation of the TBF file for internal processing of the Master Tool.

Table F.9 shows the DTI relevant features of a Device Tool.

Table F.9 – DTI feature levels of Device Tools

Function	Subclause	Conformance Class	Feature Name for PID file
Make system registry entries	F.3.2	C1	–
Provide PID file during installation procedure	F.3.3	C1	–
Avoid multiple program instances		C2	–
Interpret TPF	F.3.4	C2	–
Delete TPF	F.3.7.2	C2	–
Use the Communication Server interface		C3	–

F.8.2 Scalability of a Master Tool

A Master Tool shall support all DTI feature levels/conformance classes.

F.8.3 Interactions at conformance class combinations

Table F.10 defines how a Master Tool and a Device Tool shall interact depending on their conformance class.

Table F.10 – Interactions at conformance class combinations

Master Tool	Device Tool	Interaction
C2 or C3	C1	Device Tool is launched, no parameters are passed. The Master shall not generate a TPF because it would not be deleted by the Device Tool.
C2 or C3	C2	Device Tool is launched, Parameters are passed through TPF.
C2	C3	Device Tool is launched, Parameters are passed through TPF.
C3	C3	Device Tool is launched, Parameters are passed through TPF. Communication via Communication Server is possible.

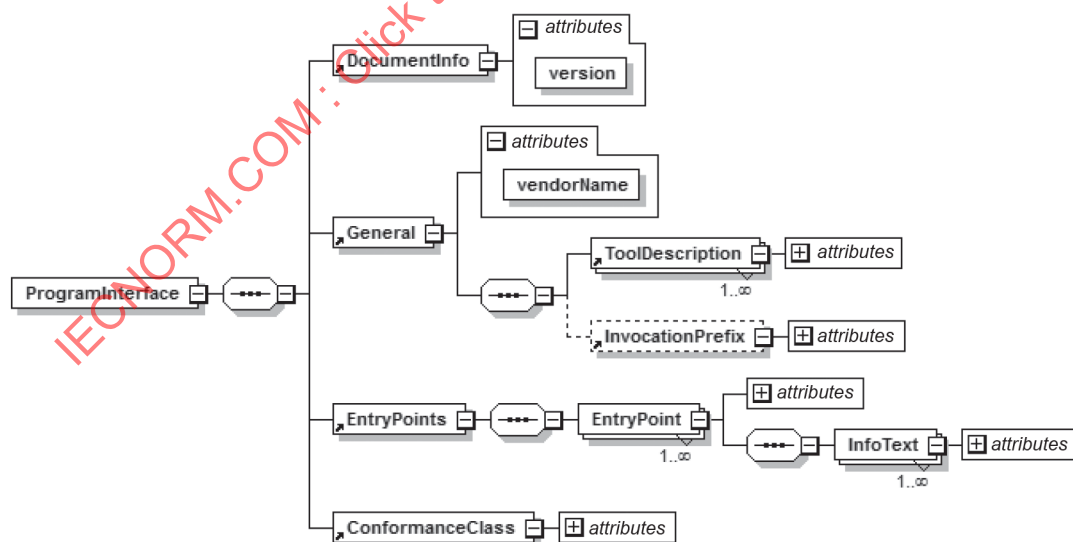
F.9 Schema definitions

F.9.1 General

The schema definitions in this Clause F.9 are for information only (see F.7.1).

F.9.2 Schema of the PID

Figure F.13 shows the XML schema of the Program Interface Description file.



IEC

Figure F.13 – XML schema of the PID file

Figure F.13 is based on the XML code as follows:

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns="http://www.io-link.com/DTI/2017/02/PID" xmlns:prim="http://www.io-
link.com/DTI/2017/02/Primitives" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.io-link.com/DTI/2017/02/PID" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0">
  <xsd:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xsd:import namespace="http://www.io-link.com/DTI/2017/02/Primitives" schemaLocation="DTI-
Primitives1.0.xsd"/>
  <xsd:element name="DocumentInfo">
    <xsd:complexType>
      <xsd:attribute name="version" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:pattern value="V\d+(\.\d+){1,7}" />
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="ToolDescription">
    <xsd:complexType>
      <xsd:attribute name="lang" type="xsd:string" use="required"/>
      <xsd:attribute name="name" type="xsd:string" use="required"/>
      <xsd:attribute name="description" type="xsd:string" use="required"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="InvocationPrefix">
    <xsd:complexType>
      <xsd:attribute name="name" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string"/>
        </xsd:simpleType>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="General">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ToolDescription" maxOccurs="unbounded"/>
        <xsd:element ref="InvocationPrefix" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="vendorName" type="xsd:string" use="required"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="EntryPoints">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="EntryPoint" maxOccurs="unbounded">
          <xsd:complexType>
            <xsd:complexContent>
              <xsd:extension base="prim:ObjectT">
                <xsd:sequence>
                  <xsd:element name="InfoText" maxOccurs="unbounded">
                    <xsd:complexType>
                      <xsd:attribute name="lang" type="xsd:string" use="required"/>
                      <xsd:attribute name="name" type="xsd:string" use="required"/>
                      <xsd:attribute name="description" type="xsd:string" use="required"/>
                    </xsd:complexType>
                  </xsd:element>
                </xsd:sequence>
                <xsd:attribute name="id" type="prim:IdT" use="required"/>
              </xsd:extension>
            </xsd:complexContent>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="ConformanceClass">
    <xsd:complexType>
      <xsd:attribute name="name" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="C1"/>
            <xsd:enumeration value="C2"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>

```

```

        <xsd:enumeration value="C3"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="ProgramInterface">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="DocumentInfo"/>
      <xsd:element ref="General"/>
      <xsd:element ref="EntryPoints"/>
      <xsd:element ref="ConformanceClass"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>

```

F.9.3 Schema of the TPF

Figure F.14 shows the XML schema of the Temporary Parameter File.

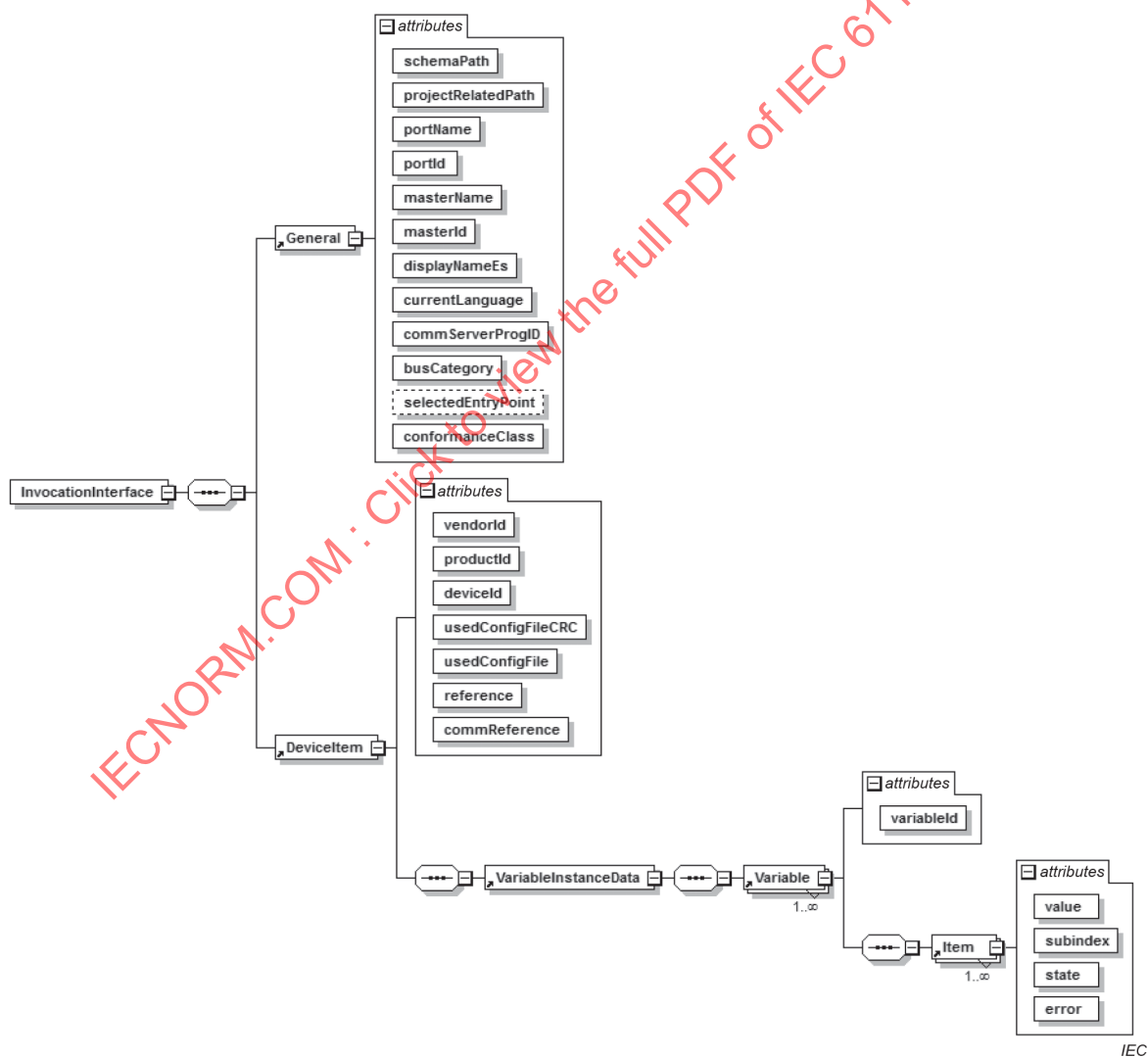


Figure F.14 – XML schema of the TPF

Figure F.14 is based on the XML code as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns="http://www.io-link.com/DTI/2017/02/TPF" xmlns:prim="http://www.io-link.com/DTI/2017/02/Primitives" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.io-link.com/DTI/2017/02/TPF" elementFormDefault="qualified"
  attributeFormDefault="unqualified" version="1.0">
  <xsd:import namespace="http://www.io-link.com/DTI/2017/02/Primitives" schemaLocation="DTI-Primitives1.0.xsd"/>
  <xsd:element name="VariableInstanceData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Variable" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Variable">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Item" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="variableId" type="prim:IdT" use="required"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Item">
    <xsd:complexType>
      <xsd:attribute name="value" type="xsd:string" use="required"/>
      <xsd:attribute name="subindex" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:unsignedShort">
            <xsd:maxInclusive value="255"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="state" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="empty"/>
            <xsd:enumeration value="initial"/>
            <xsd:enumeration value="device"/>
            <xsd:enumeration value="read error"/>
            <xsd:enumeration value="write error"/>
            <xsd:enumeration value="valid"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="error" type="xsd:integer" use="required"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="InvocationInterface">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="General"/>
        <xsd:element ref="DeviceItem"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="General">
    <xsd:complexType>
      <xsd:attribute name="schemaPath" type="xsd:string" use="required"/>
      <xsd:attribute name="projectRelatedPath" type="xsd:string" use="required"/>
      <xsd:attribute name="portName" type="xsd:string" use="required"/>
      <xsd:attribute name="portId" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:unsignedShort">
            <xsd:maxInclusive value="255"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="masterName" type="xsd:string" use="required"/>
      <xsd:attribute name="masterId" type="xsd:int" use="required"/>
      <xsd:attribute name="displayNameEs" type="xsd:string" use="required"/>
      <xsd:attribute name="currentLanguage" type="xsd:string" use="required"/>
      <xsd:attribute name="commServerProgID" type="xsd:string" use="required"/>
      <xsd:attribute name="busCategory" type="xsd:string" use="required"/>
      <xsd:attribute name="selectedEntryPoint" type="prim:IdT" use="optional"/>
      <xsd:attribute name="conformanceClass" use="required">
```

```

<xsd:simpleType>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="C1"/>
    <xsd:enumeration value="C2"/>
    <xsd:enumeration value="C3"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<!-- IOLink -->
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceItem">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="VariableInstanceData"/>
    </xsd:sequence>
    <xsd:attribute name="vendorId" type="xsd:unsignedShort" use="required"/>
    <xsd:attribute name="productId" type="xsd:string" use="required"/>
    <xsd:attribute name="deviceId" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:unsignedInt">
          <xsd:maxInclusive value="16777215"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="usedConfigFileCRC" type="xsd:int" use="required"/>
    <xsd:attribute name="usedConfigFile" type="xsd:string" use="required"/>
    <xsd:attribute name="reference" type="xsd:string" use="required"/>
    <xsd:attribute name="commReference" type="xsd:string" use="required"/>
  </xsd:complexType>
</xsd:element>
</xsd:schema>

```

F.9.4 Schema of the TBF

Figure F.15 shows the XML schema of the Temporary Backchannel File.



Figure F.15 – XML schema of a TBF

Figure F.15 is based on the XML code as follows:

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns="http://www.io-link.com/DTI/2017/02/TBF" xmlns:prim="http://www.io-link.com/DTI/2017/02/Primitives" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.io-link.com/DTI/2017/02/TBF">
  <xsd:import namespace="http://www.io-link.com/DTI/2017/02/Primitives" schemaLocation="DTI-Primitives1.0.xsd"/>
  <xsd:element name="VariableInstanceData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Variable" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Variable">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Item" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

```



```

    </xsd:sequence>
    <xsd:attribute name="variableId" type="xsd:string" use="required"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Item">
  <xsd:complexType>
    <xsd:attribute name="value" type="xsd:string" use="required"/>
    <xsd:attribute name="subindex" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:unsignedShort">
          <xsd:maxInclusive value="255"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="state" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="empty"/>
          <xsd:enumeration value="initial"/>
          <xsd:enumeration value="device"/>
          <xsd:enumeration value="read error"/>
          <xsd:enumeration value="write error"/>
          <xsd:enumeration value="valid"/>
          <!--xsd:enumeration value="changed"/-->
          <!-- should be transferred to device or stored in database before DTI invocation -->
          <!-- could be changed to empty before DTI invocation -->
          <!-- could be changed to empty or valid before DTI invocation -->
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="error" type="xsd:integer" use="required"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Response">
  <xsd:complexType>
    <xsd:attribute name="value" type="xsd:boolean" use="required"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ReturnInterfaceRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="VariableInstanceData"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ReturnInterfaceResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="Response"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:group name="ReturnInterface">
  <xsd:choice>
    <xsd:element ref="ReturnInterfaceRequest"/>
    <xsd:element ref="ReturnInterfaceResponse"/>
  </xsd:choice>
</xsd:group>
</xsd:schema>

```

F.9.5 Schema of the TAF

The schema of the TAF corresponds to the schema of the TBF in F.9.4.

F.9.6 Schema of DTI primitives

The DTI primitives are defined in the XML code as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns="http://www.io-link.com/DTI/2017/02/Primitives"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.io-link.com/DTI/2017/02/Primitives">
  <xsd:annotation>
    <xsd:documentation>In this schema, only the necessary types and attributes for DTI are used
from the Common Primitives Schema.</xsd:documentation>
    <xsd:appinfo>
      <schemainfo versiondate="20170225"/>
    </xsd:appinfo>
  </xsd:annotation>
  <!-- SIMPLE TYPES -->
  <xsd:simpleType name="IdT">
    <xsd:annotation>
      <xsd:documentation>Base Type for Object identifiers</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string"/>
  </xsd:simpleType>
  <xsd:simpleType name="GuidT">
    <xsd:annotation>
      <xsd:documentation>GUID</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="\{ [0-9A-Fa-f]{8}\-[0-9A-Fa-f]{4}\-[0-9A-Fa-f]{4}\-[0-9A-Fa-f]{4}\-[0-9A-Fa-f]{12}\}\"/>
      <xsd:pattern value="[0-9A-Fa-f]{8}\-[0-9A-Fa-f]{4}\-[0-9A-Fa-f]{4}\-[0-9A-Fa-f]{4}\-[0-9A-Fa-f]{12}\}\"/>
    </xsd:restriction>
  </xsd:simpleType>
  <!-- COMPLEX TYPES -->
  <!-- Main Types -->
  <xsd:complexType name="DocumentT">
    <xsd:annotation>
      <xsd:documentation>Type for all top level elements</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DocumentInfo" type="DocumentInfoT"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="DocumentInfoT">
    <xsd:attribute name="Version" type="xsd:string" use="required" fixed="1.1"/>
  </xsd:complexType>
  <!-- ELEMENT DECLARATIONS -->
  <!-- Text Definition Elements-->
  <xsd:complexType name="ObjectT">
    <xsd:annotation>
      <xsd:documentation>Base type</xsd:documentation>
    </xsd:annotation>
  </xsd:complexType>
  <xsd:complexType name="FeatureT">
    <xsd:annotation>
      <xsd:documentation>Base type</xsd:documentation>
    </xsd:annotation>
    <xsd:attribute name="Name" type="xsd:string" use="optional"/>
  </xsd:complexType>
  <xsd:complexType name="ParameterT" mixed="true">
    <xsd:attribute name="Name" type="xsd:string" use="required"/>
  </xsd:complexType>
  <!-- Specialized Parameters-->
  <xsd:complexType name="StringParameterT">
    <xsd:complexContent>
      <xsd:extension base="ParameterT">
        <xsd:attribute name="Value" type="xsd:string" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <!-- ELEMENT DECLARATIONS -->
  <xsd:element name="Document" type="DocumentT">
    <xsd:unique name="OBJ-ID">
      <xsd:selector xpath="//*[@*]">

```

```
<xsd:field xpath="@ID"/>
</xsd:unique>
</xsd:element>
<xsd:element name="Object" type="ObjectT"/>
<xsd:element name="Parameter" type="ParameterT"/>
<xsd:element name="StringParameter" type="StringParameterT" substitutionGroup="Parameter"/>
<xsd:element name="Feature" type="FeatureT"/>
<xsd:simpleType name="ConformanceClassEnumT">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="C1"/>
    <xsd:enumeration value="C2"/>
    <xsd:enumeration value="C3"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>
```

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

Annex G (normative)

Main scenarios of SDCI-FS

G.1 Overview

Table G.1 shows main scenarios, the initial key parameters, and the associated system activities. Its purpose is to provide a brief overview and it contains references to clauses with detailed descriptions.

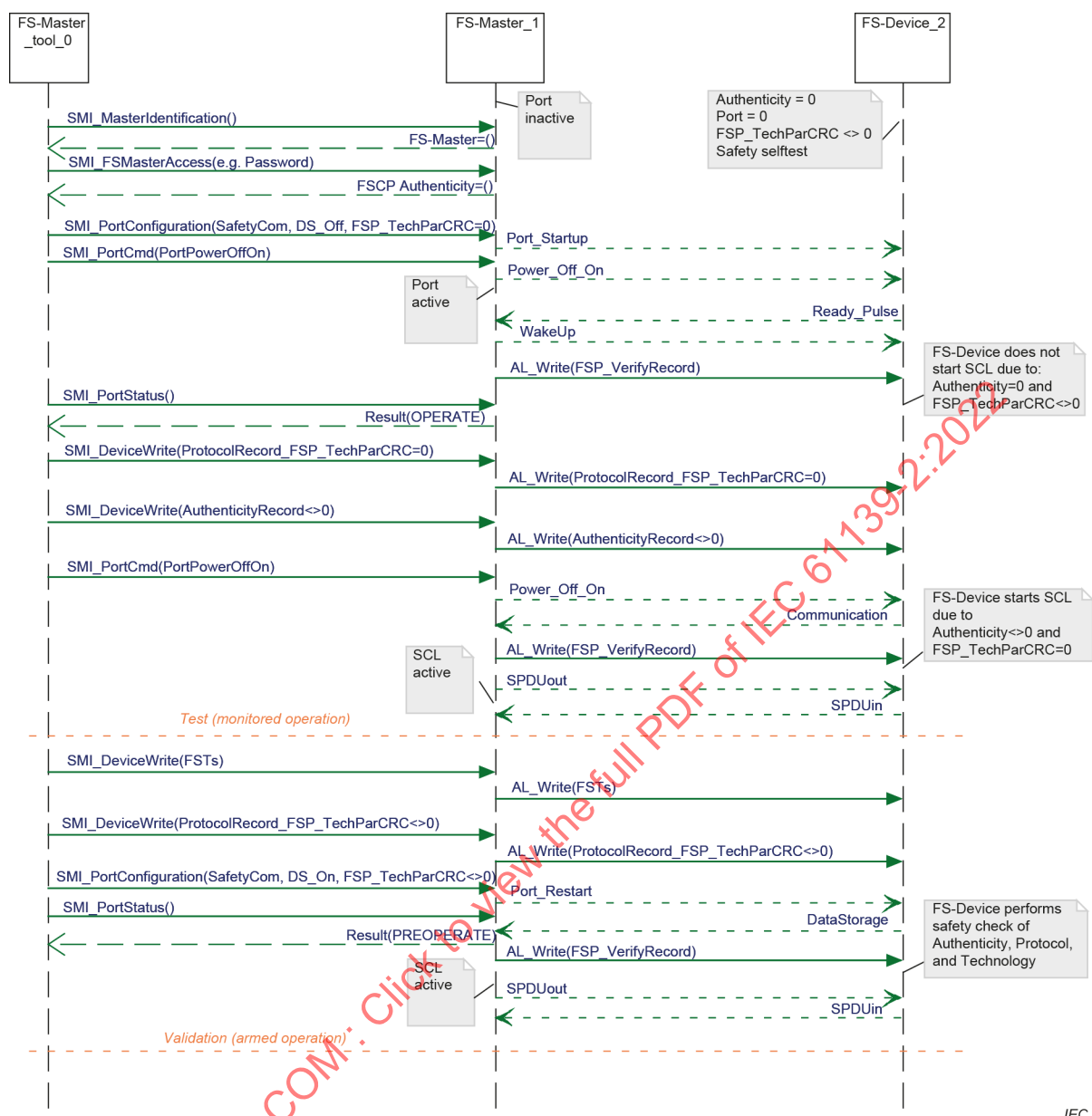
Table G.1 – Main scenarios of SDCI-FS

Scenario	Initial parameters	System activities
OSSD operation (on FS-DI or FS-Master)	Authenticity = 0 Port = 0 FSP_TechParCRC ≠ 0 (factory settings)	1. Modify FST parameter via "USB Master" tool (option; see off-site commissioning in IEC 61131-9) and IODD 2. Adapt FSP_TechParCRC (see 11.7.8) using "Dedicated Tool" 3. FS-Device evaluates validity of technology parameters (FST) via FSP_TechParCRC at STARTUP 4. Plug, validate & play (default)
Commissioning (Test = monitored operation) See Figure G.1	Authenticity = 0 Port = 0 FSP_TechParCRC ≠ 0 (factory settings)	1. Set FSP_TechParCRC = 0 temporarily (FS-Device and FSP_Verify-Record) via FS-Master Tool 2. Assign Authenticity and Port to FS-Device via FS-Master tool and via Authenticity record 3. Assign FSP parameter via FS-Master tool and assign FSP_VerifyRecord to FS-Master (see 9.4.2 "Local modifications") 4. PowerOFF/ON FS-Device (reset) for a duration of at least the time defined by auxiliary parameter FSP_MinShutDownTime (see A.2.12) 5. FS-Master transfers FSP_VerifyRecord to FS-Device (see A.2.10 and 10.4.3.1) 6. Run in test mode (No verification except FSP_TechParCRC = "0"; Data Storage disabled) 7. FST parameter can be modified 8. FS-Master Tool and FS-Master are responsible to indicate test mode and/or to prevent from running in test mode w/o Tool connection. For this purpose, the FS-Master shall store the Port configuration with FSP_TechParCRC = "0" in a volatile manner. Thus, after restart of the FS-Master ("power cycle") w/o Tool, the FS-Device will not receive a valid Port configuration and will not perform safety communication.
Commissioning (Arm and validate) See Figure G.1	Authenticity = FSCP ("A-Code", NOTE) Port = port number FSP_TechParCRC = 0	1. Assign actual FSP_TechParCRC (FS-Device and FSP_VerifyRecord) via FS-Master Tool 2. Transfer FSP Parameter records to FS-Master, secured by FSP_ProtParCRC via FS-Master Tool (SMI service) 3. Port restart after port configuration (see IEC 61131-9) 4. Upload parameters to Data Storage (FSP and FST) in PREOPERATE, see "Backup/Restore" in IEC 61131-9 5. FS-Master transfers FSP_VerifyRecord to FS-Device (see A.2.10) 6. Run in armed mode (Verification: Authenticity + FSP_TechParCRC compared), see 11.7.6 7. Validation according to safety manual of FS-Device

Scenario	Initial parameters	System activities
Armed to Commissioning	Authenticity = FSCP ("A-Code", NOTE) Port = port number FSP_TechParCRC ≠ 0	1. Set FSP_TechParCRC = "0" 2. Follow activity 4. to activity 8. in scenario "Commissioning"
Replacement by FS-Device with factory settings w/o tools	Authenticity = 0 Port = 0 FSP_TechParCRC ≠ 0	1. Download and adopt parameters from Data Storage (FSP and FST) if Authenticity and Port = 0 (see IEC 61131-9:—, 12.5.1 and 12.5.2) 2. Run in armed mode (Verification: Authenticity + FSP_TechParCRC compared), see 11.7.6 and A.2.10 3. Validation according to safety manual of FS-Device.
Misconnection of configured FS-Devices	Authenticity = FSCP ("A-Code", NOTE) Port = port number FSP_TechParCRC ≠ 0	1. No adoption of downloaded parameters from Data Storage (FSP and FST) since Authenticity and Port ≠ "0" in FS-Device 2. SCL not started (Verification: Authenticity + FSP_TechParCRC compared), see 11.7.6 and A.2.10 3. Error message: "Misconnection" (0xB003 or 0xB004, see Annex B).
NOTE "A-Code" refers to IEC 61784-3:2021		

G.2 Sequence chart of commissioning

Sequence chart in Figure G.1 illustrates major activities during commissioning of an FS-Device with factory settings. First phase is the test phase of FS-Device and safety functions while in monitored operation by personnel. Second phase comprises arming of port and corresponding FS-Device as well as validation of the safety function according to safety manuals.



IEC

Figure G.1 – Commissioning with test and armed operation

G.3 Sequence chart of replacement

Sequence chart in Figure G.2 illustrates major activities after an FS-Device replacement by one with factory settings.

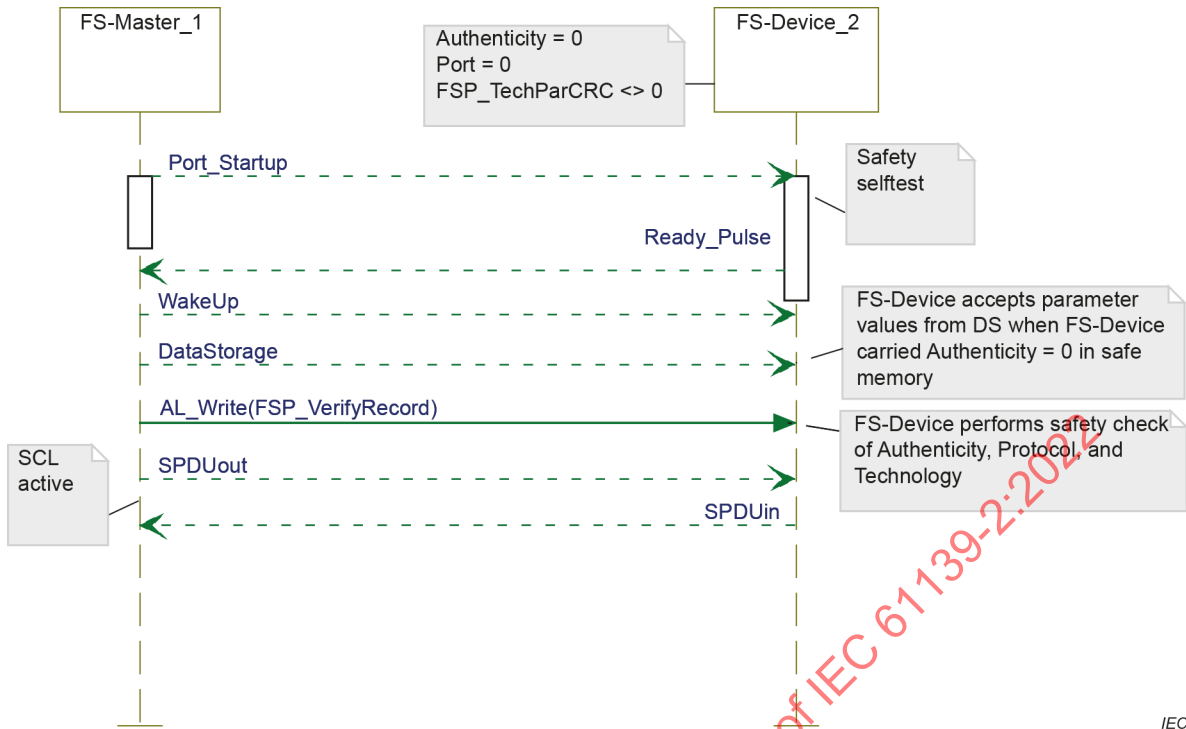


Figure G.2 – FS-Device replacement

G.4 Sequence chart of misconnection

Sequence chart in Figure G.3 illustrates major activities after an FS-Device replacement by one with other parameters than factory settings.

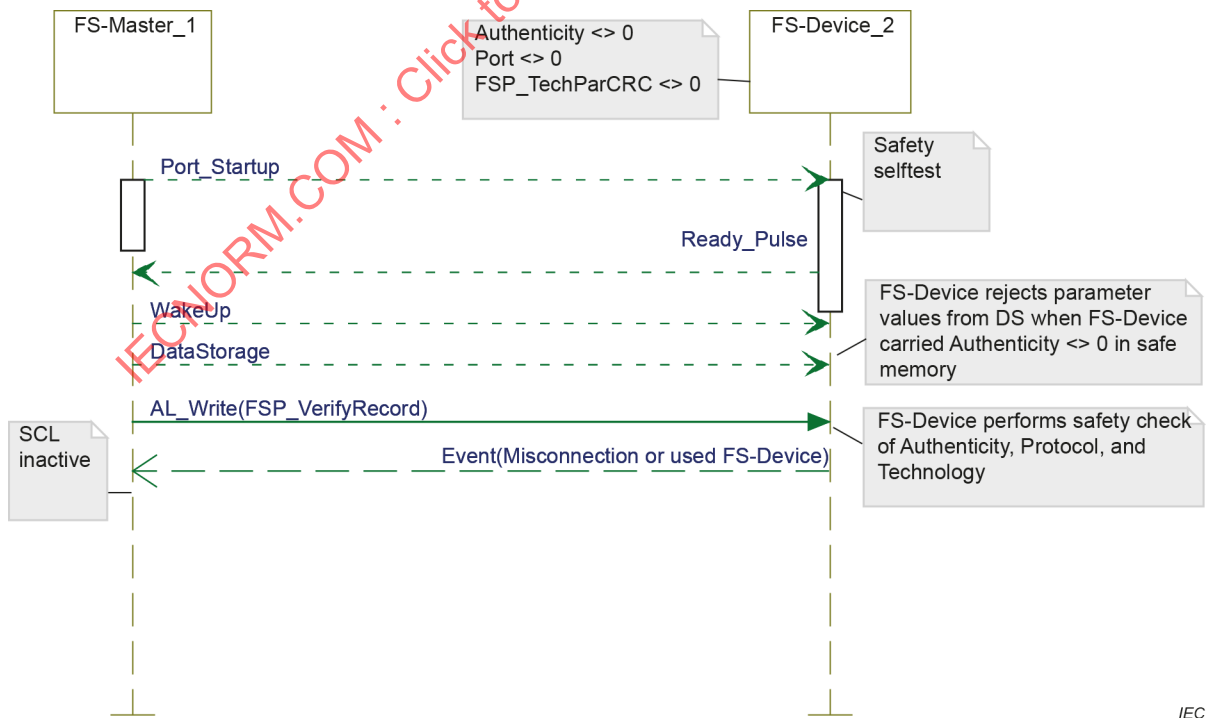


Figure G.3 – FS-Device misconnection

Annex H (normative)

System requirements

H.1 Indicators

H.1.1 General

Indicators for FS-Devices are not mandatory since for example proximity sensors may be too small for LEDs (light emitting diode).

FS-Masters and FS-Devices may be used in a mix of different technologies such as

- Fieldbus safety modules for inputs (e.g. F-DI module) or outputs (e.g. F-DO module),
- Safety devices such as light curtains connected to fieldbuses via FSCPs,
- SDCI Masters and Devices.

Thus, it is the designer's responsibility to layout the indication of the signal status, modes, or operations for FS-Masters and FS-Devices.

H.1.2 OSSDe

In case an FS-Master port is running in OSSDe mode, it behaves similar to an F-DI module port. One possibility of indication is using the same indication as with the SIO mode.

H.1.3 Safety communication

In case an FS-Master port is running in SCL mode, the normal non-safety operation indication can also be used.

H.1.4 Acknowledgment request

A machine is not allowed to restart automatically after a stop. Usually, after repair or clearance, the signal/service "ChFAckReq" is switched ON as specified in 11.11.4 and 11.11.5. It is highly recommended to indicate this signal on an FS-Master port and optionally on FS-Devices where it is likely to cause a trip due to high frequency or duration of exposure to a safety function.

H.2 Installation guidelines, electrical safety, and security

SDCI installation guidelines shall be considered (see [21]).

Only FS-Masters and FS-Devices providing a short form functional safety assessment report according to IEC 61508 or ISO 13849-1 together with a certificate of the assessment body are permitted. The short form report shall indicate all considered clauses and paragraphs of the used relevant standards and the corresponding assessment results.

Wireless connection between FS-Master and FS-Device is only permitted if interdependency with other wireless connections can be precluded, for example via inductive couplers.

No components in the link between FS-Master and FS-Device are permitted that are storing, inserting, or delaying messages.

Manufacturer/vendor of FS-Masters and/or FS-Devices shall define installation constraints for the operation of OSSD devices or FS-Devices in OSSDe mode within their safety manuals.

Requirements of IEC 61010-2-201 and IEC 60204-1 with respect to electrical safety (SELV/PELV) shall be observed.

The zones and conduit concept of IEC 62443 series applies for security and/or the rules of the applicable FSCP system.

H.3 Safety function response time

Safety manuals of FS-Master shall provide information on how to determine the safety function response time for OSSDe and for communication modes (see Clause H.6).

H.4 Duration of demands

Short demands of FS-Devices may not trip a safety function due to its chain of independent communication cycles across the network. Therefore, a demand shall last for at least two SCL (SPDU) cycles.

H.5 Maintenance and repair

FS-Devices can be replaced at runtime. Restart of the corresponding safety function is only permitted if there is no hazardous process state, after validation of the safety function(s), and after an operator acknowledgment.

H.6 Safety manual

FS-Masters and FS-Devices shall provide safety manuals according to the relevant national and international standards, for example IEC 61784-3:2021.

Manufacturer/vendor of FS-Masters and/or FS-Devices shall specify appropriate mitigation means in the safety manual for the deployment of SDCI-FS components in harsh industrial environment such as in EMC zones B and C according to IEC 61131-2.

Manufacturer/vendor of FS-Masters and/or FS-Devices shall define all constraints for the operation of OSSD devices or FS-Devices in OSSDe mode within their safety manuals.

Manufacturer/vendor of FS-Masters and/or FS-Devices shall define all constraints for the operation of FS-Devices in communication mode within their safety manuals such as limitations with respect to storing elements, inductive or optical couplers, and alike.

Manufacturer/vendor of FS-Masters and/or FS-Devices shall define the maintenance rules with respect to the PFH-Monitor (see Table 41).

Manufacturer/vendor of FS-Devices shall provide the "worst case delay time" (WCDT) value. WCDT is defined as the time from triggering an FS-Device (sensor) until its output shows a corresponding signal change or Process Data change under worst case conditions. For an FS-Device (actuator) it is the time from signal change or Process Data change to the actuator's safe state.

Manufacturer/vendor of FS-Devices shall provide the "one fault delay time" (OFDT) value. The definition of OFDT is similar to WCDT, however in case of a fault within the FS-Device at the time of triggering. Therefore, the value of OFDT is greater than the value of WCDT.

Manufacturer/vendor of FS-Masters shall provide information on how to determine the safety function response time as specified in IEC 61784-3:2021 using WCDTs and considering OFDTs.

Annex I
(informative)

**Information for test and assessment
of SDCI-FS components**

Information about test laboratories, which test and validate the conformance of SDCI-FS products such as FS-Masters and FS-Devices with IEC 61139-2 can be obtained from the National Committees of the IEC or from the following organization:

IO-Link Community
Haid-und-Neu-Str. 7
76131 Karlsruhe
GERMANY

Phone: +49 721 9658 590
Fax: +49 721 9658 589
E-Mail: info@io-link.com
URL: www.io-link.com

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

Bibliography

- [1] IEC 61784-3 (all parts), *Industrial communication networks – Profiles*
- [2] ISO/IEC 19505-2:2012, *Information technology – Object Management Group Unified Modeling Language (OMG UML) – Part 2: Superstructure*
- [3] Bruce P. Douglass, *Real Time UML – Advances in the UML for Real-Time Systems*, 3rd Edition, Addison-Wesley, ISBN 0-321-16076-2
- [4] Chris Rupp, Stefan Queins, die SOPHISTen, *UML 2 glasklar – Praxiswissen für die UML-Modellierung*. Hanser-Verlag, 2012, ISBN 978-3-446-43057-0
- [5] IEC TR 62390, *Common Automation Device – Profile guideline*
- [6] IEC 60947-5-5, *Low-voltage switchgear and controlgear – Part 5-5: Control circuit devices and switching elements – Electrical emergency stop device with mechanical latching function*
- [7] Position Paper CB24I, *Classification of Binary 24 V Interfaces – Functional Safety aspects covered by dynamic testing*, Edition 2.0.1:
<https://www.zvei.org/verband/fachverbaende/fachverband-automation/schaltgeraete-schaltanlagen-industriesteuerungen/>
- [8] Klaus Grimmer, *AIDA_IP-67-Safety_Positionspapier*, June 27th, 2013
- [9] IEC 60947-5-2, *Low-voltage switchgear and controlgear – Part 5-2: Control circuit devices and switching elements – Proximity switches*
- [10] IO-Link Community, *IO-Link Safety (Single Platform) – Requirements, Use Cases, and Concept Baseline*, V1.0, November 2014, Order No. 10.062
- [11] IEC 62769 (all parts), *Field Device Integration (FDI)*
- [12] FDT Joint Interest Group, *FDT 2.0 – Specification*, V1.0, Order No. 0001-0008-000
- [13] IEC 60947-5-6, *Low-voltage switchgear and controlgear – Part 5-6: Control circuit devices and switching elements – DC interface for proximity sensors and switching amplifiers (NAMUR)*
- [14] Philip Koopman, *Cyclic Redundancy Code (CRC) Polynomial Selection for Embedded Networks*, The International Conference on Dependable Systems and Networks, DSN-2004
- [15] Philip Koopman, *Best CRC Polynomials*, <https://users.ece.cmu.edu/~koopman/crc/> (Date: 29-Jan-2017)
- [16] CRC signature calculator for a seed value of "0":
<http://www.ghsi.de/pages/subpages/Online%20CRC%20Calculation/index.php?Polynom=10100111010101011&Message=1%0D%0A> (Date: 05-Apr-2018)
- [17] IO-Link Community, *IO Device Description (IODD)*, V1.1.3, January 2021, Order No. 10.012
- [18] IO-Link Community, *IO-Link Common Profile*, V1.1, December 2021, Order No. 10.072

- [19] IEC 62453 (all parts), *Field device tool (FDT) interface specification*
- [20] FDT Joint Interest Group, *FDT for IO-Link – Annex to FDT Specification – Based on FDT Specification Version 1.2.1*, V1.0, Order No. 0002-0013-000
- [21] IO-Link Community, *IO-Link Design Guideline*, V1.0, November 2016, Order No. 10.912
- [22] IEC 60947-5-3, *Low-voltage switchgear and controlgear – Part 5-3: Control circuit devices and switching elements – Requirements for proximity devices with defined behaviour under fault conditions*
- [23] IEC 61076-2-113, *Connectors for electronic equipment – Product requirements – Part 2-113: Circular connectors – Detail specification for connectors with M12 screw locking with power and signal contacts for data transmission with frequency up to 100 MHz*
- [24] IEC 61508-4:2010, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 4: Definitions and abbreviations*
- [25] ISO 14119:2013, *Safety of machinery – Interlocking devices associated with guards – Principles for design and selection*

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

SOMMAIRE

AVANT-PROPOS	208
INTRODUCTION.....	210
1 Domaine d'application	213
2 Références normatives	213
3 Termes, définitions, symboles, termes abrégés et conventions	214
3.1 Termes et définitions	214
3.2 Termes et définitions communs.....	214
3.3 Termes et définitions relatifs à la SDCI-FS.....	218
3.4 Symboles et abréviations	219
3.5 Conventions.....	221
3.5.1 Descriptions comportementales	221
3.5.2 Ordre des octets de mémoire et de transmission	222
4 Présentation de la SDCI-FS.....	222
4.1 Objet de la technologie et niveaux de fonctionnalité	222
4.1.1 Technologie SDCI-FS de base.....	222
4.1.2 De "analogique" et "commutation" vers la communication	223
4.1.3 Changement de paradigme réduit de FS-DI à Maître FS	224
4.1.4 Après le paradigme SDCI (SIO/OSSDe)	225
4.1.5 Port de classe B	227
4.1.6 "Maître USB" avec paramétrage de sécurité	228
4.1.7 Matrice d'interopérabilité des appareils de sécurité.....	228
4.2 Positionnement dans la hiérarchie d'automatisation	229
4.3 Câblage, connecteurs et alimentation	230
4.4 Relation avec la SDCI	230
4.5 Fonctionnalités et interfaces de communication	231
4.6 Paramétrage	231
4.7 Rôle du Maître FS et de la passerelle FS	232
4.8 Mapping vers les systèmes de niveau supérieur	232
4.9 Structure du document.....	232
5 Extensions de la couche physique (PL)	233
5.1 Vue d'ensemble	233
5.2 Extensions des services PL	234
5.2.1 PL_SetMode.....	234
5.2.2 PL_Ready.....	234
5.3 Emetteur/Récepteur	234
5.3.1 Hypothèses pour l'extension vers OSSDe	234
5.3.2 Spécificités OSSDe	235
5.3.3 Démarrage d'un Appareil FS (impulsion Ready).....	238
5.3.4 Caractéristiques électriques d'un récepteur dans l'Appareil FS et le Maître FS	238
5.4 Caractéristiques électriques et dynamiques d'un Appareil FS.....	239
5.5 Caractéristiques électriques et dynamiques d'un port de Maître FS (OSSDe).....	242
5.6 Interface FS-DI du port du Maître FS	244
5.7 Coordination de la réactivation.....	244
5.8 Démarrage rapide	245
5.9 Alimentation.....	245
5.10 Support.....	246

5.10.1	Contraintes	246
5.10.2	Connecteurs	246
5.10.3	Caractéristiques des câbles	246
6	Extensions du mode SIO	246
7	Extensions de la couche de liaison de données (DL)	246
7.1	Vue d'ensemble	246
7.2	Diagramme d'états du gestionnaire de mode DL du Maître FS	246
7.3	Diagramme d'états du gestionnaire de mode DL de l'Appareil FS	248
8	Extensions du gestionnaire de configuration (CM) du Maître	250
9	Extensions de l'Appareil FS	251
9.1	Architecture de principe et modèles	251
9.1.1	Architecture de l'Appareil FS	251
9.1.2	Modèle de l'Appareil FS	252
9.2	Gestionnaire de paramètres (PM)	253
9.3	Echange des données de processus (PDE)	253
9.4	Stockage de données (DS)	254
9.4.1	Considérations générales et extensions, y compris de sécurité	254
9.4.2	Niveaux de sauvegarde	254
10	Extensions du Maître FS	255
10.1	Architecture de principe	255
10.2	Extensions des services SMI	256
10.2.1	Vue d'ensemble	256
10.2.2	SMI_FSMasterAccess	257
10.2.3	SMI_SPDUIn	259
10.2.4	SMI_SPDUOut	259
10.2.5	SMI_FSPDInOut	259
10.3	Extensions d'ArgBlock	261
10.3.1	Vue d'ensemble	261
10.3.2	FSMasterAccess	262
10.3.3	FSCPAuthenticity	262
10.3.4	FSPortConfigList	262
10.3.5	FSPortStatusList	265
10.3.6	SPDUIn	267
10.3.7	SPDUOut	267
10.3.8	FSPDInOut	268
10.4	Gestionnaire de couche de sécurité (SLM)	269
10.4.1	Objet	269
10.4.2	FS_PortMode	269
10.4.3	Paramètre FSP	269
10.5	Echange des données de processus (PDE)	273
10.6	Stockage de données (DS)	274
11	Couche de communication de sécurité (SCL)	274
11.1	Exigences fonctionnelles	274
11.2	Erreurs de communication et mesures de sécurité	274
11.3	Services SCL	275
11.3.1	Positionnement des couches de communication de sécurité (SCL)	275
11.3.2	Services SCL du Maître FS	276
11.3.3	Services SCL de l'Appareil FS	278

11.4	Protocole SCL.....	280
11.4.1	Phases de protocole à prendre en compte	280
11.4.2	Anomalies de l'Appareil FS	281
11.4.3	PDU de sécurité (SPDU).....	282
11.4.4	Données d'entrée FS et de sortie FS	282
11.4.5	Numéro de port.....	282
11.4.6	Etat et contrôle	282
11.4.7	Signature CRC	283
11.4.8	Considérations de sécurité relatives à l'opportunité, l'authenticité et l'intégrité des données ("TADI") (informative)	284
11.4.9	Types de données pour SDCI-FS.....	285
11.5	Comportement SCL.....	287
11.5.1	Généralités	287
11.5.2	Diagramme d'états SCL du Maître FS	287
11.5.3	Diagramme d'états SCL de l'Appareil FS	290
11.5.4	Diagrammes séquentiels pour plusieurs cas d'utilisation.....	293
11.5.5	Surveillance des temps de sécurité.....	299
11.5.6	Réaction en cas de dysfonctionnement.....	300
11.5.7	Démarrage (communication).....	302
11.6	Gestion de la SCL.....	303
11.6.1	Vue d'ensemble des paramètres (FSP et FST)	303
11.6.2	Approches de paramétrage.....	304
11.7	Mesures d'intégrité	305
11.7.1	Intégrité de l'IODD	305
11.7.2	Intégrité des outils	305
11.7.3	Intégrité de la transmission.....	306
11.7.4	Enregistrement de vérification	306
11.7.5	Authentification.....	306
11.7.6	Intégrité du stockage	306
11.7.7	Intégrité de la structure de données d'E/S FS	307
11.7.8	Paramètres de technologie (FST) fondés sur l'IODD	308
11.7.9	Paramètres de technologie (FST) fondés sur un outil dédié existant (IOPD).....	309
11.8	Création des paramètres FSP et FST.....	309
11.9	Intégration des outils dédiés (IOPD).....	310
11.9.1	Interface IOPD.....	310
11.9.2	Interfaces normalisées.....	311
11.9.3	Canal de retour.....	311
11.10	Validation.....	312
11.11	Passivation	312
11.11.1	Motifs et moyens	312
11.11.2	Passivation sélective du port (Maître FS).....	313
11.11.3	Passivation sélective du signal (borne FS).....	313
11.11.4	Réglages des qualificatifs en cas de communication.....	313
11.11.5	Gestion des qualificatifs en cas d'OSSDe	314
11.12	Diagnostic SCL	315
12	Traitement de sécurité fonctionnelle (FS-P).....	315
12.1	Recommandations pour des mappings d'E/S efficaces.....	315
12.2	Contrôleur FS intégré.....	316

Annexe A (normative) Extensions des paramètres	317
A.1 Index et paramètres pour SDCI-FS	317
A.2 Détail des paramètres	318
A.2.1 FSP_Authenticity	318
A.2.2 FSP_Port	319
A.2.3 FSP_AuthentCRC	319
A.2.4 FSP_ProtVersion	319
A.2.5 FSP_ProtMode	319
A.2.6 FSP_Watchdog	320
A.2.7 FSP_IO_StructCRC	320
A.2.8 FSP_TechParCRC	321
A.2.9 FSP_ProtParCRC	321
A.2.10 FSP_VerifyRecord	322
A.2.11 FSP_TimeToReady	322
A.2.12 FSP_MinShutDownTime	322
A.2.13 FSP_WCDT	322
A.2.14 FSP_OFDT	322
A.2.15 FSP_ParamDescCRC	322
Annexe B (normative) Extensions des EventCodes	324
B.1 EventCodes d'Appareil FS supplémentaires	324
B.2 EventCodes de port supplémentaires	324
Annexe C (normative) Extensions des types de données	326
C.1 Types de données pour SDCI-FS	326
C.2 BooleanT (bit)	326
C.3 IntegerT (16)	327
C.4 IntegerT (32)	327
C.5 Code de sécurité	328
Annexe D (normative) Polynômes générateurs de CRC	330
D.1 Vue d'ensemble des polynômes générateurs de CRC	330
D.2 Probabilités d'erreurs résiduelles	331
D.3 Considérations relatives à la mise en œuvre	332
D.3.1 Vue d'ensemble	332
D.3.2 Algorithme de décalage de bits (16 bits)	332
D.3.3 Table de conversion (16 bits)	333
D.3.4 Algorithme de décalage de bits (32 bits)	334
D.3.5 Table de conversion (32 bits)	335
D.3.6 Valeurs de départ	336
D.3.7 Ordre des octets pour le calcul de la CRC	337
Annexe E (normative) Extensions de l'IODD	338
E.1 Généralités	338
E.2 Schéma	338
E.3 Contraintes de l'IODD	338
E.3.1 Règles générales	338
E.3.2 Description de la structure de l'IODD	338
E.3.3 Comportement des SystemCommands "Reset" dans la SDCI-FS	345
E.3.4 Caractéristiques de profil	346
E.3.5 ProcessDataInput et ProcessDataOutput	346
E.4 Conventions IODD	346

E.4.1	Nommage	346
E.4.2	Données de processus (PD)	346
E.4.3	Conventions IODD pour l'interface utilisateur	347
E.4.4	Fonctionnalités des outils du Maître	347
E.5	Sécurisation	347
E.5.1	Généralités	347
E.5.2	DefaultValues pour FSP	348
E.5.3	FSP_Authenticity	348
E.5.4	FSP_Protocol	348
E.5.5	FSP_IO_Description	349
E.5.6	Exemple de sérialisation pour FSP_ParamDescCRC	349
E.5.7	Paramètres FST et FSP et stockage de données	350
E.5.8	Exemple d'IODD d'un Appareil FS	351
Annexe F (normative)	Interface d'outil d'Appareil (DTI) pour SDCI	361
F.1	Objet de la DTI	361
F.2	Modèle de base	361
F.3	Interface d'appel	362
F.3.1	Vue d'ensemble	362
F.3.2	Détection de l'outil d'Appareil	363
F.3.3	Description d'interface de programme (PID)	366
F.3.4	Fichier de paramètres temporaire (TPF)	370
F.3.5	Fichier de backchannel temporaire (TBF)	375
F.3.6	Fichier d'acquiescement temporaire (TAF)	377
F.3.7	Comportement d'appel	377
F.4	Objets de données d'Appareil (DDO)	378
F.4.1	Généralités	378
F.4.2	Structure d'un paquet DDO	379
F.5	Interface de communication	379
F.5.1	Généralités	379
F.5.2	Principe des communications DTI	380
F.5.3	Passerelles	382
F.5.4	Configuration du serveur de communication	382
F.5.5	Définition de l'interface de communication	382
F.5.6	Séquence d'établissement d'une relation de communication	382
F.5.7	Utilisation du serveur de communication en mode autonome	384
F.5.8	Spécificités de la SDCI	384
F.5.9	Modification des paramètres de communication	385
F.6	Réaction en cas de comportement incorrect de l'outil	385
F.7	Compatibilité	386
F.7.1	Validation du schéma	386
F.7.2	Politique relative aux versions	386
F.8	Evolutivité	387
F.8.1	Evolutivité d'un outil d'Appareil	387
F.8.2	Evolutivité d'un outil de Maître	387
F.8.3	Interactions selon les combinaisons de classes de conformité	388
F.9	Définitions des schémas	388
F.9.1	Généralités	388
F.9.2	Schéma du PID	388
F.9.3	Schéma du TPF	390

F.9.4	Schéma du TBF	392
F.9.5	Schéma du TAF	393
F.9.6	Schéma des primitives DTI	394
Annexe G (normative) Principaux scénarios de SDCI-FS		396
G.1	Vue d'ensemble	396
G.2	Diagramme séquentiel de mise en service	397
G.3	Diagramme séquentiel de remplacement	398
G.4	Diagramme séquentiel de mauvaise connexion	399
Annexe H (normative) Exigences système		400
H.1	Indicateurs	400
H.1.1	Généralités	400
H.1.2	OSSDe	400
H.1.3	Communication de sécurité	400
H.1.4	Demande d'acquiescement	400
H.2	Lignes directrices relatives à l'installation, sûreté électrique et sécurité	400
H.3	Temps de réponse de la fonction de sécurité	401
H.4	Durée des demandes (ou sollicitations)	401
H.5	Maintenance et réparation	401
H.6	Manuel de sécurité	401
Annexe I (informative) Informations pour l'essai et l'évaluation des composants SDCI-FS		403
Bibliographie		404
Figure 1 – Positionnement de la SDCI-FS pour l'automatisation de sécurité fonctionnelle		211
Figure 2 – Relations du présent document avec les normes		212
Figure 3 – Ordre des octets de mémoire et de transmission		222
Figure 4 – Modèle de couche de communication SDCI-FS		222
Figure 5 – Extensions d'interface de port pour SDCI-FS		223
Figure 6 – Migration vers une SDCI-FS		224
Figure 7 – Changement de paradigme réduit de FS-DI à Maître FS		225
Figure 8 – Types de Maîtres FS et niveaux de fonctionnalité		225
Figure 9 – Disposition d'origine des broches de la SDCI (ports de classe A)		226
Figure 10 – Mise en service OSSDe optimisée avec un Maître FS		227
Figure 11 – Niveau "d" d'un Maître FS (classe B)		228
Figure 12 – Configuration et paramétrage hors site		228
Figure 13 – SDCI-FS dans la hiérarchie d'automatisation		230
Figure 14 – Couche physique SDCI d'un Maître FS (classe A)		233
Figure 15 – Couche physique d'un Appareil FS (classe A)		233
Figure 16 – Compatibilité transversale entre OSSD et OSSDe		234
Figure 17 – Principe de fonctionnement OSSDe		235
Figure 18 – Impulsions d'essai pour détecter les anomalies d'interconnexion		237
Figure 19 – Temporisations OSSD		237
Figure 20 – Démarrage type d'un capteur OSSD		238
Figure 21 – Démarrage d'un Appareil FS		238

Figure 22 – Seuils de commutation pour les récepteurs de l'Appareil FS et du Maître FS.....	239
Figure 23 – Schéma de référence (un canal OSSDe).....	240
Figure 24 – Définitions des niveaux de tension.....	240
Figure 25 – Capacité de charge à la mise sous tension.....	244
Figure 26 – Résolution du conflit avec le filtre d'entrée OSSDe.....	244
Figure 27 – Démarrage d'un Appareil FS.....	245
Figure 28 – Temporisations de démarrage rapide exigées.....	245
Figure 29 – Diagramme d'états du gestionnaire de mode DL du Maître FS.....	247
Figure 30 – Diagramme d'états du gestionnaire de mode DL de l'Appareil FS.....	249
Figure 31 – Extension du gestionnaire de configuration (VerifyRecord).....	250
Figure 32 – Architecture de principe de l'Appareil FS.....	252
Figure 33 – Modèle de l'Appareil FS.....	253
Figure 34 – Architecture de principe du Maître FS.....	255
Figure 35 – Extensions des services SMI.....	257
Figure 36 – Cas d'utilisation des paramètres FSP.....	270
Figure 37 – Séparateur PDE.....	273
Figure 38 – Compositeur PDE.....	274
Figure 39 – Positionnement de la couche de communication de sécurité SDCI-FS (SCL).....	276
Figure 40 – Services de la couche de communication de sécurité du Maître FS.....	277
Figure 41 – Services de la couche de communication de sécurité de l'Appareil FS.....	279
Figure 42 – Phases de protocole à prendre en compte.....	281
Figure 43 – PDU de sécurité du Maître FS et de l'Appareil FS.....	282
Figure 44 – Règle du taux de 1 % de l'IEC 61784-3:2021.....	284
Figure 45 – Diagramme d'états SCL du Maître FS.....	287
Figure 46 – Diagramme d'états SCL de l'Appareil FS.....	290
Figure 47 – Maître FS et Appareil FS tous deux sous tension (ON).....	293
Figure 48 – Maître FS hors tension puis sous tension (OFF → ON).....	294
Figure 49 – Appareil FS avec démarrage SCL différé.....	295
Figure 50 – Appareil FS hors tension puis sous tension (OFF → ON).....	296
Figure 51 – Détection d'une erreur de signature CRC par le Maître FS.....	297
Figure 52 – Détection d'une erreur de signature CRC par l'Appareil FS.....	298
Figure 53 – Surveillance de la durée de cycle SCL.....	299
Figure 54 – Types de paramètres et affectations.....	304
Figure 55 – Système centré sur l'hôte FSCP.....	305
Figure 56 – Structure de FSP_VerifyRecord.....	306
Figure 57 – Démarrage de la SDCI-FS.....	307
Figure 58 – Sécurisation des paramètres FST avec un outil dédié.....	308
Figure 59 – Modification des paramètres FST avec un outil d'Appareil.....	309
Figure 60 – Création des paramètres FSP et FST.....	310
Figure 61 – Exemple de hiérarchie de communication.....	311
Figure 62 – Motifs de la passivation sélective du port.....	312
Figure 63 – Gestionnaire de qualificatifs (communication).....	313

Figure 64 – Gestionnaire de qualificatifs (OSSDe)	314
Figure 65 – Comportement des qualificatifs par port du Maître FS	314
Figure 66 – Problèmes liés à l'efficacité du mapping	316
Figure A.1 – Instance d'une description des données d'E/S FS	321
Figure A.2 – Exemple de structure de données d'E/S FS avec des données non relatives à la sécurité	321
Figure A.3 – Sécurisation des paramètres de sécurité	323
Figure C.1 – Exemple de structure de données BooleanT	327
Figure C.2 – Code de sécurité d'un message de sortie	328
Figure C.3 – Code de sécurité d'un message d'entrée	329
Figure D.1 – Polynôme générateur de CRC-16	331
Figure D.2 – Polynôme générateur de CRC-32	332
Figure D.3 – Algorithme de décalage de bits en langage "C" (16 bits)	333
Figure D.4 – Calcul de la signature CRC-16 à l'aide d'une table de conversion	333
Figure D.5 – Algorithme de décalage de bits en langage "C" (32 bits)	335
Figure D.6 – Calcul de la signature CRC-32 à l'aide d'une table de conversion	335
Figure E.1 – Algorithme de génération des signatures CRC des paramètres FSP	348
Figure F.1 – Principe de l'interface d'appel DTI	362
Figure F.2 – Structure du registre	363
Figure F.3 – Exemple de registre DTI	364
Figure F.4 – Détection d'un outil d'Appareil dans le registre	366
Figure F.5 – Menu pour appeler l'outil d'Appareil	367
Figure F.6 – Objectif des objets de données d'Appareil (DDO)	379
Figure F.7 – Voies de communication entre l'outil d'Appareil et l'Appareil	380
Figure F.8 – Routage sur les réseaux et la SDCI	381
Figure F.9 – Serveur de communication	381
Figure F.10 – Diagramme séquentiel pour l'établissement d'une communication	383
Figure F.11 – Création d'une instance de serveur de communication	383
Figure F.12 – Exemple de document XML ConnectRequest pour SDCI	385
Figure F.13 – Schéma XML du fichier PID	388
Figure F.14 – Schéma XML du TPF	390
Figure F.15 – Schéma XML d'un TBF	392
Figure G.1 – Mise en service avec essai et fonctionnement armé	398
Figure G.2 – Remplacement d'Appareil FS	399
Figure G.3 – Mauvaise connexion d'Appareil FS	399
Tableau 1 – Modes opérationnels des niveaux de fonctionnalité "a" à "c" (ports de classe A)	227
Tableau 2 – Matrice d'interopérabilité des appareils de sécurité	229
Tableau 3 – PL_Ready	234
Tableau 4 – Etats et conditions OSSD	236
Tableau 5 – Anomalies d'interconnexion	236
Tableau 6 – Caractéristiques électriques d'un récepteur	239
Tableau 7 – Caractéristiques électriques et dynamiques d'un Appareil FS (OSSDe)	241

Tableau 8 – Caractéristiques électriques et dynamiques de l'interface du port	243
Tableau 9 – Caractéristiques des câbles.....	246
Tableau 10 – Tableaux de transition d'états du gestionnaire de mode DL du Maître FS	248
Tableau 11 – Tableaux de transition d'états du gestionnaire de mode DL de l'Appareil FS	249
Tableau 12 – Tableaux de transition d'états du gestionnaire de configuration	251
Tableau 13 – Extension du diagramme d'états de stockage de données (DS)	254
Tableau 14 – Niveaux de sauvegarde du stockage de données	254
Tableau 15 – Services SMI utilisés pour le Maître FS	256
Tableau 16 – SMI_FSMasterAccess.....	258
Tableau 17 – SMI_FSPDInOut	260
Tableau 18 – Types d'ArgBlocks et ArgBlockIDs	261
Tableau 19 – FSMasterAccess.....	262
Tableau 20 – FSCPAAuthenticity	262
Tableau 21 – FSPortConfigList	263
Tableau 22 – FSPortStatusList	266
Tableau 23 – SPDUIIn	267
Tableau 24 – SPDUIOut.....	268
Tableau 25 – FSPDInOut	268
Tableau 26 – Tableau de référence des cas d'utilisation	270
Tableau 27 – Erreurs de communication et mesures de sécurité	275
Tableau 28 – Services SCL du Maître FS.....	278
Tableau 29 – Services SCL de l'Appareil FS.....	280
Tableau 30 – Phases de protocole à prendre en compte	281
Tableau 31 – Contrôle et comptage (Control&MCnt)	283
Tableau 32 – Etat et comptage en miroir (Status&DCnt)	283
Tableau 33 – Valeurs MCount et DCount_i	283
Tableau 34 – Types de données d'E/S de processus FS.....	286
Tableau 35 – Règles de disposition des valeurs et des qualificatifs.....	286
Tableau 36 – Ordre des valeurs et des qualificatifs	286
Tableau 37 – Définition des termes utilisés dans le diagramme d'états SCL du Maître FS.....	288
Tableau 38 – Etats et transitions SCL du Maître FS	288
Tableau 39 – Définition des termes utilisés dans le diagramme d'états SCL de l'Appareil FS	290
Tableau 40 – Etats et transitions SCL de l'Appareil FS	291
Tableau 41 – Contraintes de temporisation	299
Tableau 42 – Bits qualificatifs "GOOD/BAD"	313
Tableau 43 – Tableaux de transition d'états pour le comportement des qualificatifs	315
Tableau A.1 – Index pour SDCI-FS	317
Tableau A.2 – Codage de la version de protocole	319
Tableau A.3 – Codage du mode de protocole.....	319
Tableau A.4 – Description de la structure de données d'E/S FS générique	320
Tableau B.1 – EventCodes spécifiques à la SCL de l'Appareil FS	324

Tableau B.2 – EventCodes spécifiques à la SCL du Maître FS.....	325
Tableau C.1 – Types de données pour SDCI-FS.....	326
Tableau C.2 – BooleanT pour SDCI-FS.....	326
Tableau C.3 – Exemple de BooleanT dans un RecordT	326
Tableau C.4 – IntegerT(16).....	327
Tableau C.5 – Codage d'IntegerT(16)	327
Tableau C.6 – IntegerT(32).....	328
Tableau C.7 – Codage d'IntegerT(32)	328
Tableau D.1 – Polynômes générateurs de CRC pour SDCI-FS.....	330
Tableau D.2 – Définition des variables utilisées à la Figure D.3	333
Tableau D.3 – Définition des variables utilisées à la Figure D.4	333
Tableau D.4 – Table de conversion pour le calcul de la signature CRC-16.....	334
Tableau D.5 – Définition des variables utilisées à la Figure D.5	335
Tableau D.6 – Définition des variables utilisées à la Figure D.4	335
Tableau D.7 – Table de conversion pour le calcul de la signature CRC-32.....	336
Tableau E.1 – Comportement spécifique des SystemCommands "Reset" de l'Appareil FS	346
Tableau E.2 – Actions utilisateur pour remplacer les DefaultValue	348
Tableau E.3 – RecordItems de FSP_Protocol dont les valeurs admises doivent être sérialisées	349
Tableau E.4 – Exemple de sérialisation pour FSP_ParamDescCRC.....	349
Tableau F.1 – Description des éléments du fichier PID	367
Tableau F.2 – Eléments du TPF.....	370
Tableau F.3 – Eléments du TBF.....	376
Tableau F.4 – Eléments du TAF.....	377
Tableau F.5 – Cas et comportements d'appel	378
Tableau F.6 – Mapping du schéma de communication	384
Tableau F.7 – Réaction en cas de comportement incorrect de l'outil	386
Tableau F.8 – Classes de conformité DTI	387
Tableau F.9 – Niveaux de fonctionnalité DTI des outils d'Appareil	387
Tableau F.10 – Interactions selon les combinaisons de classes de conformité	388
Tableau G.1 – Principaux scénarios de SDCI-FS	396

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

RÉSEAUX INDUSTRIELS – INTERFACE DE COMMUNICATION NUMÉRIQUE POINT À POINT –

Partie 2: Extensions de sécurité fonctionnelle

AVANT-PROPOS

- 1) La Commission Electrotechnique 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. A 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.

L'IEC 61139-2 a été établie par le sous-comité 65C: Réseaux industriels, du comité d'études 65 de l'IEC: Mesure, commande et automation dans les processus industriels. Il s'agit d'une Norme internationale.

Le texte de cette Norme internationale est issu des documents suivants:

Projet	Rapport de vote
65C/1168/FDIS	65C/1174/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à son approbation.

La langue employée pour l'élaboration de cette Norme internationale est l'anglais.

Ce 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 61139, publiées sous le titre général *Réseaux industriels – Interface de communication numérique point à point*, se trouve sur le site web de l'IEC.

Le comité a décidé que le contenu de ce document ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous webstore.iec.ch dans les données relatives au document recherché. A 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 cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

INTRODUCTION

La technologie de base IO-Link™¹ est l'objet de la Norme internationale IEC 61131-9, qui fait partie d'une série de normes sur les automates programmables et les périphériques associés, tels que les systèmes d'E/S déportés (RIO).

Elle spécifie une technologie d'interface de communication numérique point à point (appelée SDCI, *Single-drop Digital Communication Interface*), qui étend les interfaces d'entrée et de sortie de commutation traditionnelles définies dans l'IEC 61131-2 vers une liaison de communication point à point qui utilise une commutation codée. Cette technologie permet l'échange cyclique de données de processus d'entrée et de sortie numériques entre un Maître et ses Appareils associés (capteurs, actionneurs, bornes d'E/S, etc.). Le Maître peut faire partie d'un système de communication par bus de terrain ou de toute unité de traitement autonome. La technologie permet également le transfert acyclique de paramètres vers les Appareils et la propagation d'informations de diagnostic des Appareils vers le système d'automatisation de niveau supérieur (contrôleur, hôte) par l'intermédiaire du Maître.

La topologie physique point à point entre chaque Appareil et le Maître utilise 3 fils sur des distances inférieures ou égales à 20 m. L'interface physique SDCI est rétrocompatible avec la signalisation d'E/S 24 V habituelle spécifiée dans l'IEC 61131-2 et prend en charge trois taux de transmission, de 4,8 kbit/s, 38,4 kbit/s et 230,4 kbit/s.

Les principaux avantages de la technologie SDCI sont les suivants:

- double utilisation des communications à signaux de commutation (DI/DO) ou à commutation codée, respectivement;
- les capteurs et les actionneurs de commutation traditionnels fournissent désormais, en variante, une communication numérique point à point au sein d'un même Appareil;
- un câble fin, robuste et très souple, sans blindage, pour l'alimentation et la signalisation;
- communication numérique à moindre coût, jusqu'aux moindres capteurs et actionneurs.

La variante de sécurité fonctionnelle de la SDCI est appelée SDCI-FS. La Figure 1 représente un exemple de positionnement de la SDCI-FS pour l'automatisation de sécurité fonctionnelle.

¹ IO-Link™ est le nom commercial de la communauté IO-Link. Cette information est donnée à l'intention des utilisateurs de la présente spécification et ne signifie nullement que la communauté IO-Link approuve ou recommande le détenteur de la marque ou de l'un de ses produits. La conformité au présent document n'exige pas l'emploi des logos déposés pour IO-Link™. L'emploi des logos déposés pour IO-Link™ exige l'autorisation de la communauté IO-Link.

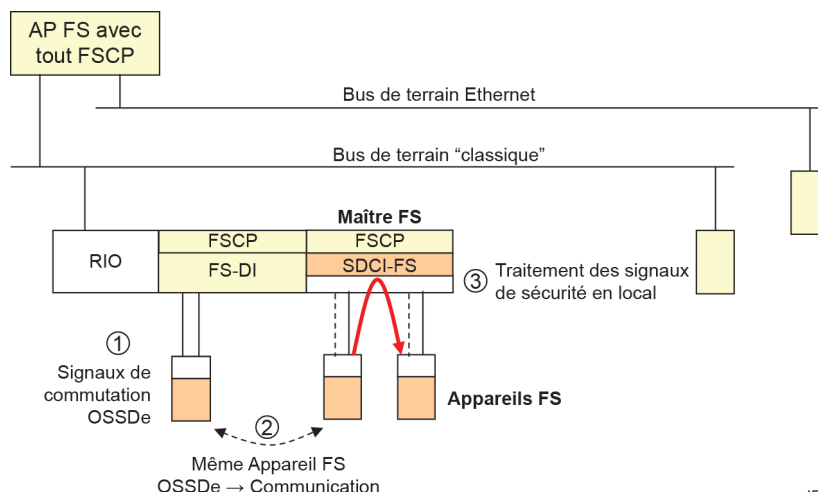


Figure 1 – Positionnement de la SDCI-FS pour l'automatisation de sécurité fonctionnelle

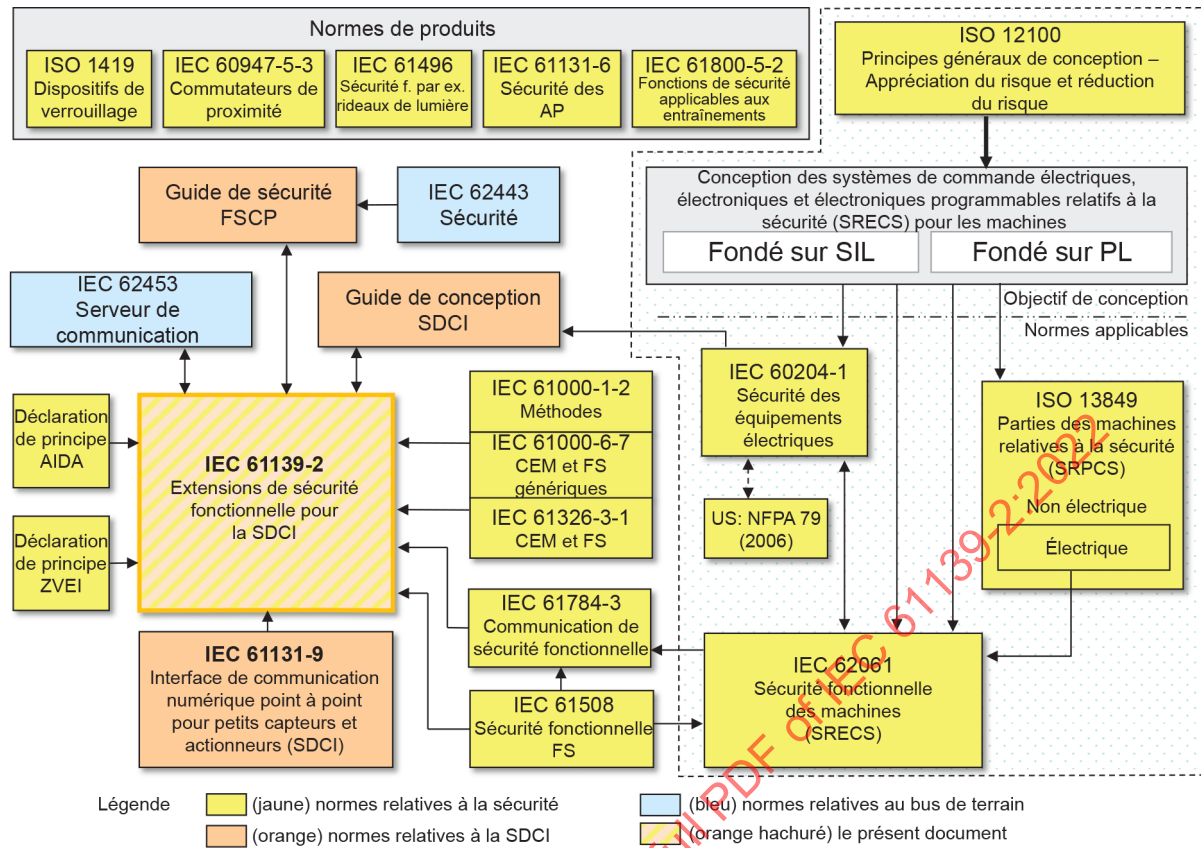
Dans cet exemple, un système d'E/S déporté est connecté à un automate programmable de sécurité fonctionnelle par l'un des FSCP de la série IEC 61784-3 [1]² pour communiquer avec un module FS-DI et une passerelle vers un Maître FS SDCI-FS. Les Appareils FS avec OSSDe peuvent être connectés à des FS-DI ou à des Maîtres FS. Tous les Appareils FS peuvent communiquer avec tout Maître FS en utilisant le protocole SDCI-FS, quel que soit le système FSCP de niveau supérieur. Il en va de même pour les actionneurs de sécurité (Appareils FS) tels que les commandes à sécurité intégrée. Cela implique une plus grande uniformité des composants ① pour les capteurs et les actionneurs, similaire aux interfaces DI et DO normalisées dans le cadre de l'IEC 61131-2.

Les capteurs de sécurité avec interfaces OSSDe (équipés d'un système de communication SDCI-FS) peuvent être paramétrés par l'intermédiaire d'outils auxiliaires, tels que des "Maîtres USB", puis connectés à un FS-DI et utilisés en mode OSSDe. Ils peuvent également être utilisés en mode OSSDe sur un Maître FS qui prend en charge OSSDe. Si ces capteurs de sécurité sont équipés d'un système de communication SDCI-FS supplémentaire, ils peuvent être utilisés dans les deux modes ②, OSSDe ou SDCI-FS. Cela correspond au paradigme SIO SDCI.

Le concept de SDCI-FS permet le traitement des signaux de sécurité en local si la passerelle/le Maître FS fournit un contrôleur de sécurité local ③.

Le présent document fournit les extensions nécessaires à l'IEC 61131-9 pour la communication de sécurité fonctionnelle, y compris la normalisation de l'OSSDe et le paramétrage dans le domaine de la sécurité des machines. La Figure 2 représente les relations avec les Normes internationales de bus de terrain et de sécurité, ainsi qu'avec les spécifications applicables (voir Article 2 et Bibliographie). Toute sécurité fonctionnelle commence par l'appréciation du risque et la réduction du risque (ISO 12100). Une possibilité de réduction du risque consiste à utiliser des systèmes de commande électriques ou électroniques. Pour la conception de ces derniers, des normes telles que l'IEC 61508, l'IEC 62061 et l'ISO 13849 peuvent être utilisées. Les conditions d'environnement, comme la CEM, sont couvertes, par exemple, par l'IEC 61000-6-7. D'autres aspects concernent les installations et les questions de sécurité. Plusieurs normes de produits, telles que l'IEC 60947-5-3 et l'ISO 14119, complètent les normes génériques ou sectorielles.

² Les chiffres entre crochets renvoient à la Bibliographie.



IEC

Figure 2 – Relations du présent document avec les normes

La SDCI-FS peut être utilisée pour les applications de sécurité fonctionnelle selon l'IEC 62061 et l'IEC 61508 jusqu'à SIL3 et/ou selon l'ISO 13849 jusqu'à PLe.

RÉSEAUX INDUSTRIELS – INTERFACE DE COMMUNICATION NUMÉRIQUE POINT À POINT –

Partie 2: Extensions de sécurité fonctionnelle

1 Domaine d'application

La présente partie de l'IEC 61139 spécifie les extensions de la SDCI de l'IEC 61131-9 pour la sécurité fonctionnelle. Elle comprend:

- une interface OSSDe normalisée pour les signaux de commutation redondants, fondée sur l'IEC 61131-2;
- des modifications/extensions mineures des diagrammes d'états de la SDCI afin de prendre en charge les opérations de sécurité;
- un protocole de communication de sécurité fonctionnelle affiné en plus de la communication SDCI normalisée, qui constitue un canal noir selon l'IEC 61784-3:2021;
- des fonctions de gestion de protocole pour la configuration, le paramétrage et la mise en service;
- des extensions de l'IODD pour la sécurité fonctionnelle;
- une interface d'outil d'Appareil pour prendre en charge les outils dédiés, conformément aux normes de sécurité fonctionnelle.

Le présent document ne couvre pas:

- les interfaces ou systèmes de communication qui incluent des liaisons multipoints;
- les interfaces ou systèmes de communication qui incluent des liaisons chiffrées ou multicanaux;
- les interfaces ou systèmes de communication sans fil;
- l'intégration de la SDCI-FS aux systèmes de niveau supérieur, tels que les bus de terrain/FSCP.

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 60204-1, *Sécurité des machines – Equipement électrique des machines – Partie 1: Exigences générales*

IEC 61010-2-201, *Exigences de sécurité pour appareils électriques de mesure, de régulation et de laboratoire – Partie 2-201: Exigences particulières relatives aux équipements de commande*

IEC 61131-2, *Mesure et commande des processus industriels – Automates programmables – Partie 2: Spécifications et essais des équipements*

IEC 61131-9:—³, *Automates programmables – Partie 9: Interface de communication numérique point à point pour petits capteurs et actionneurs (SDCI)*

IEC 61496-1, *Sécurité des machines – Equipements de protection électrosensibles – Partie 1: Exigences générales et essais*

IEC 61508-3, *Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité – Partie 3: Exigences concernant les logiciels*

IEC 61784-3:2021, *Réseaux de communication industriels – Profils – Partie 3: Bus de terrain de sécurité fonctionnelle – Règles générales et définitions de profils*

IEC 62061, *Sécurité des machines – Sécurité fonctionnelle des systèmes de commande relatifs à la sécurité*

IEC 62443 (toutes les parties), *Sécurité des systèmes d'automatisation et de commande industriels*

ISO 639-2, *Codes pour la représentation des noms de langue – Partie 2: Code alpha-3*

ISO 639-3, *Codes pour la représentation des noms de langue – Partie 3: Code alpha-3 pour un traitement exhaustif des langues*

ISO 13849-1, *Sécurité des machines – Parties des systèmes de commande relatives à la sécurité – Partie 1: Principes généraux de conception*

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 de l'IEC 61131-2 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>

3.2 Termes et définitions communs

3.2.1

adresse

partie d'une commande de séquence M utilisée pour référencer des données dans les catégories de données d'un canal de communication

3.2.2

couche d'application

AL

<SDCI>⁴ 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: L'abréviation "AL" est dérivée du terme anglais développé correspondant "application layer".

³ En cours d'élaboration. Stade au moment de la publication: IEC/CFDIS 61131-9:2022.

⁴ Les crochets obliques indiquent la validité de la définition pour la technologie SDCI.

3.2.3**paramètre de bloc**

accès de paramètre cohérent par plusieurs index ou sous-index

3.2.4**somme de contrôle**

<SDCI> partie complémentaire des mesures globales d'intégrité des données dans la couche de liaison de données, en plus du bit de parité UART

3.2.5**commutation codée**

communication SDCI, fondée sur les niveaux de signaux binaires normalisés de l'IEC 61131-2

3.2.6**COM1**

mode de communication SDCI avec un taux de transmission de 4,8 kbit/s

3.2.7**COM2**

mode de communication SDCI avec un taux de transmission de 38,4 kbit/s

3.2.8**COM3**

mode de communication SDCI avec un taux de transmission de 230,4 kbit/s

3.2.9**COMx**

un des trois modes de communication SDCI possibles, COM1, COM2 ou COM3

3.2.10**canal de communication**

connexion logique entre un Maître et un Appareil

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.

3.2.11**erreur de communication**

perturbation imprévue du protocole de transmission SDCI

3.2.12**durée de cycle**

temps nécessaire à la transmission d'une séquence M entre un Maître et son Appareil, y compris le temps de repos qui suit

3.2.13**Appareil**

homologue passif unique d'un Maître, tel qu'un capteur ou un actionneur

Note 1 à l'article: "Appareil" avec une majuscule désigne un équipement SDCI, tandis qu'"appareil" avec une minuscule est une désignation générique.

3.2.14**paramètre direct**

paramètre à adressage direct (page) transféré de manière acyclique par le canal de communication de page, sans acquittement

3.2.15

paramètre dynamique

partie du jeu de paramètres d'un Appareil définie par des interfaces utilisateur embarquées, telles que des touches d'autoapprentissage ou des panneaux de commande, en plus des paramètres statiques

3.2.16

Événement

instance d'une variation des conditions dans un Appareil

Note 1 à l'article: "Événement" avec une majuscule désigne un Événement SDCI, tandis qu'"événement" avec une minuscule est une désignation générique.

Note 2 à l'article: Un Événement est indiqué par le drapeau d'Événement dans les informations cycliques d'état de l'Appareil; le transfert acyclique des données d'Événement (en général des informations de diagnostic) est ensuite réalisé par l'intermédiaire du canal de communication de diagnostic.

3.2.17

repli

passage d'un port du mode de commutation codée au mode de signaux de commutation

3.2.18

ISDU

unité de données de service indexée utilisée pour la transmission acyclique avec acquittement de paramètres qui peuvent être segmentés en plusieurs séquences M

Note 1 à l'article: L'abréviation "ISDU" est dérivée du terme anglais développé correspondant "indexed service data unit".

3.2.19

séquence M

séquence de deux messages qui comprend un message du Maître et le message consécutif de l'Appareil

3.2.20

commande de séquence M

premier octet dans un message du Maître qui indique l'opération de lecture/écriture, le type de canal de communication et l'adresse, par exemple un décalage ou une régulation de flux)

3.2.21

type de séquence M

format particulier de séquence M parmi un ensemble de formats spécifiés de séquence M

3.2.22

Maître

homologue actif connecté par l'intermédiaire de ports à un jusqu'à n Appareils, qui fournit une interface avec la passerelle vers les systèmes de communication de niveau supérieur ou vers des automates programmables

Note 1 à l'article: "Maître" avec une majuscule désigne un équipement SDCI, tandis que "maître" avec une minuscule est une désignation générique.

3.2.23

message

<SDCI> séquence de trames UART transférée d'un Maître à son Appareil ou inversement selon les règles du protocole SDCI

3.2.24

données à la demande

données transmises de manière acyclique à la demande de l'application du Maître et constituées de paramètres ou de données d'Événement

3.2.25**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 des procédures de repli.

[SOURCE: ISO/IEC 7498-1, 7.7.2, modifié – le texte descriptif a été extrait du 7.7.2 et la note a été ajoutée.]

3.2.26**port**

<SDCI> interface de support de communication du Maître vers un Appareil

3.2.27**mode de fonctionnement du port**

état d'un port d'un Maître, qui peut être INACTIVE, DO, DI, FIXEDMODE ou SCANMODE

3.2.28**données de processus**

valeurs d'entrée ou de sortie, depuis ou vers un processus d'automatisation discret ou continu, transférées de manière cyclique, avec une priorité élevée et selon un programme configuré, automatiquement après le démarrage d'un Maître

3.2.29**SIO**

mode de fonctionnement d'un port conformément aux entrées et sorties numériques définies dans l'IEC 61131-2, qui est établi après une mise sous tension, un repli ou l'échec de tentatives de communication

3.2.30**signal de commutation**

signal binaire en provenance ou à destination d'un Appareil, en mode SIO (par opposition à la communication SDCI à "commutation codée")

3.2.31**gestion système****SM**

<SDCI> moyen de commande et de coordination des couches de communication internes et des exceptions dans le Maître et ses ports, ainsi que dans chaque Appareil

Note 1 à l'article: L'abréviation "SM" est dérivée du terme anglais développé correspondant "system management".

3.2.32**trame UART**

<SDCI> séquence de bits qui commence par un bit de départ, suivi de huit bits qui transportent un octet de données, suivis par un bit de parité paire, et qui se termine par un bit d'arrêt

3.2.33**réactivation**

procédure qui permet de faire passer un Appareil du mode SIO au mode SDCI

3.2.34

demande de réactivation

WURQ

service de couche physique utilisé par le Maître pour lancer la réactivation d'un Appareil, et le mettre dans un état prêt à recevoir

Note 1 à l'article: L'abréviation "WURQ" est dérivée du terme anglais développé correspondant "wake-up request".

3.3 Termes et définitions relatifs à la SDCI-FS

3.3.1

erreur

écart ou discordance entre une valeur ou une condition calculée, observée ou mesurée et la valeur ou la condition vraie, prescrite ou théoriquement correcte

Note 1 à l'article: Les erreurs peuvent être causées par des erreurs de conception du matériel/logiciel et/ou des informations altérées du fait d'un brouillage électromagnétique et/ou autres effets.

Note 2 à l'article: Les erreurs ne produisent pas nécessairement une *défaillance* ou une *anomalie*.

SOURCE: [IEC 61508-4:2010, 3.6.11, modifié – les notes à l'article ont été ajoutées.]

3.3.2

défaillance

cessation de l'aptitude d'une unité fonctionnelle à accomplir une fonction requise ou à fonctionner comme prévu

Note 1 à l'article: Une défaillance peut être causée par une *erreur* (problème de conception matérielle/logicielle ou rupture de message, par exemple).

SOURCE: [IEC 61508-4:2010, 3.6.4, modifié – les notes ont été supprimées et remplacées par une nouvelle note, et la figure a été supprimée.]

3.3.3

anomalie

condition anormale qui peut entraîner une réduction ou la perte de capacité d'une unité fonctionnelle à accomplir une fonction requise

Note 1 à l'article: L'IEV 191-05-01, définit le terme "fault" (en français "panne") comme un état d'inaptitude à accomplir une fonction requise, en excluant l'inaptitude due à la maintenance préventive, à d'autres actions programmées ou à un manque de ressources extérieures.

SOURCE: [IEC 61508-4:2010, 3.6.1, modifié – la référence à la Figure 4 de l'IEC 61508-4:2010 a été supprimée de la note.]

3.3.4

Appareil FS

homologue passif unique, tel qu'un capteur ou un actionneur de sécurité fonctionnelle, d'un Maître avec des capacités de sécurité fonctionnelle

3.3.5

Maître FS

homologue actif avec des capacités de sécurité fonctionnelle connecté par l'intermédiaire de ports à un jusqu'à n Appareils ou Appareils FS, qui fournit une interface avec la passerelle vers les systèmes de communication de niveau supérieur (NSR ou SR) ou vers des contrôleurs avec des capacités de sécurité fonctionnelle

3.3.6

paramètre FSP

jeu de paramètres pour l'administration et l'utilisation du protocole SDCI-FS

3.3.7**paramètre FST**

jeu de paramètres pour la technologie relative à la sécurité d'un Appareil FS, par exemple rideau de lumière

3.3.8**PDU de sécurité**

unité de données de protocole de sécurité

SPDU

PDU transférée par le biais du canal de communication de sécurité

Note 1 à l'article: L'abréviation "SPDU" est dérivée du terme anglais développé correspondant "Safety Protocol Data Unit".

[SOURCE: IEC 61784-3:2021, 3.1.47, modifié – les notes ont été supprimées et le terme admis "unité de données de protocole de sécurité" a été ajouté.]

3.4 Symboles et abréviations

AIDA	(Automatisierungsinitiative Deutscher Automobilhersteller) initiative d'automatisation des constructeurs automobiles allemands)	
AL	(application layer) couche d'application	
BEP	(bit error probability) probabilité d'erreurs sur les éléments binaires	
C/Q	connexion pour le signal de communication (C) ou de commutation (Q) (SIO)	
CRC	contrôle de redondance cyclique	
DDO	(Device data object) objet de données d'Appareil	
DI	(digital input) entrée numérique	
DIP	(dual in-line package) boîtier à double rangée de connexions	
DL	(data link layer) couche de liaison de données	
DO	(digital output) sortie numérique	
DS	(data storage) stockage de données	
DTI	(Device Tool Interface) interface d'outil d'Appareil	
DTM	(Device Type Manager) gestionnaire de type d'Appareil	[IEC 62453 (toutes les parties)]
FDI	(Field Device Integration) intégration d'appareil de terrain	[IEC 62769 (toutes les parties)]
FDT	(Field Device Tool) outil d'appareil de terrain	[IEC 62453 (toutes les parties)]
FS	(functional safety) sécurité fonctionnelle	
FSCP	(functional safety communication profile) profil de communication de sécurité fonctionnelle (série IEC 61784-3, par exemple [1])	
FS-AI/AO	(functional safety analog input/output) entrée/sortie analogique de sécurité fonctionnelle	
FS-DI/DO	(functional safety digital input/output) entrée/sortie numérique de sécurité fonctionnelle	
FSP	(functional safety protocol) protocole de sécurité fonctionnelle (paramètre)	

FST	(functional safety technology) technologie de sécurité fonctionnelle (paramètre)	
E/S	entrée/sortie	
IODD	(IO Device Description) description d'appareil d'E/S	
IOPD	outil de paramétrage et de diagnostic SDCI	
I/Q	connexion avec plusieurs options: DI, OSSD2e ou DO	
L-	alimentation (-)	
L+	alimentation (+)	
LSO, MSO	(least significant octet) octet de poids faible, (most significant octet) octet de poids fort	
M12	connecteur circulaire	[IEC 61076-2-113]
N24	alimentation supplémentaire 24 V (-); port de classe B	
NSR	(non safety-related) non relatif à la sécurité	
OD	(On-request Data) données à la demande	
OK	"OK"; valeurs correctes ou état correct	
OSSD	(output signal switching device) appareil de commutation du signal de sortie (appareil électronique à autovérification avec OSSD intégré)	[IEC 61496-1]
OSSDe	(output signal switching device) appareil de commutation du signal de sortie (appareil électronique à autovérification avec OSSD intégré conformément au présent document)	
OSSD1/2e	affectation des broches des deux signaux OSSDe	
OSSDm	(output signal switching device) appareil de commutation du signal de sortie (sorties de relais et sorties statiques)	[IEC 60947-5-5]
P24	alimentation supplémentaire 24 V (+); port de classe B	
PD	(Process Data) données de processus	
PDin	données de processus d'entrée de sécurité fonctionnelle (du point de vue du Maître FS)	
PDout	données de processus de sortie de sécurité fonctionnelle (du point de vue du Maître FS)	
PDCT	(port and Device configuration tool) outil de configuration du port et de l'Appareil	
PDU	(protocol data unit) unité de données de protocole	
PFH	(average frequency of a dangerous failure per hour) fréquence moyenne de défaillance dangereuse par heure	[IEC 61508-4]
PID	(program interface description) description de l'interface de programme	
PL	(physical layer) couche physique	
AP	automate programmable	
PS	(power supply) alimentation (mesurée en V)	
RIO	(remote I/O) système d'E/S déporté	
eff.	valeur "efficace"	
SCL	(safety communication layer) couche de communication de sécurité	

SDCI	(single-drop digital communication interface) interface de communication numérique point à point	[IEC 61131-9]
SDCI-FS	(single-drop digital communication interface for functional safety) interface de communication numérique point à point pour la sécurité fonctionnelle	
SIO	(standard input output) entrée/sortie normalisée (mode de commutation numérique)	[IEC 61131-2]
SLM	(safety layer manager) gestionnaire de couche de sécurité	
SM	(system management) gestion système	
SPDU	(safety protocol data unit) unité de données de protocole de sécurité	
SR	(safety-related) relatif à la sécurité	
SSI	(synchronous serial interface) interface série synchrone (en général pour les codeurs)	
TAF	(temporary acknowledgment file) fichier d'acquittement temporaire	
TBF	(temporary backchannel file) fichier de backchannel temporaire	
TPF	(temporary parameter file) fichier de paramètres temporaire	
UART	(universal asynchronous receiver transmitter) émetteur-récepteur asynchrone universel	
UML 2	(unified modeling language) langage de modélisation unifié, édition 2	[ISO/IEC 19505-2]
USB	(universal serial bus) bus de série universel	[www.usb.org]
WURQ	(wake-up request pulse) impulsion de demande de réactivation	
XML	(extensible markup language) langage de balisage étendu	

3.5 Conventions

3.5.1 Descriptions comportementales

Pour les descriptions comportementales, les notations UML 2 sont utilisées, principalement pour les diagrammes d'états et les diagrammes séquentiels (voir [2], [3], [4]).

Les événements qui déclenchent une transition peuvent en général être un signal, un appel de service ou une temporisation. Les conditions logiques (true/false) doivent être le résultat d'un [guard]. Pour améliorer la lisibilité et alléger la maintenance des diagrammes d'états, les diagrammes ne fournissent pas les actions associées à une transition. Ces actions sont répertoriées dans un tableau de transition d'états distinct selon [5].

Les diagrammes d'états représentés dans le présent document sont des descriptions entièrement abstraites. Ils ne constituent pas une spécification complète pour la mise en œuvre.

3.5.2 Ordre des octets de mémoire et de transmission

La Figure 3 indique l'ordre qui doit être utilisé lors du transfert de types de données WORD de la mémoire à la transmission et inversement.

NOTE Les microcontrôleurs existants peuvent différer dans la manière dont les types de données WORD sont stockés dans la mémoire: "gros-boutiste" et "petit-boutiste". Si les conceptions ne tiennent pas compte de ce fait, les octets peuvent être permutés par erreur pour transmission.

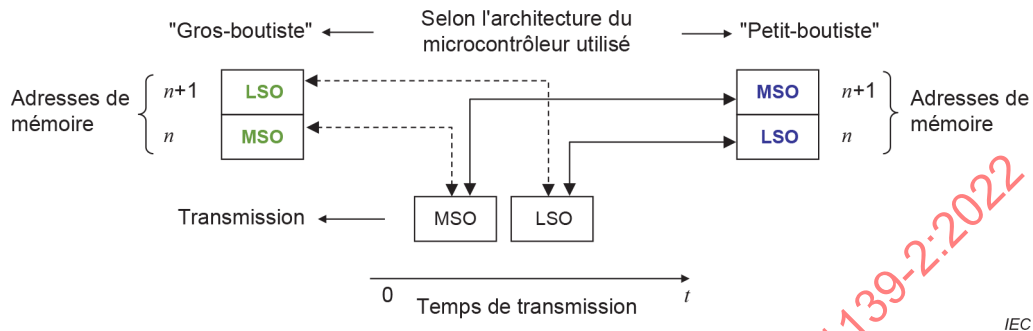


Figure 3 – Ordre des octets de mémoire et de transmission

4 Présentation de la SDCI-FS

4.1 Objet de la technologie et niveaux de fonctionnalité

4.1.1 Technologie SDCI-FS de base

Le présent document spécifie un nouveau protocole de communication de sécurité fonctionnelle affiné, en plus du système de transmission SDCI existant décrit dans l'IEC 61131-9. La Figure 4 représente la façon dont les couches de communication SDCI-FS correspondantes se situent dans les modèles architecturaux du Maître et de l'Appareil, de sorte qu'ils deviennent Maître FS et Appareil FS. La majeure partie de la conception SDCI d'origine reste inchangée pour le présent document.

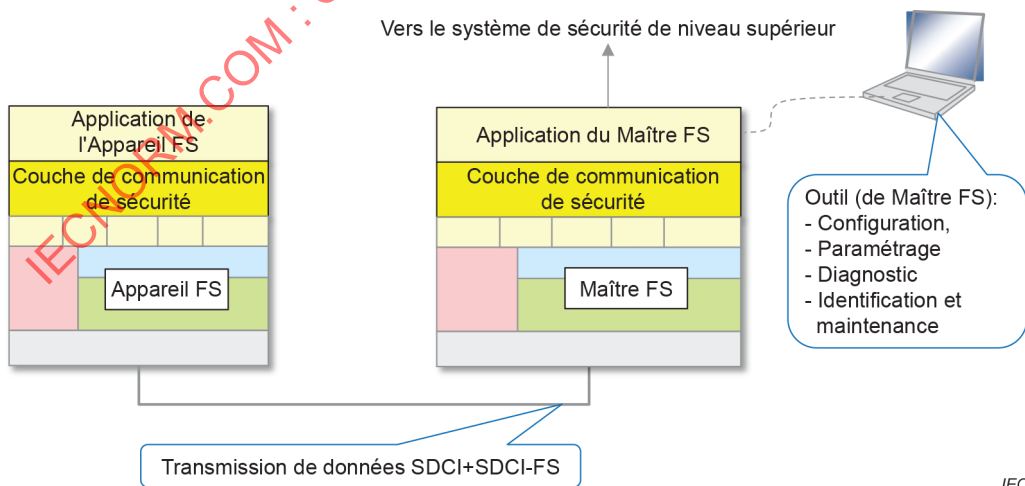


Figure 4 – Modèle de couche de communication SDCI-FS

La couche de communication SDCI-FS s'adapte au protocole de transmission de sécurité fonctionnelle. Ce protocole génère une PDU de sécurité composée de données d'E/S FS, de données de contrôle ou d'état de protocole et d'une signature CRC. La PDU de sécurité ainsi que les données facultatives non relatives à la sécurité sont transmises en tant que données de processus SDCI entre un Maître FS et un unique Appareil FS (point à point). Ceci convient pour des applications de sécurité fonctionnelle jusqu'à SIL3 ou PLe et si la PFH pour une connexion est inférieure à $10^{-9}/h$.

La SDCI-FS augmente le nombre de modes de port et exige donc des modifications de la couche physique et de la gestion de la configuration/du système.

Des modifications sont exigées pour que l'outil (logiciel) du Maître fournisse la configuration et le paramétrage de sécurité nécessaires pour le protocole (paramètres FSP) ainsi que pour la technologie d'Appareil FS particulière (paramètres FST).

La SDCI-FS prend également en charge l'OSSDe en tant que stratégie de migration, comme le mode SIO. Elle ne prend pas en charge

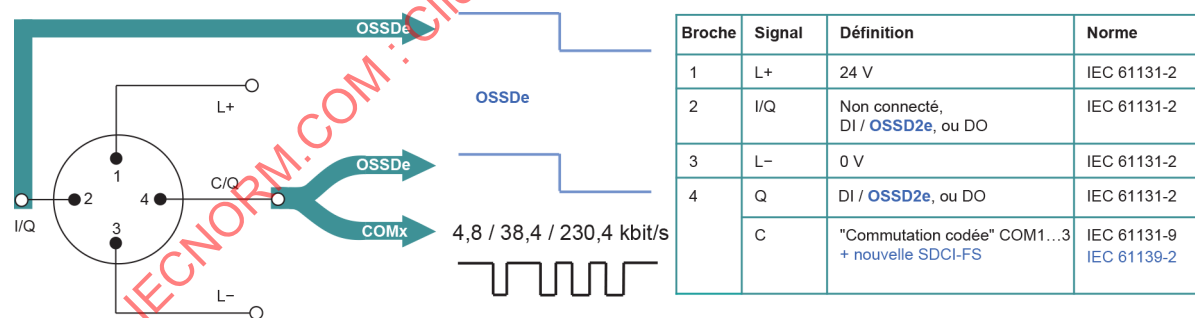
- les connexions sans fil entre le Maître FS et l'Appareil FS (voir Article H.2);
- les systèmes Maître FS/Appareil FS en cascade.

4.1.2 De "analogique" et "commutation" vers la communication

Avec la "sécurité des machines", les états de commutation (on/off) des relais ou des capteurs sont généralement transmis de la même façon qu'avec la SDCI normalisée (SIO) comme un signal de 24 V ou 0 V aux modules FS-DI dans les systèmes d'E/S déportés. Contrairement à la SDCI-FS normalisées, en raison d'exigences de sécurité, ces signaux sont redondants; ils sont soit équivalents (OSSDe = 11→00), soit antivalents (OSSDm = 01→10).

NOTE OSSDe désigne les concepts de l'IEC 61496-1 et OSSDm désigne ceux de l'IEC 60947-5-5 (voir [6]).

Les caractéristiques électriques de l'interface OSSDe sont conformes au type 1 de l'IEC 61131-2 (voir Figure 5).



Légende: OSSDe = signaux de commutation redondants équivalents

IEC

Figure 5 – Extensions d'interface de port pour SDCI-FS

Le mesurage de grandeurs physiques telles que la température, la pression, la position ou la contrainte (modules FS-AI) repose sur plusieurs solutions d'interface, par exemple 4 mA à 20 mA, 0 V à 10 V ou SSI, mais sur aucune technologie de transmission de signal commune (voir Figure , partie gauche).

Les actionneurs tels que les moteurs peuvent être mis hors tension par l'intermédiaire des modules FS-DO et des relais connectés, comme cela est représenté à la Figure 6 (partie gauche).

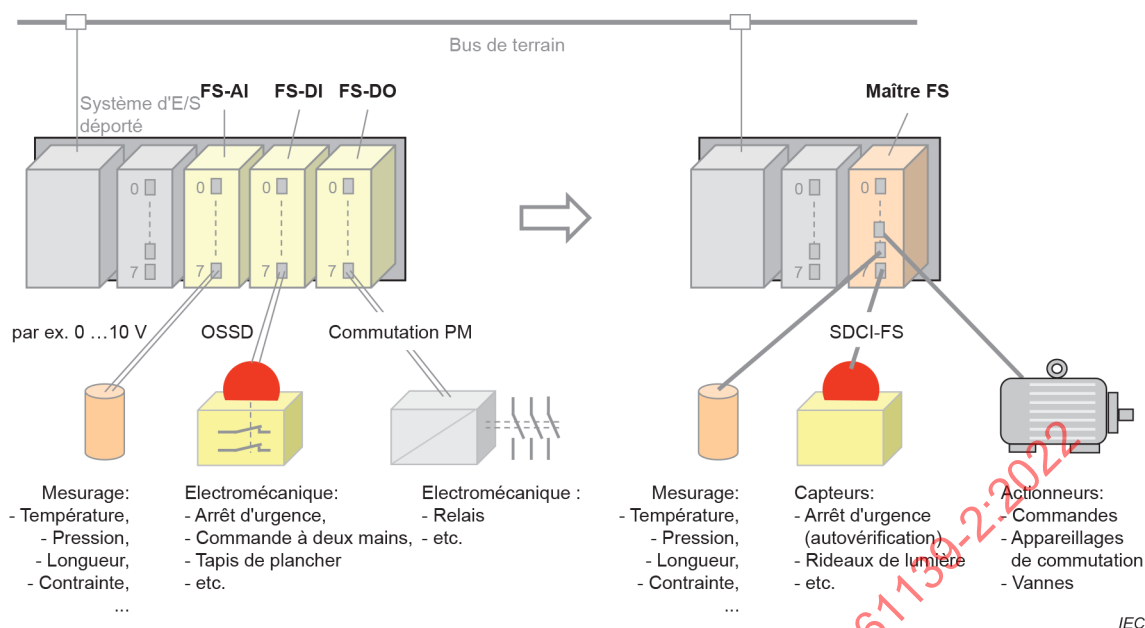


Figure 6 – Migration vers une SDCI-FS

Sans interfaces supplémentaires, il n'était en aucun cas possible de configurer ou de paramétrer les appareils de sécurité ni de recevoir des informations de diagnostic.

La SDCI-FS peut désormais fournir une solution fonctionnelle sûre et fiable pour l'échange des données de processus (états des signaux et valeurs de mesure) par communication numérique point à point (SDCI), ainsi que le paramétrage et le diagnostic (voir Figure , partie droite).

4.1.3 Changement de paradigme réduit de FS-DI à Maître FS

Comme pour les appareils de sécurité actuels pour modules FS-DI (voir Figure 7) et contrairement aux appareils de sécurité fondés sur FSCP, il n'est pas nécessaire:

- a) de configurer un *commutateur de code d'authenticité* ou une *solution logicielle adéquate*;
- b) d'affecter un *temps de fonctionnement du chien de garde*;
- c) d'utiliser un outil logiciel en cas de *remplacement d'Appareil FS*.

L'authenticité est assurée par la vérification de l'Appareil FS adéquat pour le port du Maître FS affecté lors de la mise en service comme modules FS-DI. Cependant, la SDCI-FS fournit un moyen de détecter tout branchement incorrect.

La SDCI-FS utilise un temporisateur de chien de garde pour la transmission des données de sécurité dans les temps (opportunité). Le système est capable de calculer automatiquement le temps de fonctionnement du chien de garde exigé en raison de la nature point à point de la transmission.

Le remplacement d'un Appareil FS sans outils peut être réalisé en utilisant le mécanisme de stockage de données SDCI d'origine.

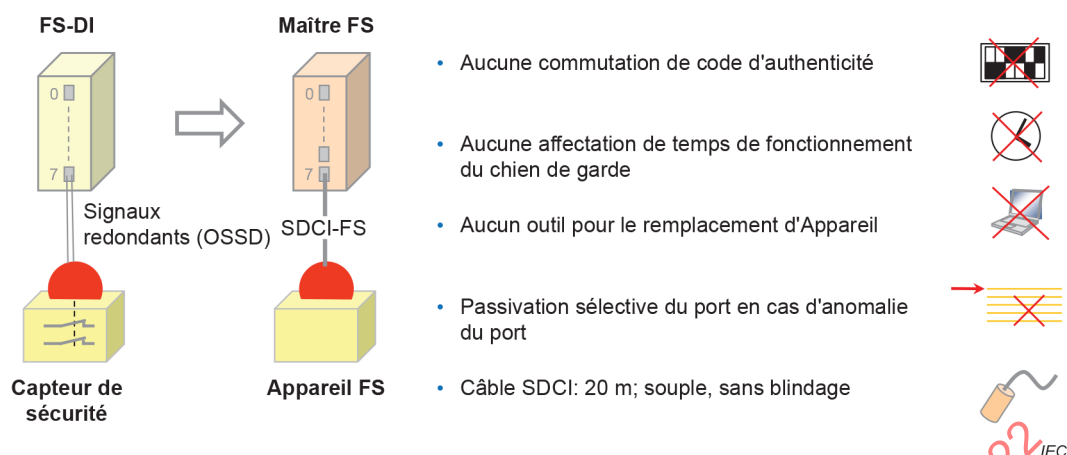


Figure 7 – Changement de paradigme réduit de FS-DI à Maître FS

Le Maître FS prend en charge la *passivation sélective du port* en cas d'anomalie du port et la *passivation granulaire du signal* en cas d'anomalie du canal, par exemple dans une borne d'E/S déportée ("concentrateur") connectée à un port du Maître FS.

Les câbles sont identiques à ceux de la SDCI, c'est-à-dire non blindés et d'une longueur maximale de 20 m. Cependant, en raison du courant d'alimentation admis plus élevé de 1 000 mA par port, la résistance de boucle globale RL_{eff} ne peut être que de 1,2 Ω (voir Tableau 9 et IEC 61131-9).

La conformité aux règles AIDA exige une couleur de câble différente du jaune. Cependant, les connecteurs doivent être de couleur jaune (RAL1004).

4.1.4 Après le paradigme SDCI (SIO/OSSDe)

La SDCI-FS normalisée prend en charge un port de type A (4 broches) sans alimentation supplémentaire et un port de type B (5 broches) avec alimentation supplémentaire 24 V (voir IEC 61131-9). La SDCI-FS gère plusieurs niveaux de spécification "a" à "d" (voir Figure 8). Le nombre de broches fait référence aux broches de Maître FS possibles.

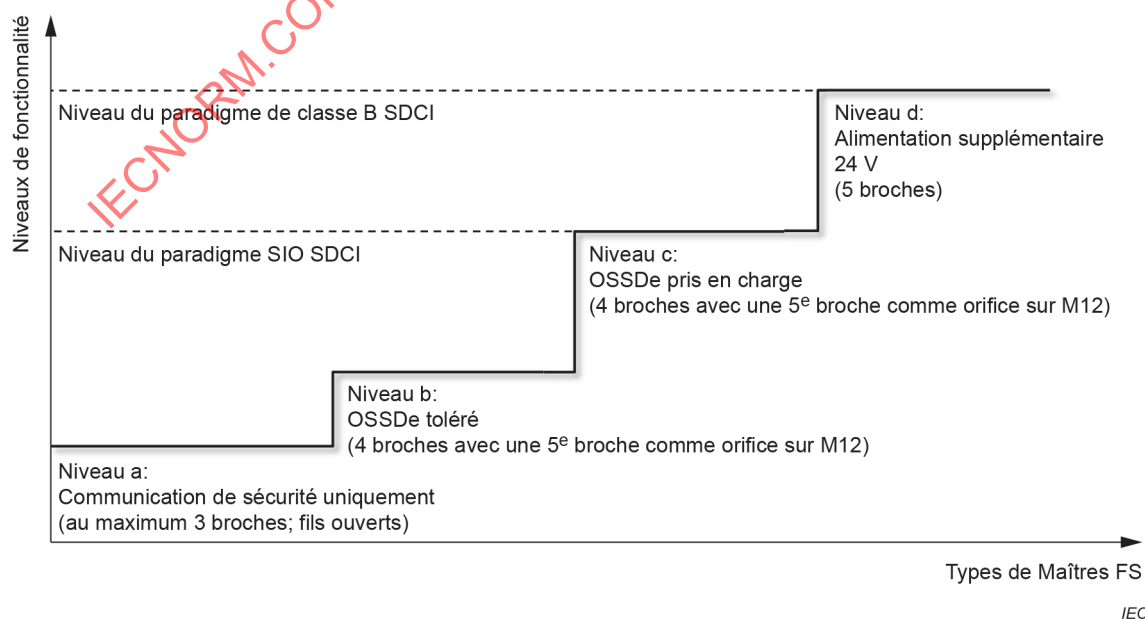


Figure 8 – Types de Maîtres FS et niveaux de fonctionnalité

La disposition d'origine des broches de la SDCI pour les ports de classe A est représentée à la Figure 9, avec les extensions pour les niveaux "a" à "c". Le Tableau 1 donne les détails de ces niveaux.

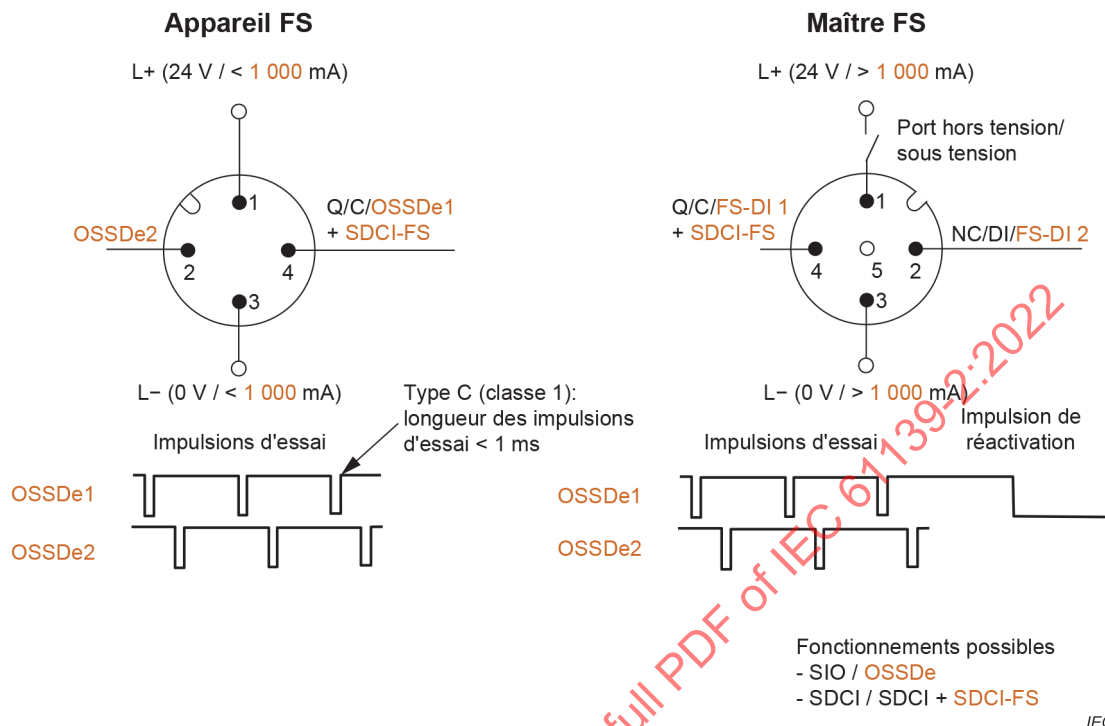


Figure 9 – Disposition d'origine des broches de la SDCI (ports de classe A)

Le niveau "a" assure uniquement la communication (broches 1, 3 et 4). Cela signifie qu'il prend en charge les Appareils FS de type capteur et les Appareils FS de type actionneur.

En raison de la nature redondante de la plupart des interfaces d'appareil de sécurité, la SDCI-FS utilise la broche 2 pour le chemin de signal redondant (OSSDe2, par exemple) et la broche 4 pour le chemin de signal principal (OSSDe1, par exemple)⁵. Ainsi, le niveau "b" permet aux Appareils FS de fournir des sorties OSSDe en plus de la capacité de communication SDCI-FS. Ils peuvent être paramétrés à l'aide d'un "Maître USB" et être connectés à tout module FS-DI en mode de commutation. Lorsqu'ils sont connectés à un Maître FS, les communications de sécurité et les communications non relatives à la sécurité normalisées sont possibles.

Le niveau "c" correspond au niveau SIO du Maître SDCI normalisé. Dans ce cas, le Maître FS prend en charge un mode OSSDe en plus de la communication (broches 1, 3, 4 et 2).

Le Tableau 1 indique la disposition des broches et les modes opérationnels possibles pour les niveaux de fonctionnalité "a" à "c" de l'Appareil FS et du Maître FS avec ports de classe A.

⁵ Les Appareils FS reposent sur l'électronique et non sur les relais. Ainsi, la version électronique de l'OSSDe est prise en compte.

**Tableau 1 – Modes opérationnels des niveaux de fonctionnalité "a" à "c"
(ports de classe A)**

Niveau de fonctionnalité	Appareil FS		Maître FS	
	Broche 2	Broche 4	Broche 2	Broche 4
"a"	- NC, DI, DO	- DI, DO - SDCI - SDCI + SDCI-FS	- NC, DI, DO	- DI, DO - SDCI - SDCI + SDCI-FS
"b"	- NC, DI, DO - OSSD2e	- DI, DO - OSSD1e - SDCI - SDCI + SDCI-FS	- NC, DI, DO	- DI, DO - SDCI - SDCI + SDCI-FS
"c"	- NC, DI, DO - OSSD2e	- DI, DO - OSSD1e - SDCI - SDCI + SDCI-FS	- NC, DI, DO - FS-DI2	- DI, DO - FS-DI1 - SDCI - SDCI + SDCI-FS

La Figure 10 représente la mise en service OSSDe optimisée avec un Maître FS:

- aucun ajustement de filtre en raison d'une longueur d'impulsion d'essai maximale fixe de 1 ms, conformément au type C et à la classe 1 de [7]; et
- aucun ajustement de temps de discordance en raison d'un écart maximal fixe.

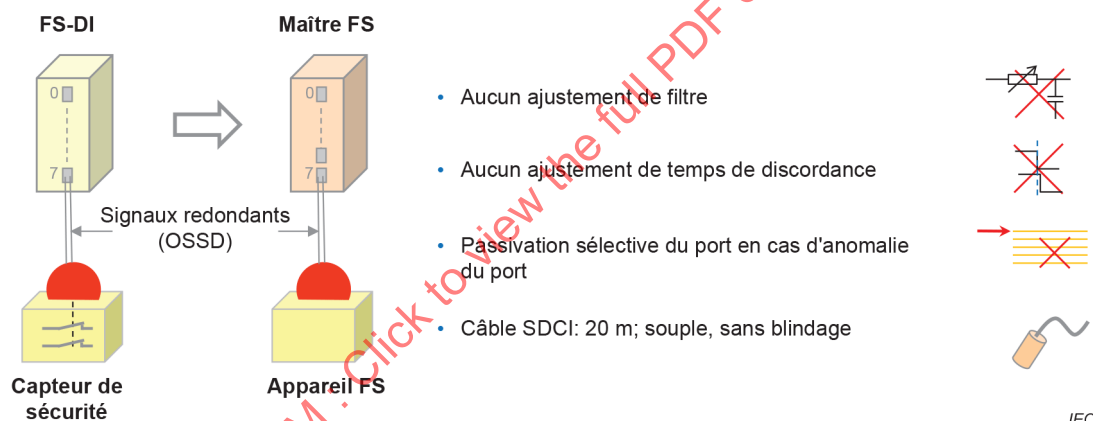


Figure 10 – Mise en service OSSDe optimisée avec un Maître FS

4.1.5 Port de classe B

Un port de classe B fournit une alimentation supplémentaire 24 V pour les actionneurs, laquelle complète l'alimentation principale 24 V de la SDCI. Voir l'IEC 61131-9 pour les contraintes liées à cette alimentation supplémentaire, en particulier l'exigence relative à l'isolation électrique de l'alimentation 2 par rapport à l'alimentation 1.

La Figure 11 représente la disposition des broches, le signal et l'affectation des alimentations, ainsi que le commutateur interne pour L+.

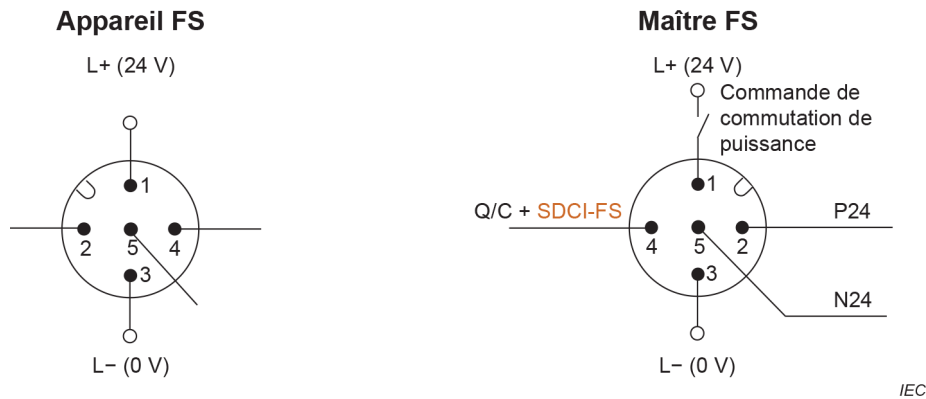


Figure 11 – Niveau "d" d'un Maître FS (classe B)

4.1.6 "Maître USB" avec paramétrage de sécurité

Il est possible d'utiliser des "Maîtres USB" mis à niveau pour la configuration, le paramétrage et l'essai hors site, comme cela est indiqué à la Figure 12.

En raison des exigences de sécurité fonctionnelle, il est nécessaire d'étendre le logiciel de l'outil de Maître pour la configuration de sécurité fonctionnelle et le paramétrage de la technologie de l'Appareil FS (paramètres FST). Voir F.4.1 pour le cas d'utilisation.

Le Tableau 2 indique les types d'appareils qui peuvent être pris en charge par un tel "Maître USB".

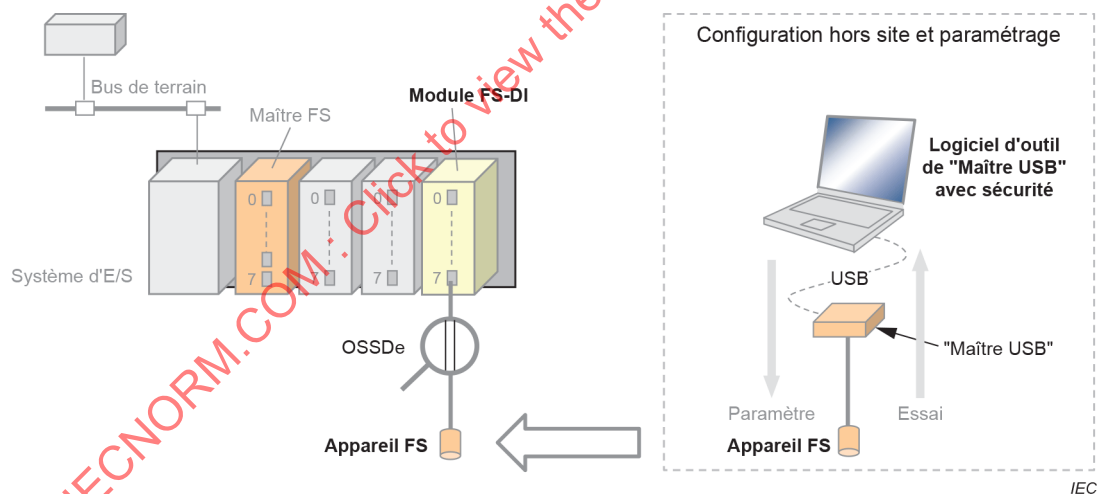


Figure 12 – Configuration et paramétrage hors site

4.1.7 Matrice d'interopérabilité des appareils de sécurité

Le Tableau 2 fournit une vue d'ensemble des capteurs et actionneurs de sécurité types et de leur interopérabilité avec des Maîtres FS de différents niveaux de fonctionnalité, un "Maître USB" mis à niveau pour le paramétrage de sécurité, et des modules FS-DI conventionnels connectés aux FSCP.

Tableau 2 – Matrice d'interopérabilité des appareils de sécurité

Type d'appareil	Maître FS			"Maître USB" avec paramétrage de sécurité	Module FS-DI (FSCP)
	Communication	OSSDe toléré	OSSDe pris en charge		
	"a"	"b"	"c"		
Capteur avec OSSDe ^a	-	-	OSSDe	-	OSSDe
Capteur avec OSSDe et SDCI	-	-	OSSDe	SDCI ^b	OSSDe
Capteur avec OSSDe et SDCI-FS	SDCI-FS	SDCI-FS	OSSDe ou SDCI-FS	SDCI	OSSDe
Capteur avec communication SDCI-FS uniquement, par exemple rideau de lumière	SDCI-FS	SDCI-FS	SDCI-FS	SDCI	
Capteur avec OSSDm, par exemple arrêt d'urgence	-	-	-	-	OSSDm
Actionneur avec SDCI-FS, par exemple entraînement de puissance 400 V, appareillage de commutation à basse tension	SDCI-FS	SDCI-FS	SDCI-FS	SDCI	-
Légende SDCI-FS = échange de données SR et NSR		USB = bus de série universel; actuellement, il s'agit de l'interface la plus courante parmi celles disponibles pour les outils de paramétrage hors site, en raison de sa communication rapide associée à son alimentation.			
^a Disposition des broches selon [8].					
^b La disposition des broches peut varier.					

4.2 Positionnement dans la hiérarchie d'automatisation

La Figure 13 représente le positionnement de la SDCI-FS dans la hiérarchie d'automatisation.

La sécurité classique est fondée sur les relais et semble donc directe, facile à gérer et fiable. Toutefois, les critères qui ont conduit au succès des bus de terrain ont également entraîné le succès des profils de communication de sécurité fonctionnelle (FSCP) en plus des bus de terrain: câblage réduit, paramétrage variable, diagnostic détaillé et flexibilité accrue. La SDCI est le parfait complément à la communication par bus de terrain et comble l'écart avec les capteurs et les actionneurs à moindre coût. Elle assure non seulement la communication, mais également l'alimentation sur un même câble souple et non blindé. Un type de capteur peut être utilisé en mode de commutation traditionnel ou en mode de commutation codée (communication).

La SDCI-FS suit exactement ce paradigme avec son OSSDe.

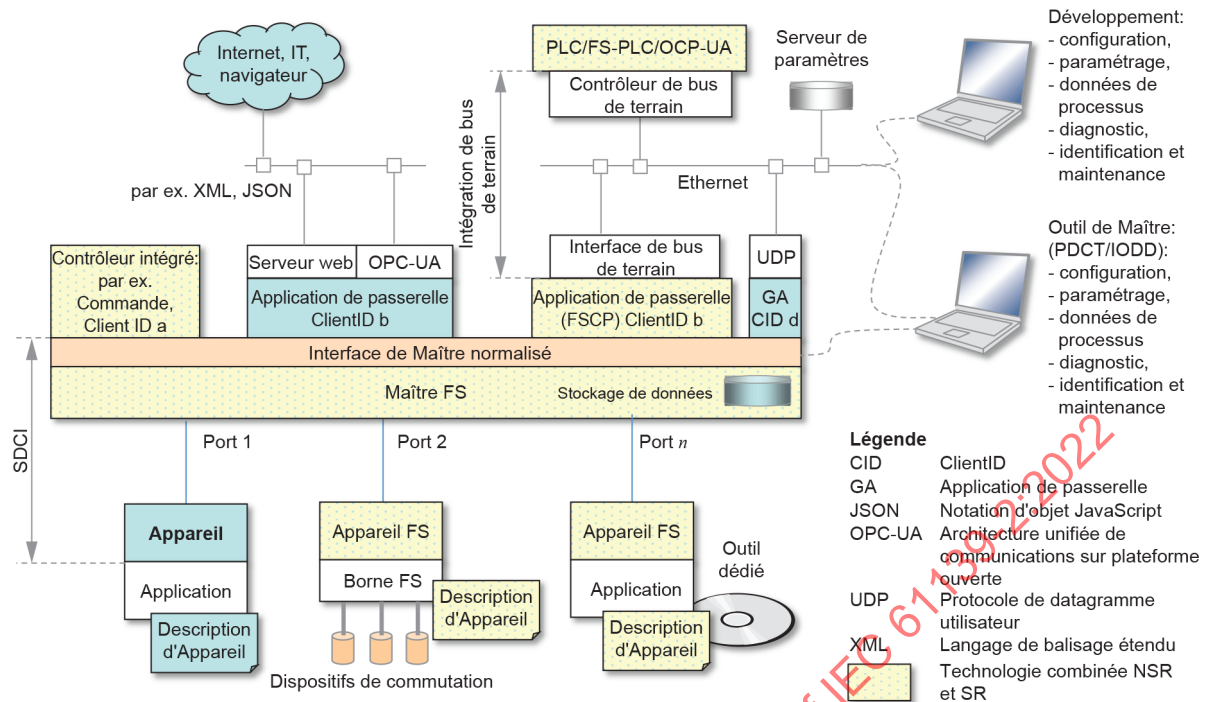


Figure 13 – SDCI-FS dans la hiérarchie d'automatisation

Elle vise deux champs d'application principaux. Le premier consiste à renforcer les fonctions de sécurité dans les communications SDCI-FS et les communications de sécurité fonctionnelle sur les bus de terrain. Le second renforce les fonctions de sécurité "en local" entre un contrôleur de sécurité "intégré" et des capteurs/actionneurs de sécurité en utilisant la communication SDCI-FS.

La SDCI-FS permet de créer des Appareils FS économiques en énergie ("ligne verte"), d'assurer l'autovérification des capteurs de sécurité afin d'éviter les essais annuels, la réduction des types d'interfaces (par exemple 0 V à 10 V, 4 mA à 20 mA, etc.) et une transmission robuste et fiable des informations de sécurité.

Enfin, il s'agit d'une condition préalable aux nouveaux concepts d'automatisation, tels que l'industrie 4.0 ou l'internet des objets (IoT, *Internet-of-Things*).

4.3 Câblage, connecteurs et alimentation

Types de ports de classe A (3 à 4 fils): les câbles et connecteurs spécifiés dans l'IEC 61131-9 pour la classe A peuvent également être utilisés pour la SDCI-FS. Cependant, en raison du courant d'alimentation admis plus élevé de 1 000 mA par port, la résistance de boucle globale RL_{eff} ne peut être que de 1,2 Ω . Aucun blindage n'est exigé.

Types de ports de classe B (5 fils): les câbles, les calibres de fil, les blindages, les courants de commutation maximaux, les interférences, les niveaux de signalisation, etc. ne sont pas spécifiés dans le présent document.

4.4 Relation avec la SDCI

La communication SDCI et son mode SIO sont utilisés comme véhicule de base ("canal noir") pour la SDCI-FS. Outre la SDCI-FS, tout port de Maître FS peut également être configuré pour le fonctionnement SDCI normalisé.

Les entrées de signaux indépendantes du mode SIO sur la broche 2 et la broche 4 sont analysées simultanément par un Maître FS pour obtenir une interface OSSDe. Le résultat est

propagé au système de sécurité de niveau supérieur comme un seul signal de sécurité. Un nouveau gestionnaire de couche de sécurité prend en charge cette fonctionnalité.

Un autre nouveau mode de configuration de port permet la communication SDCI-F. Les diagrammes d'états normalisés ont été légèrement étendus pour prendre en charge

- la détection d'une impulsion Ready depuis l'Appareil FS sur la broche 4;
- la commutation OFF/ON de l'alimentation (broche 1) lorsqu'un Appareil FS manque la séquence de réactivation et lance le fonctionnement de son OSSDe;
- la transmission des paramètres de protocole de sécurité fonctionnelle (FSP) pendant PREOPERATE depuis le Maître FS vers l'Appareil FS;
- l'activation de la couche de communication SDCI-FS (SCL);
- l'activation de l'échange des données de processus FS au sein du gestionnaire de couche de sécurité.

4.5 Fonctionnalités et interfaces de communication

Les données de processus FS depuis et vers un Appareil FS sont toujours encapsulées dans une enveloppe avec code de sécurité qui comprend le numéro de port, un compteur de PDU de sécurité, des informations de contrôle/d'état de protocole et une signature CRC de 16/32 bits. La taille de PDU de sécurité minimale est de 4 octets en l'absence de données de processus FS. La SDCI-FS utilise la séquence M TYPE_2_V.

Seul un sous-ensemble de types de données SDCI est admis: Boolean (encapsulé en tant qu'enregistrement), IntegerT(16) et IntegerT(32).

Le paramétrage dans le domaine de la sécurité des machines nécessite un "outil dédié" par Appareil FS ou famille d'Appareils FS. Une interface d'outil d'Appareil (DTI) fondée sur une technologie éprouvée a été choisie pour les liaisons entre l'outil de Maître FS, l'Appareil FS et son "outil dédié". L'outil de Maître FS doit fournir des moyens de communication à un "outil dédié", afin de permettre la transmission des paramètres de technologie de sécurité (paramètres FST) vers et depuis un Appareil FS. L'"outil dédié" et l'Appareil FS sont tous deux chargés de fournir des moyens suffisants pour sécuriser les données transmises, par exemple par signature CRC ou relecture.

4.6 Paramétrage

La SDCI-FS comprend un concept à trois niveaux. Le premier niveau repose sur l'IODD et contient tous les paramètres de base non relatifs à la sécurité d'un Appareil ou d'un Appareil FS.

Le deuxième niveau exige une extension de l'IODD pour le jeu de paramètres de protocole fixe (FSP). Ces paramètres sont relatifs à la sécurité et sécurisés par signature CRC contre les modifications involontaires du fichier IODD. L'interpréteur de l'outil de Maître FS fournit une extension de sécurité pour le traitement des paramètres FSP. En général, l'outil de Maître FS est capable de déterminer et de suggérer automatiquement les affectations de paramètres FSP (valeurs d'instance) et donc de libérer l'utilisateur de l'affectation initiale de ces valeurs. Il peut vérifier la vraisemblance des valeurs et les modifier si nécessaire.

Le troisième niveau traite des paramètres de sécurité spécifiques à la technologie (FST) d'un Appareil FS. La SDCI-FS classe les Appareil FS en deux types. Le type "de base" n'exige que quelques paramètres FST orthogonaux, tandis que le type "complexe" peut avoir un certain nombre de paramètres FST qui exigent des règles métier et des assistants de vérification ou de validation. En général, ce dernier est déjà fourni avec le logiciel PC existant ("outil dédié") utilisé avec plusieurs profils de communication de sécurité fonctionnelle pour bus de terrain.

Les paramètres FST pour le type "de base" sont codés comme tout paramètre non relatif à la sécurité dans l'IODD. Ils peuvent être modifiés et téléchargés sur l'Appareil FS de manière habituelle. Cependant, une autre solution permet de vérifier l'exactitude de ces affectations. A

la fin d'une session de paramétrage, l'utilisateur lance un "outil dédié" de sécurité (FS-IOPD) pour le calcul d'une signature CRC sur toutes les valeurs d'instance FST fournies par l'outil de Maître FS.

Pour les deux types d'Appareils FS, l'"outil dédié" présente une signature CRC, que l'utilisateur peut copier dans l'un des paramètres FSP. A réception des paramètres FSP au démarrage, l'Appareil FS calcule une signature CRC sur les valeurs d'instance stockées en local et la compare à la signature CRC reçue.

Cette méthode est également utilisée pour la vérification après utilisation du mécanisme de stockage de données SDCI.

4.7 Rôle du Maître FS et de la passerelle FS

Le rôle du Maître FS est étendu à la surveillance sécurisée des données de processus, transférées vers et depuis les Appareils FS, en ce qui concerne l'opportunité, l'authenticité et l'intégrité des données, conformément à l'IEC 61784-3:2021. En ce qui concerne l'authenticité, il utilise le code d'authenticité affecté au Maître FS par le système FSCP de niveau supérieur et le numéro de port. Cela évite les erreurs de connexion liées au port local et les erreurs de connexion lorsque plusieurs Maîtres FS se situent côte à côte.

Un Maître FS peut être équipé d'un contrôleur de sécurité (selon l'IEC 61131-6, par exemple) ou inversement, et ainsi constituer un système de sécurité autonome avec ses propres fonctions de sécurité complètes.

Avec une passerelle FS combinée au Maître FS, les fonctions de sécurité peuvent être établies sur le système FSCP de niveau supérieur en utilisant les capteurs et les actionneurs de sécurité connectés au Maître FS.

4.8 Mapping vers les systèmes de niveau supérieur

La spécification du mapping vers un système FSCP de niveau supérieur relève de l'organisation de bus de terrain. La SDCI-FS est prévue pour satisfaire à la majorité des FSCP au travers, par exemple, d'un nombre réduit de types de données, de descriptions des données d'E/S de sécurité, de la passivation sélective du port et des signaux d'acquiescement de l'opérateur destinés à empêcher le redémarrage automatique des machines.

4.9 Structure du document

La structure du présent document est en grande partie conforme à la structure de l'IEC 61131-9. L'Article 5 spécifie les extensions de la couche physique (PL), notamment les questions relatives à l'OSSDe, le comportement de réactivation et les modes de port supplémentaires. Les extensions du mode SIO sont spécifiées à l'Article 6, celles de la couche de liaison de données (DL) à l'Article 7, celles de la gestion système (SM) à l'Article 8, celles de l'Appareil FS à l'Article 9 et celles du Maître FS à l'Article 10.

La partie centrale du présent document concerne la couche de communication de sécurité (SCL), à l'Article 11. Elle comprend les services SCL, le protocole, les diagrammes d'états et la gestion. En outre, elle traite des mesures d'intégrité, des paramètres de protocole (FSP) et de technologie (FST), de l'intégration des "outils dédiés" au travers de la technologie d'interface d'outil d'Appareil (DTI), de la passivation sélective du port et du diagnostic SCL. L'Article 12 complète cette partie centrale par le traitement de la sécurité fonctionnelle, soit par mapping vers le système de niveau supérieur soit en local.

Les extensions des paramètres et des commandes sont spécifiées à l'Annex A, celles des EventCodes à l'Annex B et celles des types de données à l'Annex C. Les questions relatives au polynôme CRC sont présentées à l'Annex D, les aspects d'IODD à l'Annex E, la technologie d'interface d'outil d'Appareil à l'Annex F, les principaux scénarios à l'Annex G et les exigences système à l'Annex H. L'Annex I fournit des informations concernant l'essai et l'évaluation. Les Annexes A, C, D et E se rapportent à la sécurité.

5 Extensions de la couche physique (PL)

5.1 Vue d'ensemble

La Figure 14 représente la couche physique adaptée d'un Maître FS (classe A).

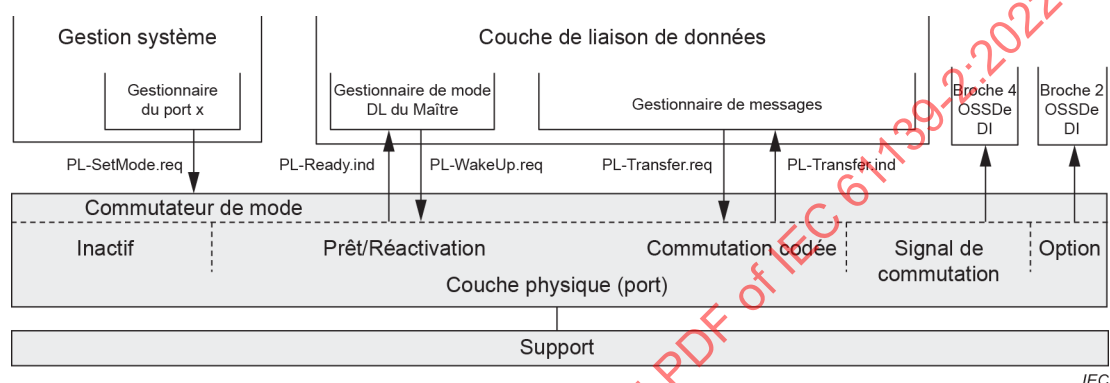


Figure 14 – Couche physique SDCI d'un Maître FS (classe A)

Les broches 2 et 4 doivent être analysées simultanément pour obtenir la fonctionnalité OSSDe. Le Maître FS doit analyser la ligne C/Q pour détecter le signal Ready de l'Appareil FS (voir A.2.11).

La Figure 15 représente la couche physique adaptée d'un Appareil FS (classe A).

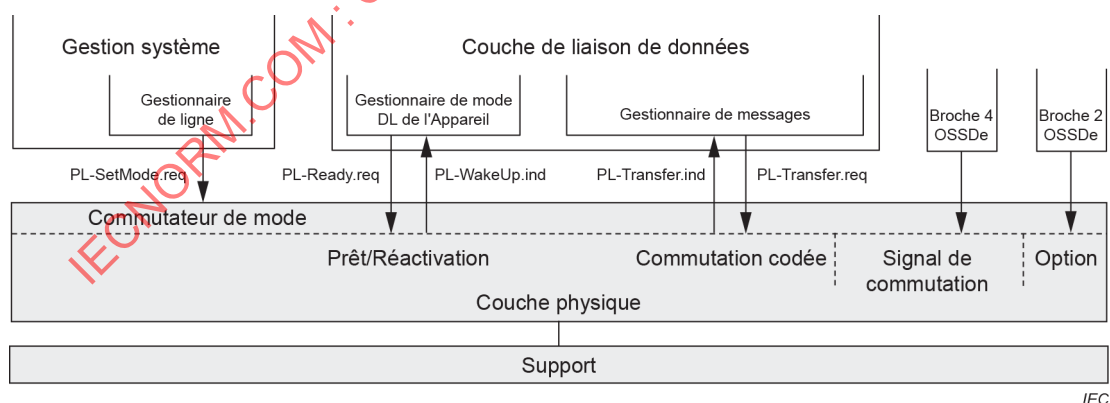


Figure 15 – Couche physique d'un Appareil FS (classe A)

Les broches 2 et 4 transportent les signaux OSSDe. L'Appareil FS doit définir l'impulsion Ready après essai interne.

5.2 Extensions des services PL

5.2.1 PL_SetMode

Le service PL_SetMode est étendu par le TargetMode supplémentaire "OSSDe" (ligne C/Q et ligne I/Q en mode d'entrée numérique).

5.2.2 PL_Ready

Le service PL_Ready lance ou indique un signal Ready sur la ligne C/Q. Chaque fois que l'Appareil FS termine ses essais internes du matériel et du logiciel de sécurité, il définit ce signal. Le Maître FS interroge ce signal et, à réception, lance la séquence de réactivation. Ce service non confirmé n'a aucun paramètre. Les primitives des services figurent dans le Tableau 3.

Tableau 3 – PL_Ready

Nom de paramètre	.req	.ind
<aucun>		

5.3 Emetteur/Récepteur

5.3.1 Hypothèses pour l'extension vers OSSDe

La Figure 16 représente la compatibilité transversale entre les capteurs de sécurité fondés sur OSSD et les Appareils FS fondés sur OSSDe.

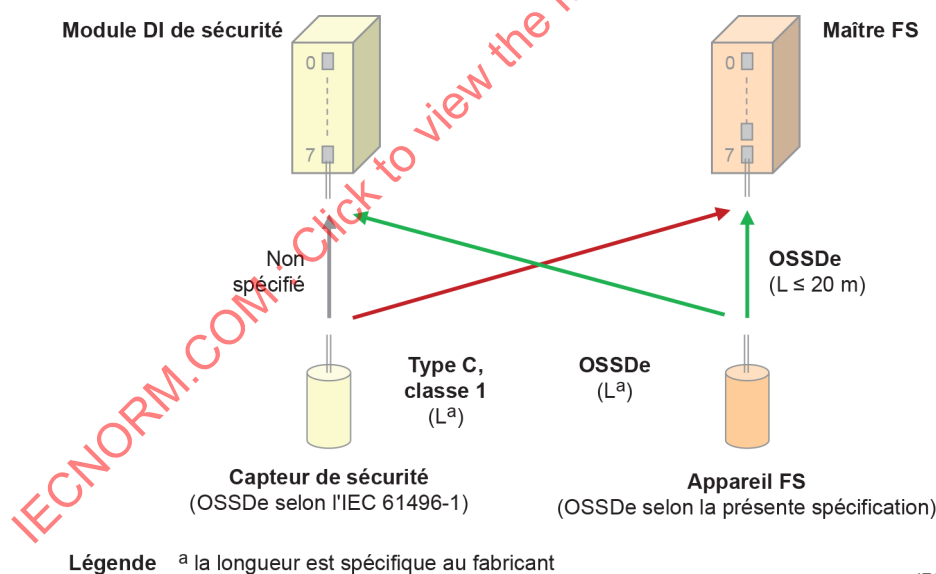


Figure 16 – Compatibilité transversale entre OSSD et OSSDe

Les hypothèses suivantes constituent la base de la conception de l'expansion OSSDe:

- le paradigme SIO de la SDCI doit s'appliquer à la SDCI-FS pour permettre aux fabricants de combiner la fonction OSSDe et la communication SDCI-FS dans un même Appareil FS;
- un port sur le Maître FS (avec "FS-DI" selon la Figure 9) doit avoir des configurations fixes, en tant que SDCI-FS ou qu'interface OSSDe, sans ajustement ou avec des ajustements mineurs en ce qui concerne l'adressage, les temps de fonctionnement du chien de garde, les temps de discordance ou les temps de filtre;

- afin de permettre la connexion des capteurs fondés sur OSSD disponibles sur le marché au Maître FS, l'interface FS-DI doit prendre en charge les ajustements nécessaires pour les appareils de classe "1" de type "C", selon [7];
- l'interface OSSDe doit uniquement être conçue comme une entrée pour le port du Maître FS (capteurs de sécurité, connecteurs de classe A). La plupart des actionneurs sont alimentés par un courant alternatif triphasé, tels que les entraînements de puissance, les appareillages de commutation à basse tension, les démarreurs de moteur, etc.;
- les actionneurs tels que les vannes mixtes et les relais doivent être pris en charge par le Maître FS avec ports de "niveau d" (voir Article 6).

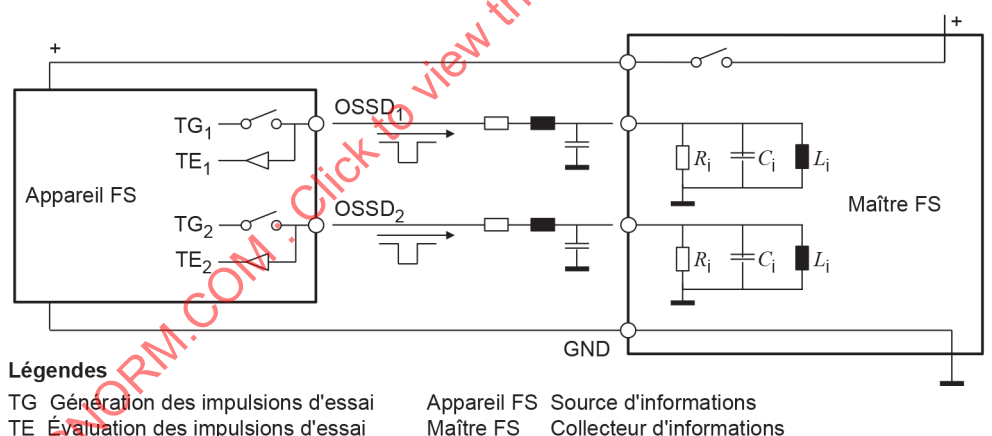
5.3.2 Spécificités OSSDe

5.3.2.1 Généralités

Comme avec l'approche SIO, le Maître FS conforme au niveau "c" prend en charge la connectivité aux appareils de sécurité fonctionnelle existants avec OSSDe. Dans le présent document, OSSDe est défini comme deux sorties de signaux qui commutent de manière équivalente (par opposition à la manière antivalente), un signal étant normalement désactivé et l'autre normalement activé (OSSDm).

Le port du Maître FS est conçu pour atteindre une compatibilité maximale avec les appareils OSSD existants qui utilisent une interface de classe 1 de type C, définie dans [7].

La Figure 17 représente un modèle de référence correspondant repris de [7] et adapté à la SDCI-FS. La "source" d'informations à gauche correspond par exemple à un capteur, tandis que le "collecteur" d'informations à droite représente une entrée du port de Maître FS de classe A. L'électricité est fournie par le collecteur.



IEC

Figure 17 – Principe de fonctionnement OSSDe

Les valeurs les plus défavorables pour la résistance et la capacité de ligne sont définies dans le Tableau 9. Dans le cas de la SDCI-FS, l'inductance de ligne est négligeable à une longueur de 20 m. La conception du port de Maître FS doit assurer des valeurs de R_i , C_i et L_i qui assurent un comportement correct des signaux selon le Tableau 8.

Le Tableau 4 donne les états et conditions OSSD définis dans l'IEC 61496-1:2020.

Tableau 4 – Etats et conditions OSSD

Etat	Cause	Plage de tensions	Courant
OFF	Demande	- 3 V à + 2 V en valeur efficace (+ 5 V en valeur de crête)	< 2 mA (fuite) ^a
ON	Aucune demande	+ 11 V à + 30 V	> 6 mA
^a L'IEC 61131-9 admet 5 mA pour la plage de tensions de 5 V à 15 V			

Etat OFF:

Pour cette interface, l'état OFF est défini comme un état "hors tension", où la tension et le courant d'au moins un OSSDe doivent se trouver dans les limites définies (pour la tension) et au-dessous des limites définies (pour le courant) (voir Tableau 4). Si la fonction de sécurité est demandée ou si la source (l'appareil) détecte une anomalie, les signaux OSSDe doivent basculer vers l'état OFF. Les niveaux de tension antivalents (appelés "discordances") sur les deux sorties OSSDe de l'appareil doivent être traités comme l'état OFF. La durée de cet état doit se situer dans un temps de tolérance de discordance spécifié. Si le temps de tolérance est dépassé, le port est considéré comme défectueux.

Etat ON:

Pour cette interface, l'état ON est défini comme un état "sous tension", où la tension et le courant sur les deux sorties OSSDe doivent se situer dans la plage de tensions et au-dessus des limites de courant définies, lorsqu'ils sont collectés par des entrées IEC 61131-2 (voir Tableau 4). Les impulsions d'essai dans les plages spécifiées pour les niveaux de tension, les durées et les intervalles sont admises. Les niveaux de tension antivalents (appelés "discordances") sur les deux sorties OSSDe de l'appareil doivent être traités comme l'état OFF.

5.3.2.2 Détection des anomalies d'interconnexion

Des essais sont exigés pour la détection des anomalies d'interconnexion spécifiées dans l'IEC 61496-1 et indiquées dans le Tableau 5.

Tableau 5 – Anomalies d'interconnexion

Anomalie	Diagnostics
Court-circuit entre OSSD 1 et OSSD 2	Impulsions d'essai (diagnostic d'exécution)
Court-circuit entre OSSD 1 ou OSSD 2 et V+	Impulsions d'essai (diagnostic d'exécution)
Court-circuit entre OSSD 1 ou OSSD 2 et V-	Impulsions d'essai (diagnostic d'exécution)
Circuit ouvert du câble de retour d'alimentation (V-)	Essai de type, courant de fuite maximal
Circuit ouvert du conducteur fonctionnel de (mise à la) terre	Essai de type, sans mise à la terre fonctionnelle
Circuit ouvert de l'écran du câble blindé	Non exigé en raison de l'absence de blindage
Câblage incorrect	Câblage discret uniquement, problème organisationnel (essai lors de la mise en service)

Les moyens utilisés pour détecter les courts-circuits consistent en des impulsions d'essai à l'exécution. Le moyen utilisé pour soumettre à l'essai le comportement en cas de circuits ouverts consiste en un essai de type pendant l'évaluation. La Figure 18 représente l'approche

par impulsions d'essai pour la détection des anomalies d'interconnexion. Une anomalie d'interconnexion se produit si un canal détecte les deux profils de signaux.

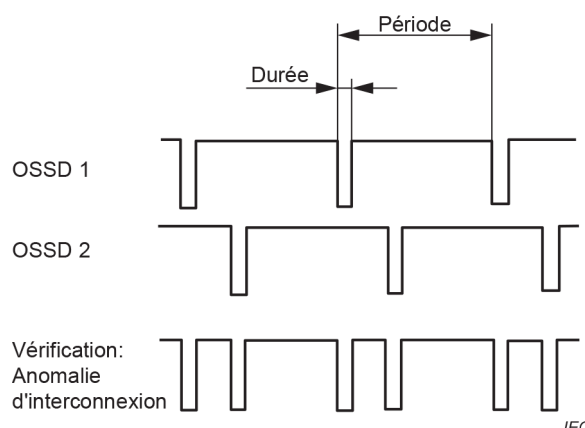


Figure 18 – Impulsions d'essai pour détecter les anomalies d'interconnexion

Trois méthodes d'essais (intervalles) sont communément utilisées:

- impulsions d'essai à chaque cycle de programme de l'appareil de sécurité (dépendance à la configuration);
- impulsions d'essai à instants fixes;
- impulsions d'essai après toute commutation OFF → ON.

5.3.2.3 Essai des sorties OSSDe de l'Appareil FS

Les impulsions d'essai de ce type d'interface pour soumettre à l'essai la ligne de transmission sont créées et évaluées du côté de l'appareil de sécurité. De cette façon, la source peut diagnostiquer le bon fonctionnement de l'étage de sortie. En cas d'erreur détectée, les deux sorties OSSDe doivent être basculées vers l'état de sécurité (condition de blocage = OFF).

Les impulsions d'essai sont créées de manière périodique sur les deux lignes OSSD. Afin de détecter les courts-circuits entre les lignes, ou entre les lignes et l'alimentation, les impulsions d'essai des deux lignes peuvent être décalées dans le temps (voir Figure 19).

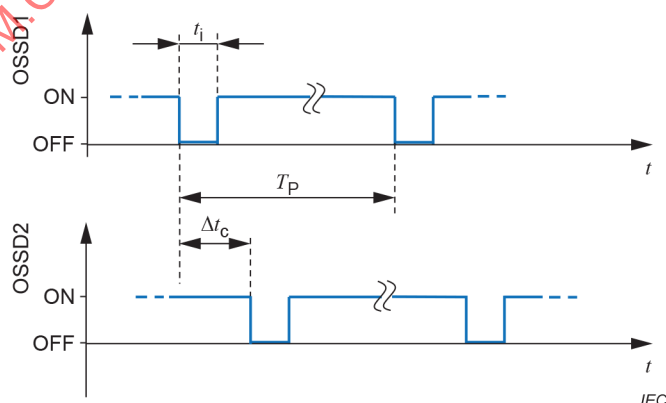


Figure 19 – Temporisations OSSD

Les paramètres suivants spécifient les caractéristiques des impulsions d'essai sur l'interface OSSD:

- période des impulsions d'essai (T_P);

- durée des impulsions d'essai (t_i);
- décalage temporel entre les impulsions d'essai des deux canaux (Δt_c).

Les caractéristiques des impulsions d'essai sont classées dans [7]. Les Appareils FS doivent satisfaire aux exigences du type C et de la classe 1 avec une longueur d'impulsion d'essai $t_i \leq 1\,000\ \mu\text{s}$ (voir Tableau 7).

5.3.3 Démarrage d'un Appareil FS (impulsion Ready)

La Figure 20 représente la séquence de démarrage type d'un capteur OSSD sans capacité SDCI-FS. Lors de l'autovérification de sécurité fonctionnelle, les deux signaux OSSD doivent être OFF. Au terme de l'autovérification, le capteur bascule sur ON et lance les impulsions d'essai. Une demande provoque le basculement du capteur sur OFF. Une anomalie entraîne le basculement du capteur sur la condition de blocage (OFF) et son maintien dans cet état jusqu'à réparation.

NOTE A des fins de simplicité, la Figure 20 ne représente qu'un seul canal OSSD.

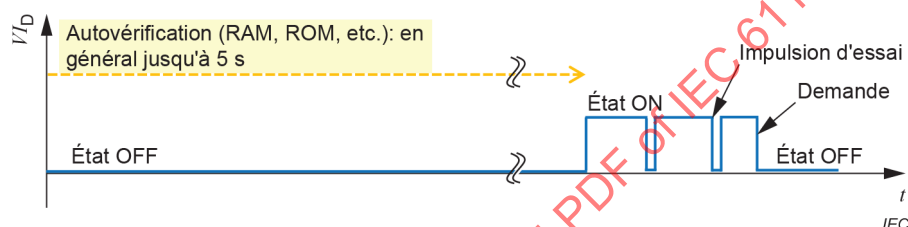


Figure 20 – Démarrage type d'un capteur OSSD

La Figure 21 représente le démarrage d'un Appareil FS avec capacité OSSDe connecté à un module FS-DI classique.

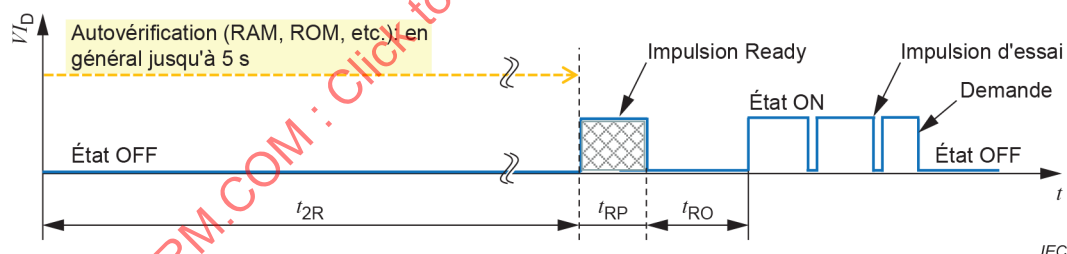


Figure 21 – Démarrage d'un Appareil FS

Contrairement à un capteur classique, l'Appareil FS fournit uniquement sur la broche 4 (voir Figure 9) une impulsion "Ready" d'une longueur donnée pour indiquer au Maître FS qu'il est prêt après une autovérification. Après un temps de rétablissement, l'Appareil FS bascule sur ON et lance les impulsions d'essai comme un capteur de sécurité classique.

Les temporisations et le comportement de réactivation de l'Appareil FS sont spécifiés en 5.7.

5.3.4 Caractéristiques électriques d'un récepteur dans l'Appareil FS et le Maître FS

Les définitions de la plage de tensions et des seuils de commutation sont les mêmes pour le Maître FS et l'Appareil FS, puisque les ports du Maître FS doivent pouvoir fonctionner avec des Appareils SDCI non relatifs à la sécurité. Les définitions du Tableau 6 conformes à l'IEC 61131-9 s'appliquent.

Tableau 6 – Caractéristiques électriques d'un récepteur

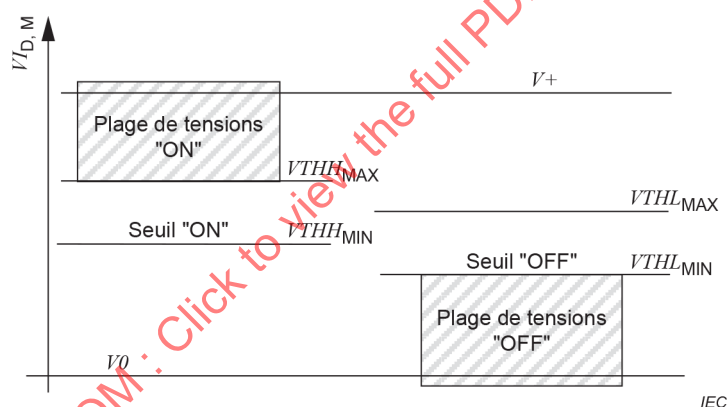
Propriété	Désignation	Minimum	Type	Maximum	Unité	Remarque
$V_{THH_{D, M}}$	Seuil d'entrée "ON"	10,5	n/a	13	V	Voir NOTE 1
$V_{THL_{D, M}}$	Seuil d'entrée "OFF"	8	n/a	11,5	V	Voir NOTE 1
$V_{HYS_{D, M}}$	Hystérésis entre les seuils d'entrée "ON" et "OFF"	0	n/a	n/a	V	Ne doit pas être négative Voir NOTE 2
$V_{IL_{D, M}}$	Plage admissible de tensions "OFF"	$V_{0_{D, M}} - 1,0$	n/a	n/a	V	En référence à la tension d'alimentation négative applicable
$V_{IH_{D, M}}$	Plage admissible de tensions "ON"	n/a	n/a	$V^{+}_{D, M} + 1,0$	V	En référence à la tension d'alimentation positive applicable

NOTE 1 Les seuils sont compatibles avec les définitions des entrées numériques de type 1 de l'IEC 61131-2.

NOTE 2 Tension d'hystérésis $V_{HYS} = V_{THH} - V_{THL}$.

La Figure 22 indique les seuils de commutation pour la détection des signaux OFF et ON.

NOTE "OFF" et "ON" correspondent à "L" (Low, faible) et "H" (High, élevé) dans l'IEC 61131-9.

**Figure 22 – Seuils de commutation pour les récepteurs de l'Appareil FS et du Maître FS**

Le Maître FS ignore les impulsions au-dessous de 11 V (15 mA max. ou 30 mA max.) et d'une durée inférieure à 1 ms.

5.4 Caractéristiques électriques et dynamiques d'un Appareil FS

En général, les valeurs et les plages spécifiées dans l'IEC 61131-9 s'appliquent (voir Figure 23).

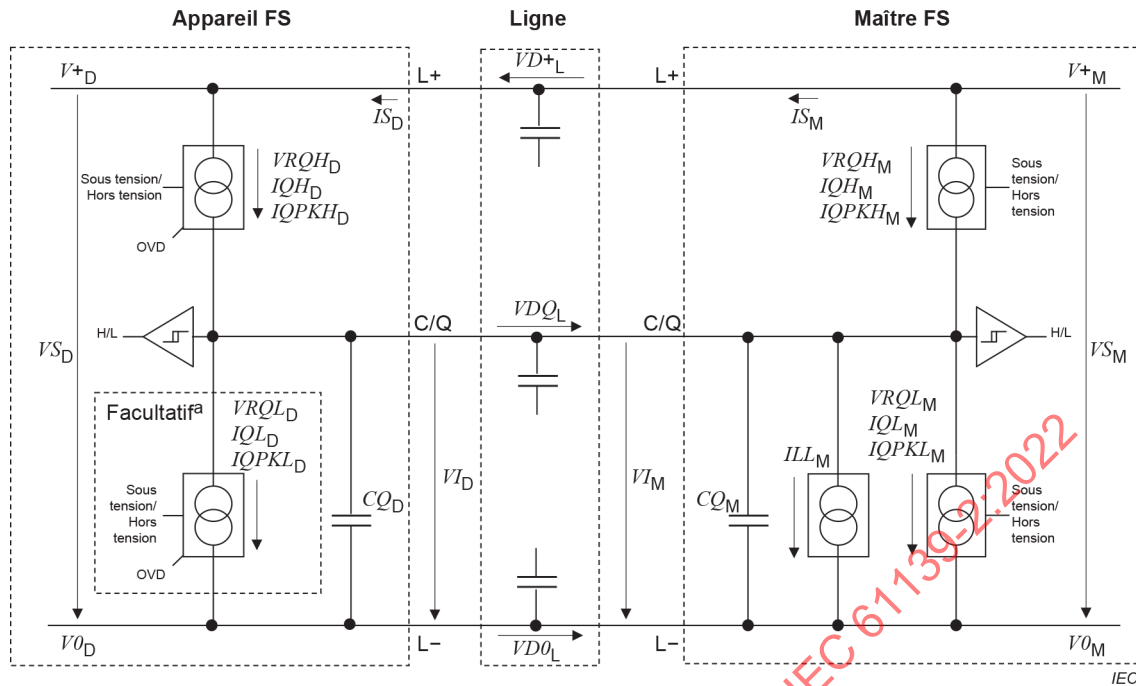


Figure 23 – Schéma de référence (un canal OSSDe)

Les représentations et tableaux de paramètres suivants font référence aux définitions des niveaux de tension de la Figure 24.

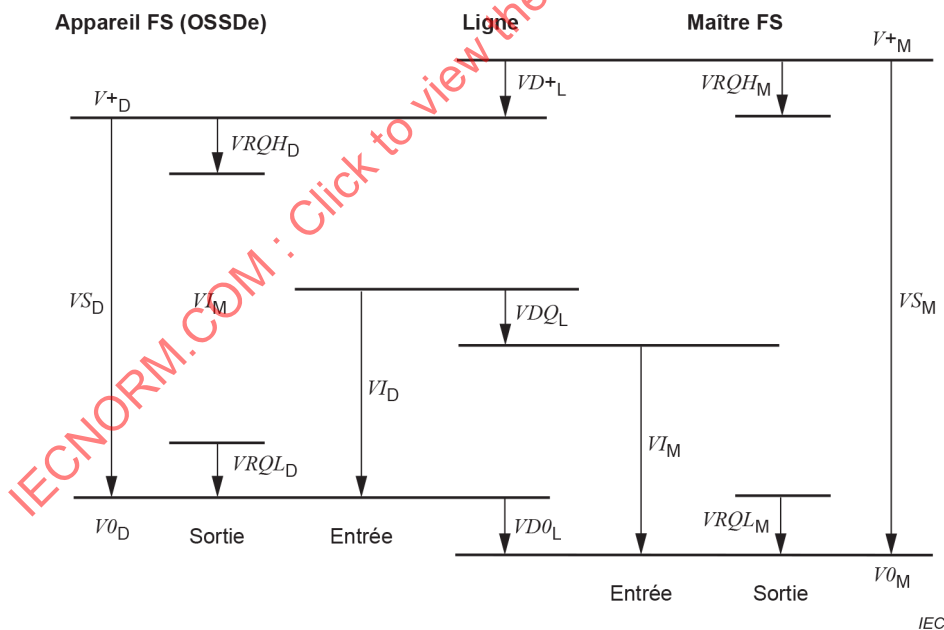


Figure 24 – Définitions des niveaux de tension

Les paramètres électriques et dynamiques de l'interface OSSDe d'un Appareil FS sont spécifiés dans le Tableau 7, conformément à l'IEC 61131-9.

Tableau 7 – Caractéristiques électriques et dynamiques d'un Appareil FS (OSSDe)

Propriété	Désignation	Minimum	Type	Maximum	Unité	Remarque
V_{SD}	Tension d'alimentation	18	24	30	V	Voir Figure 24
ΔV_{SD}	Ondulation	n/a	n/a	1,3	V_{pp}	Les limites des valeurs absolues crête à crête ne doivent pas être dépassées. f_{ripple} = courant continu à 100 kHz
I_{SD}	Courant d'alimentation	n/a	n/a	1 000	mA	Voir 5.9
QI_{SD}	Consommation à la mise sous tension	n/a	n/a	70	mAs	Voir Equation (1) et texte associé
VRQ_{HD}	Tension résiduelle "ON"	n/a	n/a	3	V	Chute de tension par rapport à V_{+D} (voir [9])
VRQ_{LD}	Tension résiduelle "OFF"	n/a	n/a	3	V	Chute de tension par rapport à V_{0D} NOTE 1
IQH_D	Courant d'excitation continu Sortie de commutation P (état "ON")	50	n/a	valeur minimale ($IQPKL_M$)	mA	Valeur minimale due au repli de l'entrée numérique conformément à l'IEC 61131-2, type 2
IQL_D	Courant d'excitation continu Sortie de commutation N (état "ON")	0	n/a	valeur minimale ($IQPKH_M$)	mA	Uniquement pour les étages de sortie d'envoi-réception
IQQ_D	Courant au repos sur V_{0D} (état "OFF")	0	n/a	15	mA	Dépolarisation ou courant résiduel avec étages de pilotes de sortie désactivés
CQ_D	Capacité d'entrée	0	n/a	1,0	nF	Capacité efficace entre C/Q et L+ ou L- de l'appareil en état de réception. Voir l'IEC 61131-9 pour les contraintes sur les taux de transmission.
t_{2R}	Délai jusqu'à l'impulsion Ready	n/a	n/a	5 (par défaut), > 5 à 655 en tant qu'exception	s	Voir Figure 21 et A.2.11
t_{RP}	Durée de l'impulsion Ready	500	n/a	1 000	μ s	Voir Figure 21
t_{RW}	Fin de l'impulsion Ready jusqu'à l'état de préparation pour la réactivation	n/a	n/a	50	μ s	Voir Figure 27
t_{RO}	Fin de l'impulsion Ready jusqu'au mode OSSD	1,1	n/a	Fiche technique	s	Voir Figure 21
t_i/T_P	Rapport d'impulsion d'essai	Fiche technique	n/a	1/100	–	Voir [7] et Figure 19

Propriété	Désignation	Minimum	Type	Maximum	Unité	Remarque
t_i	Durée d'une impulsion d'essai	n/a	n/a	1 000	µs	Voir Figure 19
Δt_c	Décalage temporel	0	n/a	Fiche technique	ms	Voir Figure 19
t_{disD}	Temps de discorde	n/a	n/a	1	ms	Des demandes peuvent survenir pendant les essais

NOTE 1 Dépolarisation de la tension résiduelle avec étage de pilotes de sortie désactivé du côté supérieur et étage de pilotes activé du côté inférieur (par exemple, pilotes d'envoi-réception, si disponibles) avec un courant d'excitation continu externe limité de 50 mA au plus.

NOTE 2 Les caractéristiques du présent tableau prennent pour hypothèse un OSSD de type "C" et de classe "1" ou de classe "2" selon [7] et une interface de type 1 selon l'IEC 61131-2.

Il incombe au concepteur de l'Appareil FS de choisir les ASIC appropriés selon l'IEC 61131-9 et/ou de fournir des circuits d'atténuation pour satisfaire aux exigences de l'IEC 61496-1.

L'Appareil FS doit être capable d'atteindre un état opérationnel stable (prêt pour la réactivation: T_{RDL}) tout en consommation la charge maximale (voir Equation (1)).

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

5.5 Caractéristiques électriques et dynamiques d'un port de Maître FS (OSSDe)

En général, les valeurs et les plages spécifiées dans l'IEC 61131-9 s'appliquent (voir Figure 23 et Figure 24). Les définitions du Tableau 8 sont valides pour les caractéristiques électriques d'un port de Maître FS.

Tableau 8 – Caractéristiques électriques et dynamiques de l'interface du port

Propriété	Désignation	Minimum	Type	Maximum	Unité	Remarque
V_{S_M}	Tension d'alimentation pour Appareils FS	20	24	30	V	Voir Figure 24
I_{S_M}	Courant d'alimentation pour Appareils FS	200	n/a	1 000	mA	Les règles du 5.9 doivent être prises en compte
$ISIR_M$	Capacité d'impulsion de courant pour Appareils FS	400	n/a	n/a	mA	Voir Figure 25
ILL_M	Courant de charge ou de décharge pour 0 V < V_{I_M} < 5 V 5 V < V_{I_M} < 15 V 15 V < V_{I_M} < 30 V	0 5 5	n/a n/a n/a	15 15 15	mA mA mA	Voir NOTE 1
$VRQH_M$	Tension résiduelle "H"	n/a	n/a	3	V	Chute de tension par rapport à V_{+M} au courant d'excitation maximal IQH_M
$VRQL_M$	Tension résiduelle "L"	n/a	n/a	3	V	Chute de tension par rapport à V_{0M} au courant d'excitation maximal IQL_M
IQH_M	Courant d'excitation continu "H"	100	n/a	n/a	mA	–
$IQPKH_M$	Courant de sortie de crête "H"	500	n/a	n/a	mA	Valeur absolue Voir NOTE 2
IQL_M	Courant d'excitation continu "L"	100	n/a	n/a	mA	–
$IQPKL_M$	Courant de sortie de crête "L"	500	n/a	n/a	mA	Valeur absolue Voir NOTE 2
CQ_M	Capacité d'entrée	n/a	n/a	1,0	nF	$f = 0$ MHz à 4 MHz
t_{disM}	Temps de discordance	n/a	n/a	3	ms	Voir NOTE 3 et Tableau 7
t_i	Durée d'une impulsion d'essai	n/a	n/a	1	ms	Voir Figure 19
T_P	Période des impulsions d'essai	10	n/a	n/a	ms	Voir Figure 19

NOTE 1 Les courants sont compatibles avec la définition des entrées numériques de type 1 de l'IEC 61131-2. Toutefois, pour la plage 5 V < V_{I_M} < 15 V, le courant minimal est de 5 mA au lieu de 2 mA, afin d'atteindre des vitesses de variation suffisamment courtes pour les Appareils à commutation p exclusivement.

NOTE 2 Courant de demande de réactivation (voir IEC 61131-9:—, 5.3.3.3).

NOTE 3 Les caractéristiques prennent pour hypothèse un OSSD de type "C" et de classe "1" selon [7].

Le Maître doit fournir une charge d'au moins 20 mA dans les 50 ms qui suivent un temps de mise sous tension de 5 ms sans arrêt de surcharge (voir Figure 25). Après 50 ms, les limites de courant pour I_{S_M} selon le Tableau 8 s'appliquent.

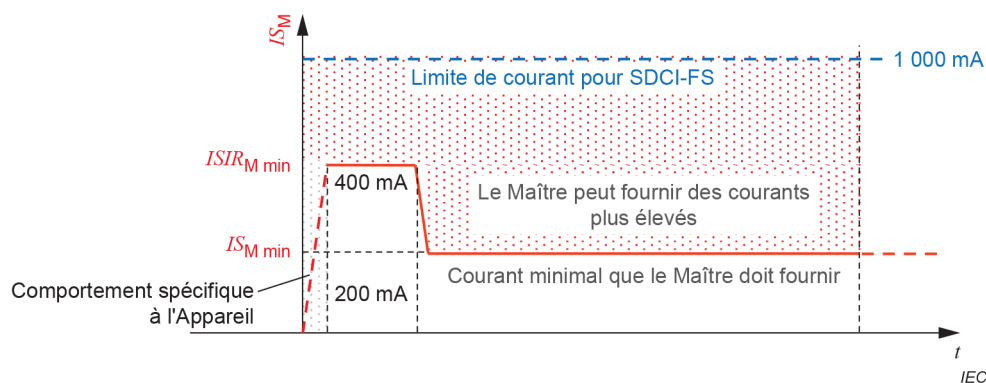


Figure 25 – Capacité de charge à la mise sous tension

5.6 Interface FS-DI du port du Maître FS

Etant donné que les capteurs de sécurité OSSD peuvent fournir différents profils d'impulsions d'essai, le port du Maître FS doit disposer d'un filtre d'entrée approprié ou d'un algorithme d'évaluation. Aux fins de la CEM, une combinaison des deux peut être utilisée. Cela signifie que le laps de temps pendant lequel le signal est inférieur à U_{Hmin} doit être inférieur à la durée maximale admise de l'impulsion d'essai.

Tout état différent des deux signaux "élevés", à l'exception des impulsions d'essai, doit être interprété comme un état de sécurité.

Les niveaux de CEM doivent être pris en compte pour la disposition d'un filtre d'entrée. Le taux de transmission de communication de 230 kbit/s n'est pas compatible avec le filtre d'entrée. Une résolution possible de ce conflit est représentée à la Figure 26.

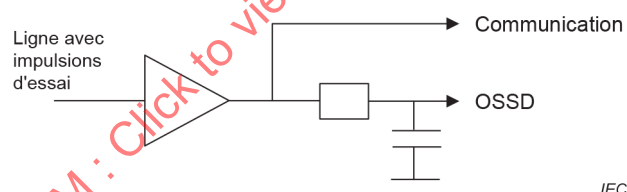


Figure 26 – Résolution du conflit avec le filtre d'entrée OSSDe

En général, les valeurs et les plages spécifiées dans l'IEC 61131-9 s'appliquent. L'interface de type 1 selon l'IEC 61131-2 constitue la base. Les paramètres électriques et dynamiques divergents et supplémentaires pour les interfaces FS-DI sont spécifiés dans le Tableau 8.

5.7 Coordination de la réactivation

La Figure 27 représente le démarrage d'un Appareil FS (voir IEC 61131-9 pour les définitions des temporisations normalisées). Après les autovérifications, l'Appareil indique qu'il est prêt pour la réactivation par l'intermédiaire d'une impulsion ON/Ready sur la ligne C/Q (voir A.2.11). Si aucune réactivation ne se produit dans un délai défini, il lance les impulsions d'essai (voir Figure 20).

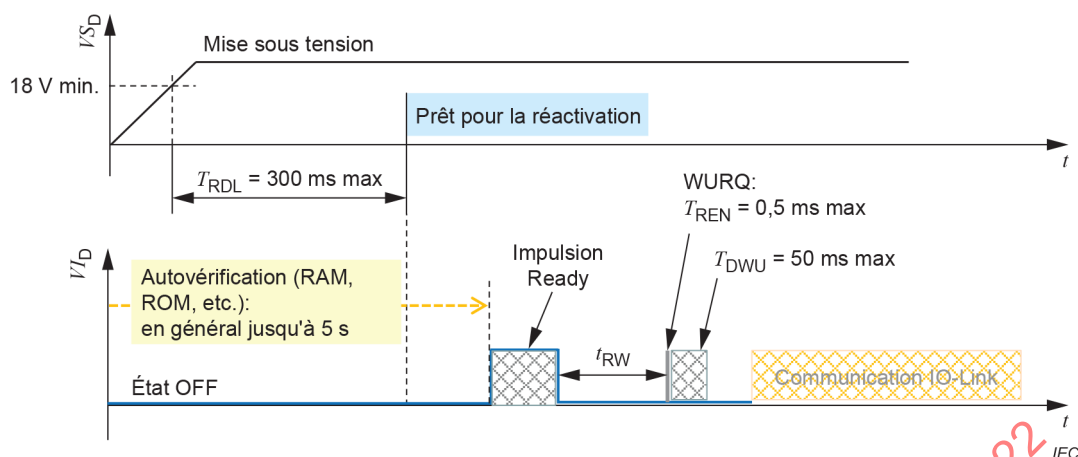


Figure 27 – Démarrage d'un Appareil FS

NOTE En réalité, certains fournisseurs de rideaux de lumière de sécurité proposent l'activation de la fonctionnalité si certaines conditions de connexion sont activées au cours de la phase de démarrage (écrasement, par exemple).

5.8 Démarrage rapide

La Figure 28 représente les exigences pour certaines applications de sécurité fonctionnelle, par exemple pour un changeur d'outils sur un robot. Un démarrage dit "rapide" doit être réalisé dans un délai de 0,5 s dans les cas non relatifs à la sécurité et dans un délai de 1 s dans les cas de sécurité fonctionnelle.

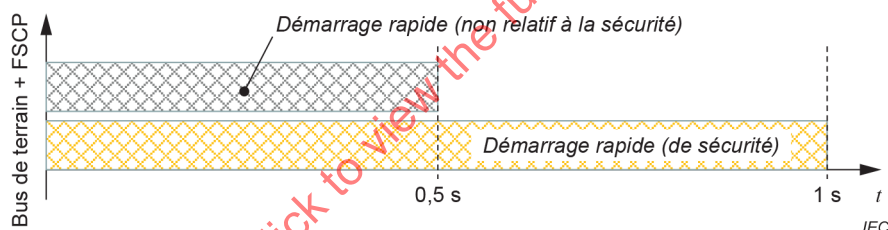


Figure 28 – Temporisations de démarrage rapide exigées

NOTE Les appareils de sécurité actuels nécessitent généralement jusqu'à 5 secondes d'autovérification avant de basculer en sécurité fonctionnelle.

Le concept d'impulsion Ready permet de satisfaire plus facilement à ces exigences (voir A.2.11).

5.9 Alimentation

Un port de Maître FS doit être capable d'activer et de désactiver son alimentation. Cela permet au Maître FS de redémarrer un Appareil FS lorsqu'il ne réussit pas à établir la communication et lance le fonctionnement OSSDe à la place.

Le port du Maître FS est la seule source d'alimentation des parties de l'Appareil FS liées à la SDCI. Toute source d'alimentation externe de l'Appareil FS doit être totalement non réactive à ces parties.

Le Maître FS doit fournir à tous les ports une alimentation minimale de 200 mA et au moins à un port une alimentation minimale de 1 000 mA. Le Maître FS doit spécifier la consommation de courant maximale totale de tous ses ports ainsi que les règles de dégradation.

Les courants supérieurs peuvent entrer en conflit avec les composants de commutation de puissance et causer des interférences avec les lignes de signaux. L'exigence d'"ondulation" donnée dans le Tableau 7 doit être prise en compte. La résistance de boucle globale du câble ne doit pas dépasser 1,2 Ω (voir Tableau 8 et Tableau 9).

5.10 Support

5.10.1 Contraintes

Par souci de simplicité de la technologie et de la mise en service, la SDCI-FS demande une connexion point à point câblée ou une transmission et une alimentation cohérentes équivalentes entre le Maître FS et l'Appareil FS. Aucun élément de stockage n'est admis entre les deux.

5.10.2 Connecteurs

Les connecteurs spécifiés dans l'IEC 61131-9 pour la classe A sont admis.

5.10.3 Caractéristiques des câbles

Le Tableau 9 donne les caractéristiques des câbles pour les Appareils SDCI-FS et les Appareils non relatifs à la sécurité si des courants d'alimentation supérieurs à 200 mA sont appliqués.

Tableau 9 – Caractéristiques des câbles

Propriété	Désignation	Minimum	Type	Maximum	Unité
L	Longueur de câble	0	n/a	20	m
RL_{eff}	Résistance de boucle globale	n/a	n/a	1,2	Ω
CL_{eff}	Capacité de ligne efficace	n/a	n/a	3,0	nF (< 1 MHz)
NOTE Ces caractéristiques peuvent s'écarter des caractéristiques d'origine définies dans l'IEC 61131-9.					

6 Extensions du mode SIO

Le mode SIO est uniquement défini pour la broche 4 du port du Maître/de l'Appareil dans l'IEC 61131-9. OSSD exige l'inclusion de la broche 2 spécifiée à l'Article 5. La configuration peut être effectuée au sein de la couche d'application du Maître/de l'Appareil (voir Figure 32 et Figure 34).

7 Extensions de la couche de liaison de données (DL)

7.1 Vue d'ensemble

La Figure 32 et la Figure 34 représentent les blocs fonctionnels DL de l'Appareil FS et du Maître FS. Aucun nouveau service ni aucun service modifié n'est exigé. Toutefois, les deux gestionnaires de mode DL sont étendus par la fonctionnalité d'impulsion Ready, comme cela est indiqué en 7.2 et en 7.3.

7.2 Diagramme d'états du gestionnaire de mode DL du Maître FS

La Figure 29 représente les modifications du gestionnaire de mode DL du Maître FS par rapport au gestionnaire de mode DL du Maître de l'IEC 61131-9.

Un nouvel état "WaitOnReadyPulse_10" prend en compte l'exigence qui impose au Maître FS d'attendre l'impulsion Ready d'un Appareil FS (voir 5.7) avant d'établir la communication par DL_SetMode_STARTUP.

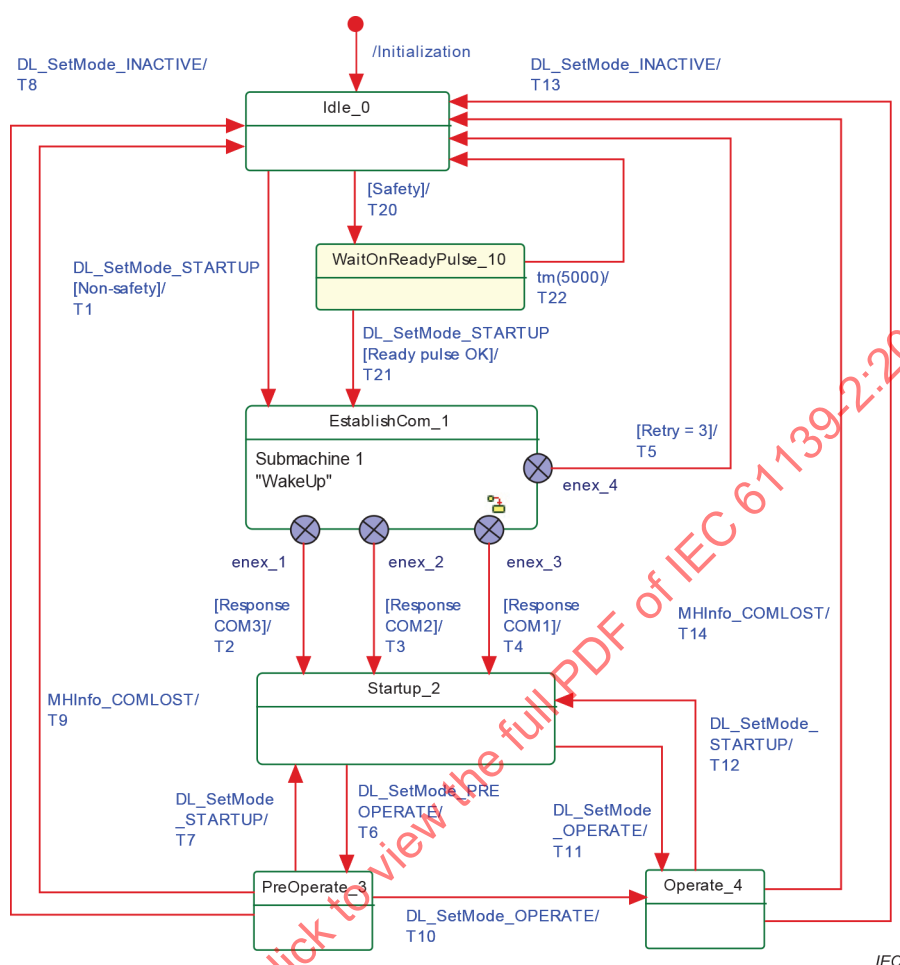


Figure 29 – Diagramme d'états du gestionnaire de mode DL du Maître FS

Le délai d'attente maximal est t_{2R} , comme cela est défini dans le Tableau 7. Chaque fois que le délai expire, le Maître FS doit exécuter un cycle de mise hors tension/sous tension de l'Appareil FS connecté, afin de procéder à la relance d'une autre impulsion Ready.

Le critère d'utilisation du chemin supplémentaire est le protecteur [safety], qui est issu de la nouvelle configuration de port "FS_PortModes" (voir 10.4.2).

Le Tableau 10 indique l'état et les transitions supplémentaires ainsi que les éléments internes relatifs à la fonctionnalité d'impulsion Ready.

Tableau 10 – Tableaux de transition d'états du gestionnaire de mode DL du Maître FS

NOM D'ÉTAT		DESCRIPTION DE L'ÉTAT	
Idle_0 à SM: Retry_9		Voir l'IEC 61131-9:—, Tableau 42	
WaitOnReadyPulse_10		Attente de l'impulsion Ready de l'Appareil FS. Un temporisateur est lancé avec la valeur donnée du paramètre FSP_TimeToReady dans FSPortConfigList (voir A.2.11 et 10.3.4).	
TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T1 à T19	*	*	Voir l'IEC 61131-9:—, Tableau 42
T20	0	10	Ce chemin n'est emprunté que si le nouveau paramètre de configuration "Safety" a été affecté à "SafetyCom" ou "MixedSafetyCom", respectivement.
T21	10	1	Définir Retry = 0.
T22	10	0	Le Maître FS n'a pas été en mesure de détecter une impulsion Ready dans les 5 s. Il lance un cycle de mise hors tension/sous tension de l'Appareil FS, afin de relancer l'impulsion Ready.
ÉLÉMENTS INTERNES		TYPE	DÉFINITION
MH_xxx à xx_Conf...		Appel	Voir l'IEC 61131-9:—, Tableau 42
Safety		Protecteur	Nouveau paramètre de configuration "Safety": soit "SafetyCom" soit "MixedSafetyCom"
Impulsion Ready OK		Protecteur	Impulsion Ready détectée

7.3 Diagramme d'états du gestionnaire de mode DL de l'Appareil FS

La Figure 30 représente les modifications du gestionnaire de mode DL de l'Appareil FS par rapport au gestionnaire de mode DL de l'Appareil de l'IEC 61131-9.

Un nouvel état "SelfTesting_5" prend en compte l'exigence qui impose à l'Appareil FS d'indiquer qu'il est prêt pour la procédure de réactivation après son autovérification de sécurité interne par l'intermédiaire d'une impulsion d'essai sur la broche 4. En réalité, l'autovérification peut durer plus longtemps que le temps de démarrage maximal admis T_{RDL} d'un Appareil non relatif à la sécurité (voir 5.7).

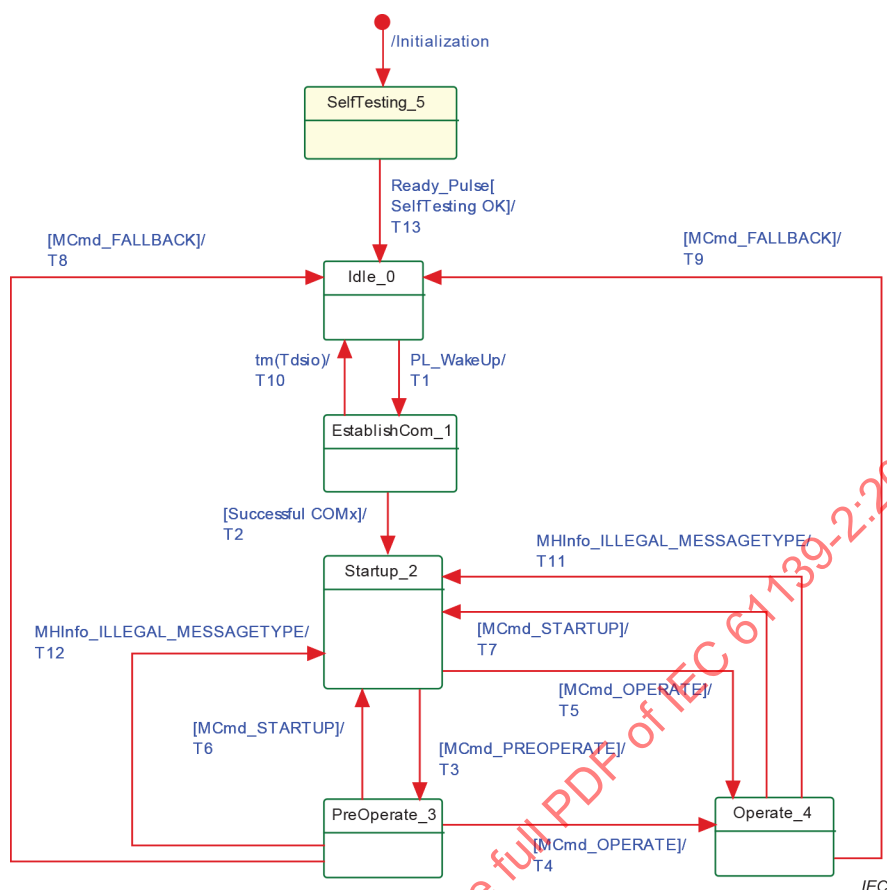


Figure 30 – Diagramme d'états du gestionnaire de mode DL de l'Appareil FS

Le Tableau 11 indique l'état et les transitions supplémentaires ainsi que les éléments internes relatifs au temps d'autovérification d'un Appareil FS.

Tableau 11 – Tableaux de transition d'états du gestionnaire de mode DL de l'Appareil FS

NOM D'ÉTAT		DESCRIPTION DE L'ÉTAT	
Idle_0 à Operate_4		Voir l'IEC 61131-9:—, Tableau 43	
SelfTesting_5		Contrôle de sécurité par autovérification du μ C, de la RAM, etc. L'opération peut durer plus longtemps que le temps de démarrage admis T_{RDL} d'un Appareil non relatif à la sécurité (voir A.2.11).	
TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T1 à T12	*	*	Voir l'IEC 61131-9:—, Tableau 43
T13	5	0	Créer un signal (Ready_Pulse) sur la broche 4 pendant t_{RP} lorsque l'autovérification est terminée (voir t_{2R} dans le Tableau 7).
ÉLÉMENTS INTERNES		TYPE	DÉFINITION
T_{RDL}		Durée	Voir l'IEC 61131-9:—, Tableau 10
t_{RP}		Durée	Voir Tableau 7
Self-testing OK		Protecteur	Autovérification terminée

8 Extensions du gestionnaire de configuration (CM) du Maître

Une partie des mesures d'intégrité est un enregistrement de vérification (VerifyRecord) qu'un Maître FS envoie à l'Appareil FS pendant le démarrage, comme cela est expliqué en 11.7.4 et représenté à la Figure 57. Ceci exige une extension du gestionnaire de configuration, comme cela est représenté à la Figure 31.

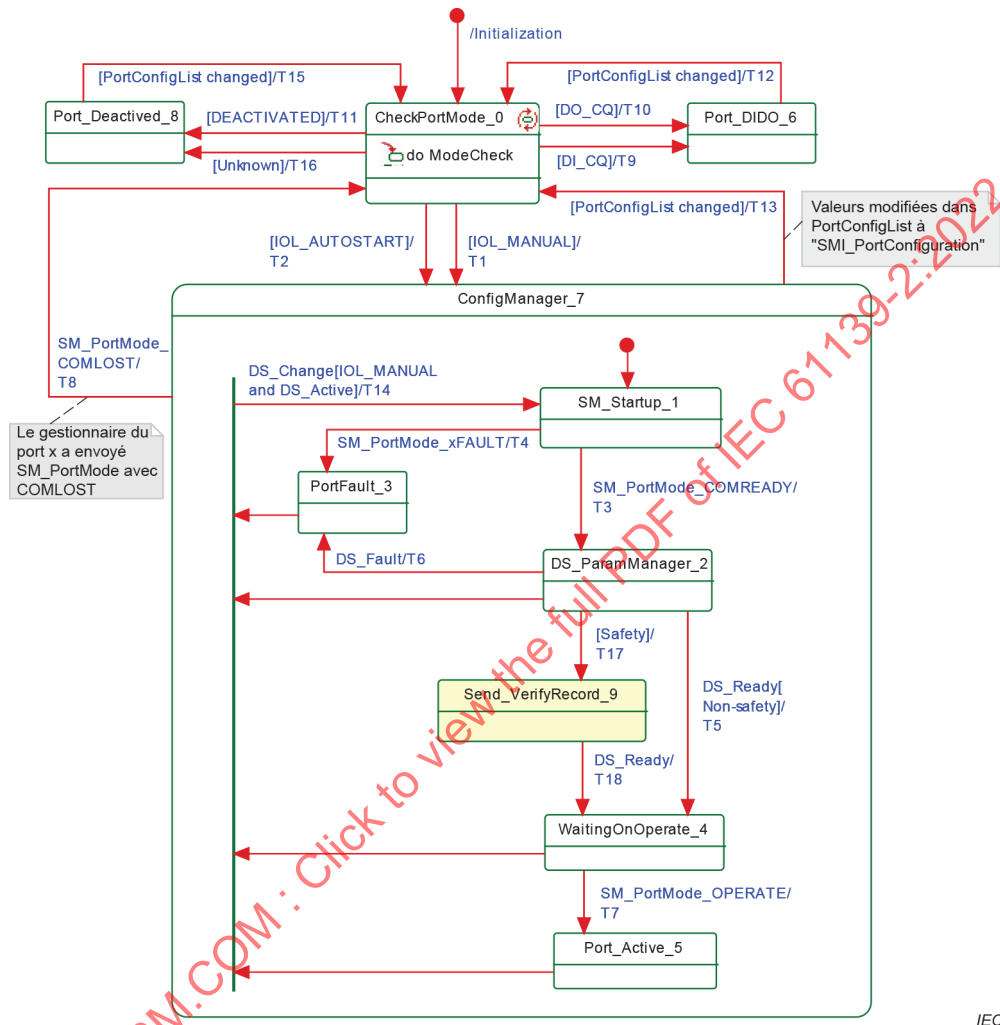


Figure 31 – Extension du gestionnaire de configuration (VerifyRecord)

Un nouvel état "Send_VerifyRecord_9" prend en compte l'exigence qui impose au Maître FS d'envoyer le VerifyRecord, ainsi que l'état, les transitions et les éléments internes supplémentaires du Tableau 12.

Tableau 12 – Tableaux de transition d'états du gestionnaire de configuration

NOM D'ÉTAT		DESCRIPTION DE L'ÉTAT	
CheckPortMode_0 à Port_Deactivated_8		Voir l'IEC 61131-9:—, Tableau 126	
Send_VerifyRecord_9		FS_Master envoie son FSP_VerifyRecord stocké à l'index "masqué" 0x4202 (voir A.2.10)	
TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T1 à T16	*	*	Voir l'IEC 61131-9:—, Tableau 43
T17	2	9	–
T18	9	4	SM_Operate
ÉLÉMENTS INTERNES		TYPE	DÉFINITION
Safety		Protecteur	Appareil FS en mode FS
Non-safety		Protecteur	Appareil FS en mode non FS

9 Extensions de l'Appareil FS

9.1 Architecture de principe et modèles

9.1.1 Architecture de l'Appareil FS

La Figure 32 représente l'architecture de principe de l'Appareil FS. Elle n'inclut pas les mesures de sécurité pour la mise en œuvre, comme la redondance des parties relatives à la sécurité.

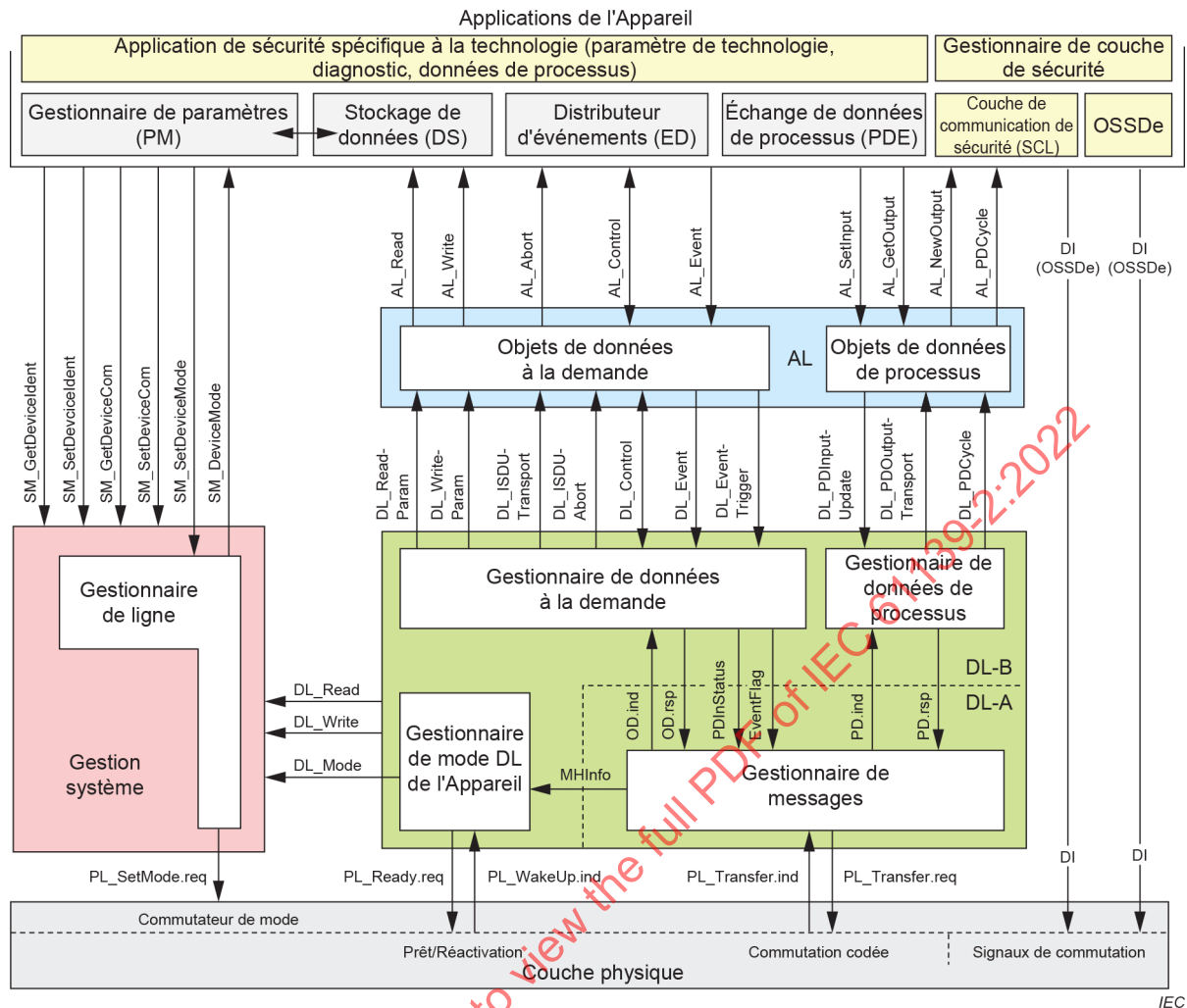


Figure 32 – Architecture de principe de l'Appareil FS

Un Appareil FS comprend tout d'abord l'application de sécurité fonctionnelle spécifique à la technologie. Par exemple, les appareils de sécurité avec "arrêt d'urgence" peuvent être conçus de telle sorte que le fonctionnement OSSDe "classique" ou la communication de sécurité puisse être configuré(e). Le gestionnaire de couche de sécurité est responsable du traitement d'un bit de sécurité par l'intermédiaire du bloc fonctionnel OSSDe ou d'une PDU qui utilise la couche de communication de sécurité (voir Article 11).

Il vérifie l'exactitude des paramètres à chaque démarrage de l'Appareil FS, chaque fois que FSP_VerifyRecord est écrit pendant PREOPERATE. Le moteur de la couche de communication de sécurité (SCL) démarre si tous les paramètres vérifiés sont corrects. Dans le cas contraire, un message d'erreur s'affiche et la SCL reste inactive ou s'arrête.

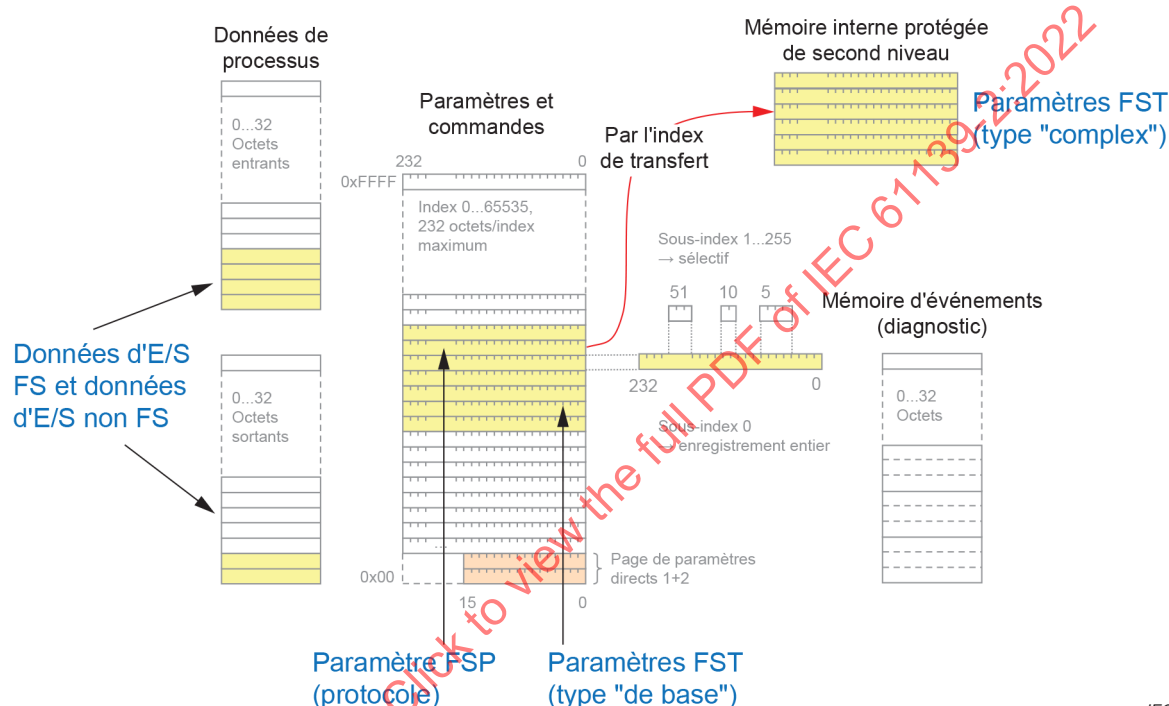
9.1.2 Modèle de l'Appareil FS

Selon l'exigence relative aux données de paramètres et de processus mixtes NSR et SR, le modèle d'Appareil FS a été modifié et adapté.

Cela signifie que le modèle d'index de l'Appareil FS est divisé entre une partie NSR et une partie SR. La Figure 33 représente les zones concernées. L'attribution de la partie SR (paramètre "FSP") est définie dans l'IODD de l'Appareil FS.

Lors de la mise en service, l'affectation des valeurs de paramètre FSP a lieu. Ces valeurs d'instance sont sécurisées par des signatures CRC et transférées sous forme d'enregistrement au Maître et à l'Appareil FS (voir 11.7.4). A chaque démarrage d'un Appareil FS, l'enregistrement de vérification (VerifyRecord) complet stocké dans le Maître FS est transféré d'une autre manière et l'Appareil FS peut vérifier l'intégrité des valeurs de paramètre d'instance stockées en local par comparaison et par les signatures CRC. Cette vérification comprend les paramètres "FST" spécifiques à la technologie, qui ne sont pas transférés à chaque démarrage. L'Appareil FS affiche ses paramètres FSP selon des index prédéfinis (voir Figure 33).

Les paramètres spécifiques à la technologie (FST) peuvent être traités de manière ouverte, dans une certaine mesure, comme des paramètres normalisés non relatifs à la sécurité (voir 11.7.8), ou de manière protégée, en mémoire interne masquée (voir 11.7.9).



IEC

Figure 33 – Modèle de l'Appareil FS

L'espace maximal de partage des données d'E/S FS et des données d'E/S non FS est de 32 octets. L'espace doit être rempli en premier lieu par les données d'E/S FS, puis par les données d'E/S non FS. La limite est variable. En prenant pour hypothèse une queue de protocole de sécurité maximale de 6 octets, l'espace maximal possible pour les données d'E/S FS est de 25 octets.

9.2 Gestionnaire de paramètres (PM)

Aucune extension ni aucune modification du gestionnaire de paramètres n'est exigée.

9.3 Echange des données de processus (PDE)

Selon le paramètre de configuration "Safety", l'échange des données de processus traite ou transmet les données de processus FS (voir 11.4.3, PDU de sécurité) depuis/vers le gestionnaire de couches de sécurité.

9.4 Stockage de données (DS)

9.4.1 Considérations générales et extensions, y compris de sécurité

Les paramètres spécifiques à la technologie (FST) sont sécurisés par une signature CRC particulière (FSP_TechParCRC) incluse dans le jeu de paramètres FSP. Des paramètres d'authenticité supplémentaires sont utilisés en cas de remplacement d'Appareil FS.

Le mécanisme de stockage de données pour les Appareils FS est fondé sur le mécanisme général pour les Appareils non relatifs à la sécurité, comme cela est spécifié dans l'IEC 61131-9. Cette version du stockage de données exige que les bits "0" et "1" de verrouillage de l'accès à l'Appareil (index 0x000C) doivent être toujours déverrouillés (= "0").

Une courte extension du diagramme d'états de stockage de données (DS) de l'Appareil FS est exigée en ce qui concerne le mécanisme de démantèlement (valeurs d'authenticité = "0"). La transition T8 du Tableau 100 de l'IEC 61131-9:— tient compte de cette vérification, comme cela est indiqué dans le Tableau 13.

Tableau 13 – Extension du diagramme d'états de stockage de données (DS)

TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T1 à T7	*	*	Voir l'IEC 61131-9:—, Tableau 100
T8	2	3	Avec "Safety": si les conditions de démantèlement sont incorrectes, renvoie 0x8022 – <i>Service temporairement indisponible – Contrôle de l'Appareil</i> ; si elles sont correctes, verrouille l'accès aux paramètres locaux, définit State_Property = "Upload" ou "Download"
T9 à T11	*	*	Voir l'IEC 61131-9:—, Tableau 100

9.4.2 Niveaux de sauvegarde

Le Tableau 14 répertorie les niveaux de sauvegarde de stockage de données spécifiés dans l'IEC 61131-9. Le 9.4.2 décrit certaines spécialités à prendre en compte pour la sécurité fonctionnelle.

Tableau 14 – Niveaux de sauvegarde du stockage de données

Niveau de sauvegarde	Comportement
Mise en service ("désactivation")	Les modifications des paramètres actifs dans l'Appareil ne sont <i>pas</i> copiées/sauvegardées. Remplacement de l'Appareil <i>sans</i> stockage de données automatique/semi-automatique.
Production ("sauvegarde/restauration")	Les modifications des paramètres actifs dans l'Appareil sont copiées/sauvegardées. Remplacement de l'Appareil <i>avec</i> stockage de données automatique/semi-automatique pris en charge.
Production ("restauration")	Les modifications des paramètres actifs dans l'Appareil ne sont <i>pas</i> copiées/sauvegardées. Si le jeu de paramètres est marqué pour être sauvegardé, les paramètres "figés" sont restaurés par le Maître. Cependant, le remplacement de l'Appareil <i>avec</i> stockage de données automatique/semi-automatique <i>des paramètres figés</i> est pris en charge.

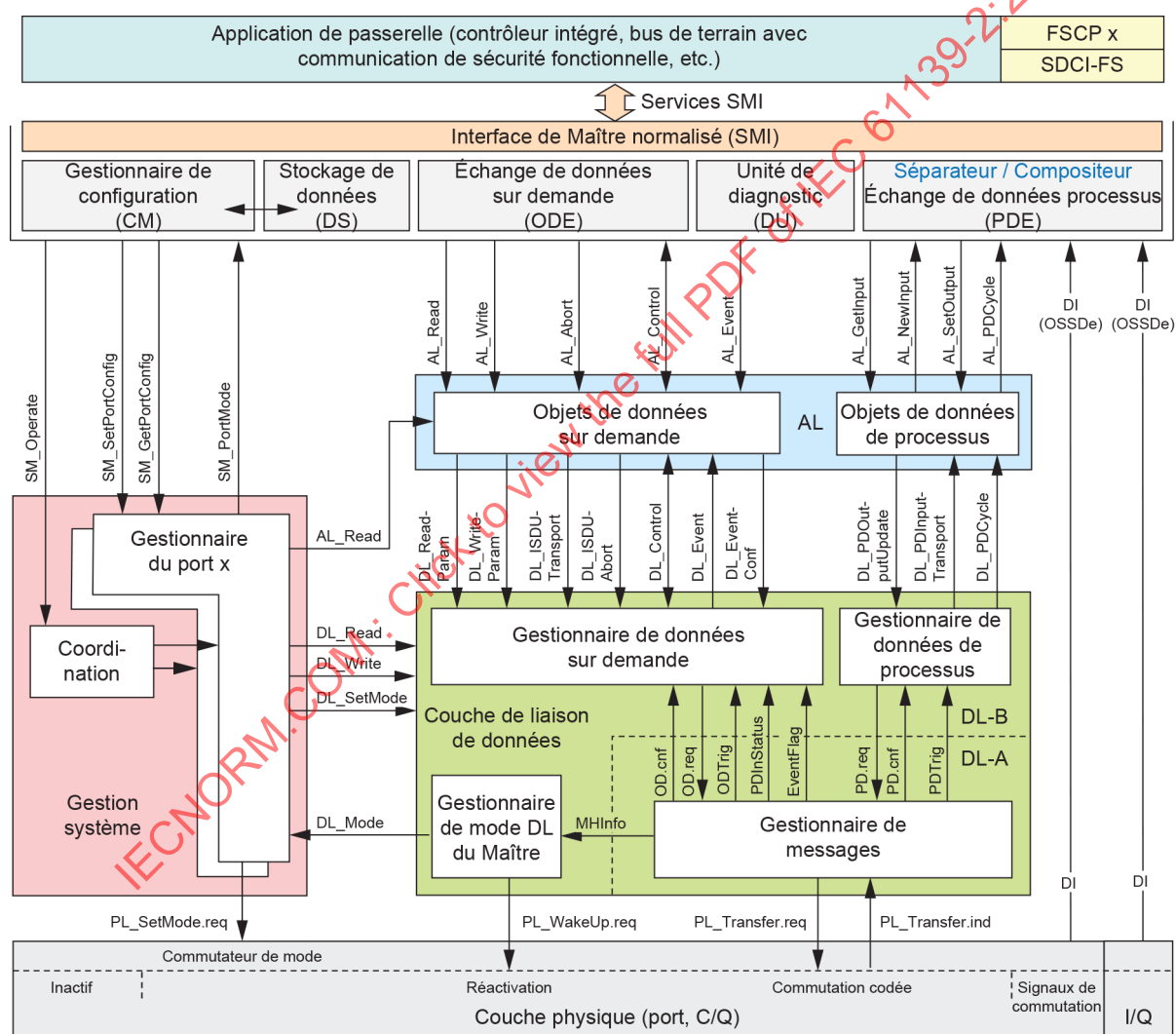
Pour la sécurité fonctionnelle, la mise en service ne peut être effectuée sans vérification et validation des paramètres FSP et FST ainsi que de l'ensemble des fonctions de sécurité conformément aux manuels de sécurité applicables.

La différence par rapport aux "critères de sauvegarde des paramètres" de l'IEC 61131-9 est que, en cas de modification des paramètres locaux (par exemple par le biais de touches d'autoapprentissage ou de panneaux de commande), FSP_TechParCRC doit être affecté à l'aide de l'outil de Maître FS et de l'"outil dédié".

10 Extensions du Maître FS

10.1 Architecture de principe

La Figure 34 représente l'architecture de principe du Maître FS qui offre l'interface de Maître normalisé (SMI, *Standard Master Interface*) étendue conformément à l'IEC 61131-9. Elle permet une séparation stricte du Maître normalisé en tant que "canal noir" et des parties de sécurité fonctionnelles de la SDCI-FS, et un FSCP x qui peut être "encapsulé" dans la couche d'application de la passerelle.



Légende Les parties marquées en jaune sont relatives à la sécurité ; la partie du Maître sous SMI est le "canal noir"

IEC

Figure 34 – Architecture de principe du Maître FS

Un Maître FS contient le Maître normalisé d'origine ("canal noir"), sauf pour l'impulsion "Ready" et son traitement (voir 5.3.3 et 7.2), la seconde DI de la broche 2 (M12) pour le fonctionnement OSSDe. Le gestionnaire de configuration (CM) de l'application du Maître a été modifié pour satisfaire à plus de configurations de port et pour envoyer un enregistrement de vérification à chaque démarrage. L'application d'échange des données de processus (PDE/séparateur/compositeur) est maintenant responsable de la séparation des données de processus mixtes SR et NSR entrantes, respectivement, pour la composition des données de processus SR et NSR sortantes.

10.2 Extensions des services SMI

10.2.1 Vue d'ensemble

Les fondements des services SMI ont été introduits dans l'IEC 61131-9. Dans le présent document, deux services SMI supplémentaires sont spécifiés, comme cela est indiqué dans le Tableau 15 et à la Figure 35: SMI_SPDUIn et SMI_SPDUOut. Ces deux services gèrent les parties de sécurité (SPDU = données de sécurité complètes et code de sécurité) des données de processus mixtes SR et NSR. Le Tableau 15 fournit une vue d'ensemble des services SMI importants utilisés pour les Maîtres FS. Le jeu complet de services peut être extrait de l'IEC 61131-9.

Tableau 15 – Services SMI utilisés pour le Maître FS

Nom de service	Maître	ArgBlockID	Remarque
SMI_MasterIdentification	R	0x0001	–
SMI_FSMasterAccess	R	0x0100	Voir 10.2.2
SMI_PortConfiguration	R	0x8100	Voir IEC 61131-9
SMI_ReadbackPortConfiguration	R	0x8100	Voir IEC 61131-9
SMI_PortStatus	R	0x9100	Voir IEC 61131-9
SMI_DSToParServ	R	0x7000	Stockage de données vers un serveur de paramètres
SMI_ParServToDS	R	0x7000	Stockage de données depuis un serveur de paramètres
SMI_DeviceWrite	R	0x3000	Transport par ISDU
SMI_DeviceRead	R	0x3001	Transport par ISDU
SMI_PortPowerOffOn	R	0x7003	–
SMI_DeviceEvent	I	–	–
SMI_PortEvent	I	–	–
SMI_PDIn	R	0x1001	–
SMI_PDOut	R	0x1002	–
SMI_PDInOut	R	0x1003	–
SMI_SPDUIn	R	0x1101	Voir 10.2.3
SMI_SPDUOut	R	0x1102	Voir 10.2.4
SMI_FSPDInOut	R	0x1103	Voir 10.2.5
SMI_PDInIQ	R	0x1FFE	–
SMI_PDOutIQ	R	0x1FFF	–
Légende			Les services marqués en jaune sont soit des services étendus, soit des services supplémentaires pour SDCI-FS
I Initiateur de service			
R Récepteur (répondeur) du service			

La Figure 35 fournit une vue d'ensemble des services SMI importants utilisés pour le Maître FS, des couches de sécurité de la passerelle et des détails des applications du Maître FS.

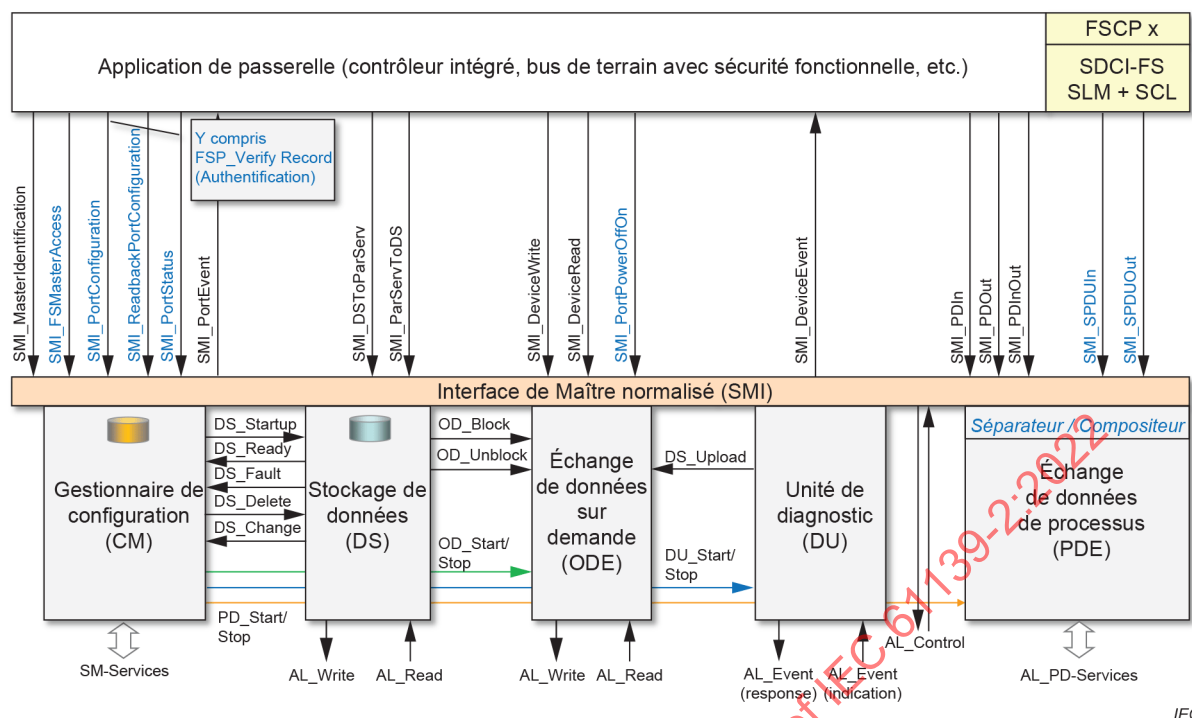


Figure 35 – Extensions des services SMI

SMI_MasterIdentification présente un Maître FS comme MasterType (= "3" selon l'IEC 61131-9). Le service SMI_FSMasterAccess correspondant fournit les codes d'authenticité FSCP du Maître FS en tant qu'appareil FSCP sur un bus de terrain de sécurité. Les services SMI pour la configuration et l'état du port sont uniquement étendus en utilisant différents arguments (ArgBlocks), comme cela est indiqué en 10.3. Au moyen du service SMI "SMI_PortConfiguration", par exemple, les informations concernant l'authenticité, le protocole et la structure de données d'E/S sont transférées au gestionnaire de configuration qui les stocke. Voir 10.4 pour savoir comment ces informations sont utilisées pour s'adapter aux couches de communication de sécurité et pour authentifier le fonctionnement de sécurité. Le service de commande de port "SMI_PortPowerOffOn" est responsable de la mise hors tension et sous tension d'un port particulier.

Deux nouveaux services SMI permettent d'accéder aux parties de sécurité d'une structure de données de processus d'E/S mixtes SR et NSR, comme cela est indiqué en 10.2.3, en 10.2.4 et en 10.5.

10.2.2 SMI_FSMasterAccess

Le rôle utilisateur et le mot de passe correspondant peuvent être fournis aux projets de sécurité du Maître FS et les informations spécifiques MasterType peuvent être récupérées par ce service SMI (voir Figure 35).

Le Tableau 16 décrit la structure du service suivant les règles définies dans l'IEC 61131-9.

Tableau 16 – SMI_FSMasterAccess

Nom de paramètre	.req	.cnf
Argument		
ClientID	M	
PortNumber	M	
ExpArgBlockID (0x0101)	M	
ArgBlockLength	M	
ArgBlock ("FSMasterAccess": 0x0100)	M	
Result (+)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID de l'ArgBlock demandé 0x0100)		M
ArgBlockLength		M
ArgBlock (associé à ExpArgBlockID)		M
Result (-)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID de l'ArgBlock demandé 0x0100)		M
ArgBlockLength		M
ArgBlock (JobError: 0xFFFF)		M

Argument

Les paramètres spécifiques de la demande de service sont transmis dans l'argument.

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID de "FSCPAAuthenticity" (0x0101, voir Tableau 20)

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant

ArgBlock

Ce paramètre contient l'ArgBlock "FSMasterAccess" (0x0100, voir Tableau 19)

Result (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient comme référence l'ID de l'ArgBlock envoyé par la demande (0x0100)

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir Tableau 20)

Result (-):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient comme référence l'ID de l'ArgBlock envoyé par la demande (0x0100)

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF, voir IEC 61131-9)

Valeurs admises par ordre de priorité:

PORT_NUM_INVALID	(numéro de port incorrect)
ARGBLOCK_NOT_SUPPORTED	(ArgBlock inconnu)
ARGBLOCK_LENGTH_INVALID	(longueur d'ArgBlock incorrecte)
ARGBLOCK_INCONSISTENT	(type de contenu ArgBlock incorrect)
SERVICE_TEMP_UNAVAILABLE	(Maître occupé)

10.2.3 SMI_SPDUIn

Ce service permet la lecture cyclique des unités de données de protocole de sécurité (SPDU) à partir d'un FSInBuffer (voir 10.5) et doit être disponible uniquement à la SCL pour un usage interne. Le codage de ce service SMI suit les définitions de l'IEC 61131-9.

L'ArgBlockID attendu est "0x1101". L'ArgBlock est spécifié en 10.3.6.

10.2.4 SMI_SPDUOut

Ce service permet l'écriture cyclique des unités de données de protocole de sécurité (SPDU) sur un FSOutBuffer (voir 10.5) et doit être disponible uniquement à la SCL pour un usage interne. Le codage de ce service SMI suit les définitions de l'IEC 61131-9.

L'ArgBlockID est "0x1102". L'ArgBlock est spécifié en 10.3.7.

10.2.5 SMI_FSPDUInOut

Ce service permet la lecture périodique des données d'entrée d'un FSInBuffer et d'un InBuffer, et des données de sortie d'un FSOutBuffer et d'un OutBuffer (voir 10.5). Le Tableau 17 présente la structure du service.

Tableau 17 – SMI_FSPDInOut

Nom de paramètre	.req	.cnf
Argument		
ClientID	M	
PortNumber	M	
ExpArgBlockID (0x1103)	M	
ArgBlockLength	M	
ArgBlock (VoidBlock: 0xFFFF0)	M	
Result (+)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID de l'ArgBlock demandé 0xFFFF0)		M
ArgBlockLength		M
ArgBlock (associé à ExpArgBlockID)		M
Result (-)		S
ClientID		M
PortNumber		M
RefArgBlockID (ID de l'ArgBlock demandé 0xFFFF0)		M
ArgBlockLength		M
ArgBlock (JobError: 0xFFFF)		M

Argument

Les paramètres spécifiques de la demande de service sont transmis dans l'argument.

ClientID

PortNumber

ExpArgBlockID

Ce paramètre contient l'ArgBlockID de "FSPDInOut" (0x1103, voir Tableau 25)

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant

ArgBlock

Ce paramètre contient l'ArgBlock "VoidBlock" (0xFFFF0)

Result (+):

Ce paramètre de sélection indique que la demande de service a été exécutée avec succès.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient comme référence l'ID de l'ArgBlock envoyé par la demande (0xFFFF0)

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock suivant

ArgBlock

Ce paramètre contient l'ArgBlock associé à l'ExpArgBlockID (voir Tableau 25)

Result (-):

Ce paramètre de sélection indique que la demande de service a échoué.

ClientID

PortNumber

RefArgBlockID

Ce paramètre contient comme référence l'ID de l'ArgBlock envoyé par la demande (0xFFFF0)

ArgBlockLength

Ce paramètre contient la longueur de l'ArgBlock "JobError".

ArgBlock

Ce paramètre contient l'ArgBlock "JobError" (0xFFFF)

Valeurs admises par ordre de priorité:

PORT_NUM_INVALID (numéro de port incorrect)
 ARGBLOCK_NOT_SUPPORTED (ArgBlock inconnu)
 ARGBLOCK_LENGTH_INVALID (longueur d'ArgBlock incorrecte)
 DEVICE_NOT_IN_OPERATE (données de processus non accessibles)

10.3 Extensions d'ArgBlock

10.3.1 Vue d'ensemble

Le Tableau 18 décrit six nouveaux types d'ArgBlocks pour les Maîtres FS: "FSMasterAccess", "FSCPAuthenticity", "FSPortConfigList", "FSPortStatusList", "SPDUIn", "SPDUOut" et "FSPDInOut".

Tableau 18 – Types d'ArgBlocks et ArgBlockIDs

Type d'ArgBlock	ArgBlockID	Remarque
FSMasterAccess	0x0100	Voir 10.3.2
FSCPAuthenticity	0x0101	Voir 10.3.3
PDIn	0x1001	–
PDOOut	0x1002	–
PDInOut	0x1003	–
SPDUIn	0x1101	Voir 10.3.6
SPDUOut	0x1102	Voir 10.3.7
FSPDInOut	0x1103	Voir 10.3.8
DS_Data	0x7000	Objet de stockage de données
PortPowerOffOn	0x7003	–
PortConfigList	0x8000	–
FSPortConfigList	0x8100	Voir 10.3.4
PortStatusList	0x9000	–
FSPortStatusList	0x9100	Voir 10.3.5
Légende  les ArgBlocks marqués en jaune sont les ArgBlocks supplémentaires pour SDCI-FS		

10.3.2 FSMasterAccess

L'ArgBlock "FSMasterAccess" du Tableau 19 indique le mot de passe pour l'accès du Maître FS et le mot de passe correspondant pour réinitialiser l'ensemble du projet du Maître FS, y compris le mot de passe existant.

Tableau 19 – FSMasterAccess

Décalage	Nom d'élément	Définition	Type de données	Valeurs
0	ArgBlockID	ID unique	Unsigned16	0x0100
2	FSMasterPassword	Cet élément contient le mot de passe pour accéder à l'ensemble du projet du Maître FS. La valeur par défaut est 0x0000.	Unsigned32	–
6	FSResetMasterPW	Cet élément contient le mot de passe pour réinitialiser l'ensemble du projet du Maître FS, y compris le FSMasterPassword. La valeur par défaut est 0x0000.	Unsigned32	–
10	FSUserRole	Réservé. Valeur par défaut: 0x00	Unsigned8	–

10.3.3 FSCPAuthenticity

L'ArgBlock "FSCPAuthenticity" du Tableau 20 indique les codes d'authenticité FSCP affectés au Maître FS par le biais de l'outil de développement FSCP de niveau supérieur ou par des commutateurs DIP.

Tableau 20 – FSCPAuthenticity

Décalage	Nom d'élément	Définition	Type de données	Valeurs
0	ArgBlockID	ID unique	Unsigned16	0x0101
2	FSP_Authenticity1	Code A du FSCP, partie 1	Unsigned32	–
6	FSP_Authenticity2	Code A du FSCP, partie 2	Unsigned32	–

10.3.4 FSPortConfigList

Le Tableau 21 décrit l'ArgBlockType "FSPortConfigList" adapté aux Maîtres FS. Il prend en compte des PortModes supplémentaires et est étendu par les longueurs de PDU de sécurité, le FSP_VerifyRecord (voir 10.3.3 et A.2.10) et la description de la structure de données d'E/S FS (voir 11.7.7 et Tableau A.4).

Tableau 21 – FSPortConfigList

Décalage	Nom d'élément	Définition	Type de données	Valeurs
0	ArgBlockID	ID unique	Unsigned16	0x8100
2	PortMode	Cet élément contient le mode de port attendu par le client SMI, par exemple l'application de la passerelle. Tous les modes sont obligatoires, sauf OSSDe. Ils doivent être mappés vers les modes cibles de "SM_SetPortConfig" (voir IEC 61131-9). 0: DEACTIVATED (SM: INACTIVE → Le port est désactivé; les données de processus d'entrée et de sortie sont à "0"; le Maître ne doit pas effectuer d'activités sur ce port) 1: IOL_MANUAL (SM: CFGCOM → Le mode cible est fondé sur la configuration définie par l'utilisateur, y compris la validation des RID, VID et DID) 2: IOL_AUTOSTART^a (SM: AUTOCOM → Le mode cible est sans configuration et sans validation du VID/DID; Le RID obtient la révision la plus élevée prise en charge par le Maître; validation: NO_CHECK) 3: DI_C/Q (broche 4 sur M12)^b (SM: DI → Port en mode d'entrée SIO) 4: DO_C/Q (broche 4 sur M12)^b (SM: DO → Port en mode de sortie SIO) 5 à 48: Réservés pour des versions futures 49: SAFETYCOM (implique un comportement IOL_MANUAL) 50: MIXEDSAFETYCOM (implique un comportement IOL_MANUAL) 51: OSSDE (broche 2 + broche 4 sur M12) (sauf pour les Maîtres FS avec port de classe B; Les valeurs de décalage 14 à 35 sont sans importance; SPDUInLength de décalage 38 = 1 octet; la valeur de décalage 39 est sans importance) 52 à 96: Réservés pour les extensions telles que "Safety" ou "Wireless" 97 à 255: Spécifiques au fabricant	Unsigned8	0 à 0xFF
3	Validation&Backup	Cet élément contient l'InspectionLevel que l'Appareil doit effectuer, ainsi que le comportement de sauvegarde/restauration. 0: Aucune vérification d'Appareil 1: Appareil de type compatible V1.0 2: Appareil de type compatible V1.1 3: Appareil de type compatible V1.1 Sauvegarde + Restauration 4: Appareil de type compatible V1.1 Restauration 5 à 255: Réservés	Unsigned8	0 à 0xFF

Décalage	Nom d'élément	Définition	Type de données	Valeurs
4	Comportement I/Q (Spécifique au fabricant ou au profil)	Cet élément définit le comportement du signal I/Q (broche 2 sur le connecteur M12). Toutes les affectations sont "sans importance" si PortMode est choisi comme étant OSSDE. 0: Désactivé 1: Entrée numérique 2 à 4: Réservés 5: Alimentation 2 (port de classe B) 6 à 255: Réservés	Unsigned8	0 à 0xFF
5	PortCycleTime	Cet élément contient la durée de cycle de port attendue par le client SMI. AFAP est la valeur par défaut. Ces éléments doivent être mappés vers la ConfiguredCycleTime de "SM_SetPortConfig" 0: AFAP ("As Fast As Possible", le plus rapidement possible – SM: FreeRunning → La temporisation du cycle de port n'est pas limitée. Valeur par défaut en mode port IOL_MANUAL) 1 à 255: TIME (SM: pour le codage, voir l'IEC 61131-9:—, Tableau B.3. L'Appareil doit atteindre la durée de cycle de port indiquée. Une erreur doit être créée si cette valeur est inférieure à MinCycleTime de l'Appareil ou en cas d'autres inadéquations)	Unsigned8	0 à 0xFF
6	VendorID	Cet élément contient le VendorID d'une longueur de 2 octets attendu par le client SMI	Unsigned16	1 à FFFF
8	DeviceID	Cet élément contient le DeviceID d'une longueur de 3 octets attendu par le client SMI	Unsigned32	1 à FFFFF
12	FSP_TimeToReady	Cet élément indique le délai entre la mise sous tension et l'impulsion Ready de l'Appareil FS, de sorte que le Maître FS connaît la durée d'attente. Le délai maximal par défaut est de 5 s (voir A.2.11)	Unsigned16	1 à 7FFF (voir Tableau A.1)
14	FSP_MinShutDown Time	Cet élément fournit le délai minimal d'arrêt de l'Appareil FS après la mise hors tension du port et avant un redémarrage	Unsigned16	64 à 03E8 (voir Tableau A.1)
16	FSP_Authenticity1	Code A du FSCP, partie 1 (voir série IEC 61784-3 [1])	Unsigned32	–
20	FSP_Authenticity2	Code A du FSCP, partie 2 (voir série IEC 61784-3 [1])	Unsigned32	–
24	FSP_Port	Numéro de port	Unsigned8	1 à 0xFF
25	FSP_AuthentCRC	Signature CRC pour l'authentification complète	Unsigned16	–
27	FSP_ProtVersion	Version de protocole SDCI-FS	Unsigned8	1 à 0xFF
28	FSP_ProtMode	Mode de protocole SDCI-FS	Unsigned8	–
29	FSP_Watchdog Time	Temps de fonctionnement du chien de garde du Maître FS et de l'Appareil FS	Unsigned16	1 à FFFF
31	FSP_IO_Struct CRC	Signature CRC pour la description des données d'E/S FS	Unsigned16	–
33	FSP_TechParCRC	Signature CRC pour le paramètre de technologie	Unsigned32	–
37	FSP_ProtParCRC	Signature CRC pour le paramètre de protocole	Unsigned16	–

Décalage	Nom d'élément	Définition	Type de données	Valeurs
39	IO_DescVersion	Version de la description de structure générique	Unsigned8	1
40	SPDUInLength	Données OSSDe (1 octet) ou longueur de la SPDU entrante (m); voir 10.5 et Tableau A.4	Unsigned8	1 ou 5 ($32 - n$) octets
41	TotalOfInBits	Jeu de BooleanT d'entrée (bits)	Unsigned8	0 à 0xFF
42	TotalOfInOctets	Jeu de BooleanT d'entrée (octets)	Unsigned8	–
43	TotalOfInInt16	IntegerT(16) d'entrée	Unsigned8	–
44	TotalOfInInt32	IntegerT(32) d'entrée	Unsigned8	–
45	SPDUOutLength	Longueur de la SPDU sortante (o); voir 10.5 et Tableau A.4	Unsigned8	5 à ($32 - p$) octets
46	TotalOfOutBits	Jeu de BooleanT de sortie (bits)	Unsigned8	0 à 0xFF
47	TotalOfOutOctets	Jeu de BooleanT de sortie (octets)	Unsigned8	–
48	TotalOfOutInt16	IntegerT(16) de sortie	Unsigned8	–
49	TotalOfOutInt32	IntegerT(32) de sortie	Unsigned8	–
^a En PortMode "IOL_Autostart", les paramètres VendorID, DeviceID et Validation&Backup sont traités sans importance. ^b En PortMode "DI_C/Q" et "DO_C/Q", tous les paramètres sont sans importance, à l'exception de "InputDataLength" et "Output DataLength".				

10.3.5 FSPortStatusList

Le Tableau 22 décrit l'ArgBlockType "FSPortStatusList" adapté aux Maîtres FS. Celui-ci autorise uniquement le rapport d'état de la partie "canal noir" du Maître FS. Le contenu de "FSPortStatusInfo" doit être issu de "PortMode" selon l'IEC 61131-9.

Tableau 22 – FSPortStatusList

Décalage	Nom d'élément	Définition	Type de données	Valeurs
0	ArgBlockID	ID unique	Unsigned16	0x9100
2	PortStatusInfo	Cet élément contient les informations d'état du port. 0: NO_DEVICE Aucune communication (COMLOST). Toutefois, la configuration de port IOL_MANUAL ou IOL_AUTOSTART a été définie. 1: DEACTIVATED La configuration de port DEACTIVATED est définie. 2: PORT_DIAG Cette valeur est à définir si une DiagEntry indique un diagnostic futur du port au démarrage, à la validation et au stockage de données (erreur groupée). L'Appareil est sur PREOPERATE et DiagEntry contient la cause du diagnostic. 3: Réservé 4: OPERATE Cette valeur est à définir si l'Appareil est sur OPERATE, même en cas d'erreur de l'Appareil. 5: DI_C/Q La configuration de port "DI" est définie. 6: DO_C/Q La configuration de port "DO" est définie. 7: OSSDE (DI sur C/Q et I/Q) 8: SCL_ENABLED Port prêt pour l'échange de données de sécurité) 9 à 253: Réservé 254: PORT_POWER_OFF Arrêt du port actif causé par SM PortPowerOffON 255: NOT_AVAILABLE (Etat du port actuellement indisponible)	Unsigned8 (énum)	0 à 0xFF
3	PortQualityInfo	Cet élément contient les informations d'état des données de processus Bit 0: 0 = VALID 1 = INVALID Bit1: 0 = PDOUTVALID 1 = PDOUTINVALID Bit2 à Bit7: Réservé	Unsigned8	–
4	RevisionID	Cet élément contient des informations sur la révision de protocole SDCl de l'Appareil 0: NOT_DETECTED (Aucune communication sur ce port) <>0: Copié depuis la page des paramètres directs, adresse 4	Unsigned8	0 à 0xFF

Décalage	Nom d'élément	Définition	Type de données	Valeurs
5	Transmission Rate	Cet élément contient des informations sur le taux de transmission effectif du port. 0: NOT_DETECTED (Aucune communication sur ce port) 1: COM1 (taux de transmission 4,8 kbit/s) 2: COM2 (taux de transmission 38,4 kbit/s) 3: COM3 (taux de transmission 230,4 kbit/s) 4 à 255: Réservés pour une utilisation future	Unsigned8	0 à 0xFF
6	MasterCycle Time	Cet élément contient des informations sur la durée de cycle du Maître.	Unsigned8	–
7	InputDataLength	Cet élément contient la longueur des données d'entrée, sous forme de nombre d'octets de l'Appareil fourni par le service PDIn	Unsigned8	0 à 0x20
8	OutputData Length	Cet élément contient la longueur des données de sortie, sous forme de nombre d'octets de l'Appareil accepté par le service PDOut	Unsigned8	0 à 0x20
9	VendorID	Cet élément contient le VendorID d'une longueur de 2 octets attendu par le client SMI	Unsigned16	1 à 0xFFFF
11	DeviceID	Cet élément contient le DeviceID d'une longueur de 3 octets attendu par le client SMI	Unsigned32	1 à 0xFFFF
15	NumberOfDiags	Cet élément contient le nombre x d'entrées de diagnostic (DiagEntry0 à DiagEntry x)	Unsigned8	0 à 0xFF
16	DiagEntry0	Cet élément contient les "EventQualifier" et "EventCode" d'un diagnostic (Événement).	Struct Unsigned8/16	–
19	DiagEntry1	Entrées supplémentaires, jusqu'à x , le cas échéant...	...	–

10.3.6 SPDUIn

Le Tableau 23 décrit la structure de l'ArgBlock "SPDUIn", comme cela est représenté en 10.5.

Tableau 23 – SPDUIn

Décalage	Nom d'élément	Définition	Type de données
0	ArgBlockID	0x1101	Unsigned16
2	SPDUIn0	Unité de données de protocole de sécurité entrante (octet 0)	Unsigned8
3	SPDUIn1	Unité de données de protocole de sécurité entrante (octet 1)	Unsigned8
...			
SPDUInLength + 2	SPDUIn m	Unité de données de protocole de sécurité entrante (octet m)	Unsigned8

10.3.7 SPDUOut

Le Tableau 24 décrit la structure de l'ArgBlock "SPDUOut", comme cela est représenté en 10.5.

Tableau 24 – SPDUIOut

Décalage	Nom d'élément	Définition	Type de données
0	ArgBlockID	0x1102	Unsigned16
2	SPDUOut0	Unité de données de protocole de sécurité sortante (octet 0)	Unsigned8
3	SPDUOut1	Unité de données de protocole de sécurité sortante (octet 1)	Unsigned8
...			
SPDUOutLength + 2	SPDUOuto	Unité de données de protocole de sécurité sortante (octet o)	Unsigned8

10.3.8 FSPDInOut

Le Tableau 25 décrit la structure de l'ArgBlock "FSPDInOut", comme cela est représenté en 10.5.

Tableau 25 – FSPDInOut

Décalage	Nom d'élément	Définition	Type de données	Valeurs
0	ArgBlockID	ID unique	Unsigned16	0x1103
2	PQI	Informations de qualificatif de port	Unsigned8	–
3	OE	Activation de la sortie	Unsigned8	–
4	InputDataLength ($m+n+2$)	Cet élément contient la longueur des données de processus d'entrée de l'Appareil contenues dans les éléments suivants (SR + NSR).	Unsigned8	0 à 0x20
5	SPDUIn0	Unité de données de protocole de sécurité entrante (octet 0)	Unsigned8	0 à 0xFF
6	SPDUIn1	Unité de données de protocole de sécurité entrante (octet 1)	Unsigned8	0 à 0xFF
...				
($m+1$) + 4	SPDUIn m	Unité de données de protocole de sécurité entrante (octet m)	Unsigned8	0 à 0xFF
($m+1$) + 5	PDI0	Données de processus d'entrée (octet 0)	Unsigned8	0 à 0xFF
($m+1$) + 6	PDI1	Données de processus d'entrée (octet 1)	Unsigned8	0 à 0xFF
...				
($m+1$) + ($n+1$) + 4	PDI n	Données de processus d'entrée (octet n)	Unsigned8	0 à 0xFF
($m+1$) + ($n+1$) + 5	OutputDataLength ($o+p+2$)	Cet élément contient la longueur des données de processus de sortie de l'Appareil contenues dans les éléments suivants (SR + NSR).	Unsigned8	0 à 0x20
($m+1$) + ($n+1$) + 6	SPDUOut0	Unité de données de protocole de sécurité sortante (octet 0)	Unsigned8	0 à 0xFF
($m+1$) + ($n+1$) + 7	SPDUOut1	Unité de données de protocole de sécurité sortante (octet 1)	Unsigned8	0 à 0xFF
...				
($m+1$) + ($n+1$) + ($o+1$) + 5	SPDUOuto	Unité de données de protocole de sécurité sortante (octet o)	Unsigned8	0 à 0xFF
($m+1$) + ($n+1$) + ($o+1$) + 6	PDO0	Données de processus de sortie (octet 0)	Unsigned8	0 à 0xFF
($m+1$) + ($n+1$) + ($o+1$) + 7	PDO1	Données de processus de sortie (octet 1)	Unsigned8	0 à 0xFF
...				
($m+1$) + ($n+1$) + ($o+1$) + ($p+1$) + 5	PDO p	Données de processus de sortie (octet p)	Unsigned8	0 à 0xFF

10.4 Gestionnaire de couche de sécurité (SLM)

10.4.1 Objet

Le gestionnaire de couche de sécurité gère la PDU de sécurité, chaque fois que la communication de sécurité a été configurée ou affiche une valeur d'un bit de sécurité, chaque fois que l'OSSDe a été configuré pour un port particulier. Il utilise les services SMI à cette fin, comme cela est spécifié en 10.2.3 et en 10.2.4.

Il contient les paramètres FSP qui comprennent l'enregistrement d'authenticité et l'enregistrement de protocole (voir 11.7.4) ainsi que la description de la structure d'E/S FS (voir Tableau A.1 et E.5.5) pour FS_IO_Data_Mapper.

Après vérification de l'ensemble des paramètres, le moteur de la couche de communication de sécurité (SCL) démarre et PortStatusInfo est défini à "8" (SCL_ENABLED) (voir 10.3.5).

Le SLM ou la SCL, respectivement, utilise l'unité de diagnostic (DU) de la Figure 34 pour propager les PortEventCodes du Tableau B.2.

10.4.2 FS_PortMode

Le Maître FS doit prendre en charge les PortMode suivants des Maîtres NSR normalisés:

- DEACTIVATED
- IOL_MANUAL (base du fonctionnement SCL)
- IOL_AUTOSTART (en général, uniquement en cas d'Appareils connectés totalement inconnus)
- DI_C/Q
- DO_C/Q (les Appareils et les Appareils FS sont protégés contre les courts-circuits)

En outre, le Maître FS doit prendre en charge deux FS_PortModes et un en option:

SAFETYCOM

Ce paramètre permet une communication de sécurité exclusivement sans les données de processus NSR d'un port.

MIXEDSAFETYCOM

Ce paramètre permet la communication de sécurité des données de processus SR et NSR d'un port.

OSSDE

Ce réglage permet le fonctionnement OSSDe d'un port. Ce mode est facultatif.

Tous ces PortModes peuvent être configurés par SMI_PortConfiguration (voir 10.2.1) et par l'ArgBlock "FSPortConfigList" (voir 10.3.4).

10.4.3 Paramètre FSP

10.4.3.1 Cas d'utilisation des paramètres FSP

La Figure 36 représente certains cas d'utilisation associés aux paramètres FSP (voir A.1).

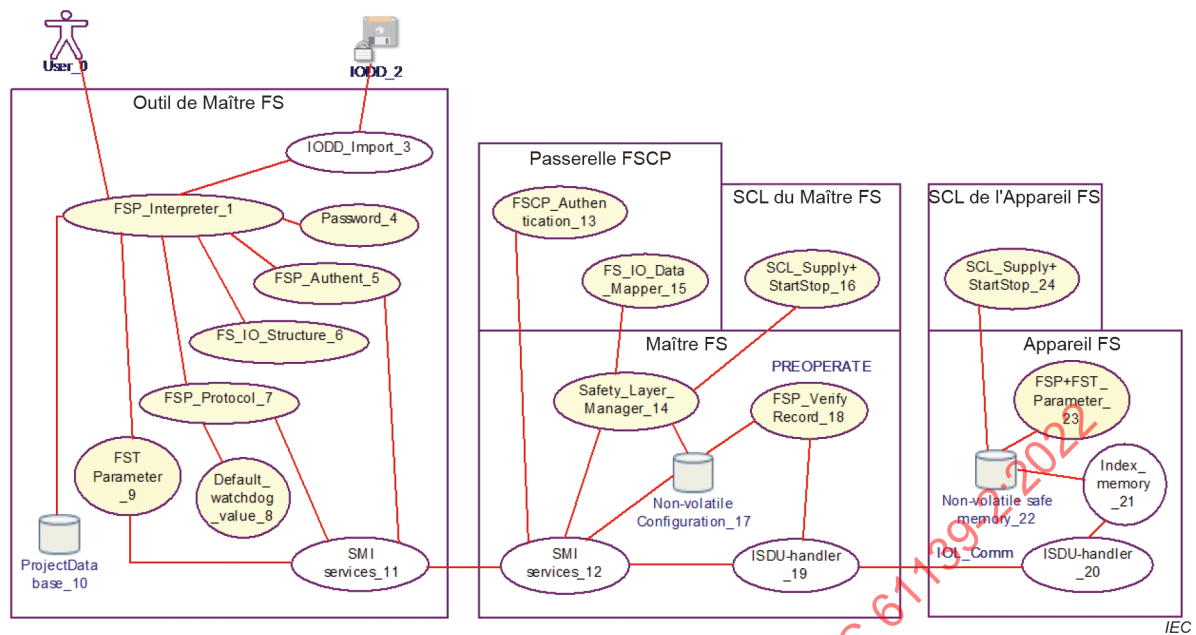


Figure 36 – Cas d'utilisation des paramètres FSP

Le Tableau 26 donne une liste des éléments de la Figure 36 et renvoie aux articles et paragraphes du présent document ou de l'IEC 61131-9.

Tableau 26 – Tableau de référence des cas d'utilisation

N°	Élément	Type	Référence	Remarques
0	Utilisateur	Rôles: - Observateur - Chargé de maintenance - Spécialiste	–	Responsabilité du fabricant de l'outil logiciel
1	FSP_Interpreter	Fonctions GUI	Figure 58, par exemple	–
2	IODD (sécurisée)	Description de l'Appareil	Annex E	–
3	IODD_Import	Activité	Annex E	–
4	Mot de passe	Activité	10.4.3.2	Protection de l'accès
5	FSP_Authent	Activité	11.7.5	–
6	FS_IO_Structure	Description des E/S FS	Article A.1	–
7	FSP_Protocol	Activité	11.7.6	–
8	Default_watchdog_value	Activité	A.2.6	–
9	Paramètre FST	Activité	–	–
10	ProjectDatabase	Outil de Maître FS	–	Propriétaire
11	Interface de Maître normalisé (SMI)	Communication	10.4.3.1	–
12	Interface de Maître normalisé (SMI)	Communication	10.4.3.1	–
13	FSCP_Authentication	Activité	11.7.5	–
14	Safety_Layer_Manager	Activité	10.4	–
15	FS_IO_Data_Mapper	Application de passerelle	12.1	Intégration FSCP
16	SCL_Supply+StartStop	SCL du Maître FS	11.5.2	–

N°	Élément	Type	Référence	Remarques
17	Mémoire non volatile	Maître FS	–	Mise en œuvre
18	FSP_VerifyRecord	Vérification	11.7.4	–
19	Gestionnaire ISDU	DL du Maître FS	IEC 61131-9	SDCI normalisée
20	Gestionnaire ISDU	DL de l'Appareil FS	IEC 61131-9	SDCI normalisée
21	Index_memory	Activité	IEC 61131-9	SDCI normalisée
22	Mémoire non volatile	Appareil FS	–	Mise en œuvre
23	Paramètres FSP+FST	Activités	–	–
24	SCL_Supply+StartStop	SCL de l'Appareil FS	11.5.3	–

Ce qui suit décrit la session de paramétrage type d'un projet dans la ProjectDatabase, avec planification, configuration et paramétrage d'un nouvel Appareil FS pour un port particulier. Après installation de l'IODD et de l'outil dédié associé, l'utilisateur d'un outil de Maître FS ouvre la page des onglets de paramètres (voir représentation à la Figure 58). Après entrée du mot de passe pour les projets de sécurité (voir 10.4.3.2), les paramètres FSP sont activés pour être affichés et les outils dédiés sont activés pour être lancés.

En ligne, l'outil de Maître FS utilise l'interface de Maître normalisé (SMI) avec le Maître FS spécifié dans l'IEC 61131-9. Toute erreur de transmission (voir Tableau 27) peut falsifier les bits de message, aussi chaque enregistrement de paramètre FSP est sécurisé par signature CRC.

Le choix de la technologie d'appel de service SMI relève de l'intégration respective dans un bus de terrain (voir IEC 61131-9).

Les valeurs des *paramètres d'authenticité* sont par défaut "0" dans l'IODD d'un Appareil FS. L'outil de Maître FS acquiert les valeurs d'authenticité FSCP à l'aide du service SMI_FSMasterAccess (voir 10.2.2) et les propose comme valeurs réelles. Pour plus d'informations, voir 10.4.3.3.

L'IODD contient la *description de la structure de données d'E/S* des données de processus de sécurité sous forme d'enregistrement sécurisé par signature CRC (voir A.2.7 et E.5.6). Ces informations sont utilisées pour le mapping vers les données d'E/S FSCP et vérifiées par l'Appareil FS (voir 11.7.7).

La plupart des valeurs des *paramètres de protocole* sont prédéfinies par les valeurs par défaut fournies par le fabricant de l'Appareil FS dans l'IODD, à l'exception de la valeur FSP_TechParCRC, qui a une responsabilité particulière. Une valeur de "0" suivie d'une mise hors tension/sous tension du port indique une mise en service/un essai (voir Annex G). Les conséquences sont les suivantes:

- seule la vérification de FSP_TechParCRC = 0 est exigée pour démarrer la SCL;
- les modifications des paramètres FST prennent effet dès l'acceptation de l'Appareil FS;
- aucune sauvegarde du stockage de données.

Désormais, un système SDCI-FS peut fonctionner en "mode opérationnel surveillé". Cela signifie qu'un personnel est exigé pour surveiller la machine.

L'utilisateur peut désormais saisir et soumettre à l'essai les paramètres spécifiques à la technologie (voir représentation à la Figure 58). Après vérification et validation, l'utilisateur lance l'outil dédié, confirme les affectations de valeur et transfère la signature CRC vers le champ FSP_TechParCRC. La SMI_PortConfiguration correspondante tient compte du FSP_VerifyRecord dans le Maître FS. Avec une valeur FSP_TechParCRC ≠ "0", le système peut être armé:

- stockage de données;
- vérification de l'authenticité, du protocole et des paramètres de technologie, ainsi que de la description des données d'E/S au démarrage.

Après l'affectation des paramètres, les valeurs d'instance des paramètres FSP et FST peuvent être stockées dans la ProjectDatabase.

Une autre mise hors tension/sous tension du port permet à l'Appareil FS d'effectuer des autovérifications de sécurité en amont de la communication et au Maître FS de transmettre le FSP_VerifyRecord à l'Appareil FS. Son gestionnaire de couche de sécurité vérifie tous les paramètres, transmet les paramètres de protocole applicables et lance la SCL. En cas de non-concordance, un Événement correspondant est activé et la SCL ne fonctionne pas.

Le SLM propage la description de la structure d'E/S au FS_IO_DataMapper. Le FSP_VerifyRecord est propagé à la couche de communication de sécurité du Maître FS local (SCL). Il vérifie tous les paramètres, transmet les paramètres de protocole applicables, lance la SCL et définit PortStatusInfo à "8" (SCL_ENABLED). En cas de non-concordance, un Événement correspondant est activé et la SCL ne fonctionne pas.

10.4.3.2 Protection

L'outil de Maître FS crée et gère des projets de sécurité et le Maître FS stocke les paramètres de sécurité dans les données de configuration. Les deux exigent une protection contre les manipulations faciles. L'outil de Maître FS peut utiliser des paramètres de mot de passe dans le service SMI_FSMasterAccess (10.2.2) à cette fin. L'utilisation dépend du concept de sécurité du système de niveau supérieur et du concept d'héritage de la protection par mot de passe. Elle est facultative.

Les outils dédiés peuvent avoir des mécanismes de mot de passe pour leurs Appareils FS indépendants du Maître FS (voir 10.2.2).

10.4.3.3 Enregistrement des paramètres d'authenticité FSP

Les paramètres d'authenticité FSP sont spécifiés en A.1. L'enregistrement d'authenticité FSP inclut le code d'authenticité FSCP, un numéro de port et une signature CRC. L'outil de Maître FS doit toujours mettre à jour la signature CRC lorsque des modifications surviennent, et écrire uniquement des enregistrements complets et cohérents.

Pour les Maîtres FS autonomes, l'entrée de valeurs uniques et univoques par Maître FS est exigée par machine ou par centre de production, s'il existe une possibilité d'erreur de connexion des Appareils FS entre différents Maîtres FS.

10.4.3.4 Enregistrement des paramètres de protocole FSP

Les paramètres de protocole FSP sont spécifiés en A.1. Le fabricant/fournisseur prédéfinit des valeurs et définit des plages dans l'IODD pour la version et le mode de protocole, le mode de port, le chien de garde et TechParCRC.

Le fabricant/fournisseur doit déterminer la valeur prédéfinie du temporisateur de chien de garde en tenant compte du temps de réponse de l'Appareil FS au taux de transmission indiqué (COMx). L'outil de Maître FS peut calculer et suggérer une valeur fondée sur les données de performances du Maître FS utilisé et sur la valeur prédéfinie de l'IODD.

L'outil de Maître FS doit toujours mettre à jour la signature CRC lorsque des modifications surviennent, et écrire uniquement des enregistrements complets et cohérents.

10.4.3.5 Description de la structure de données d'E/S FS

A l'aide de ces informations, le processus de mapping dans la passerelle FSCP peut être contrôlé ou surveillé (voir 11.7.7 et A.2.7).

L'Appareil FS doit vérifier la validité de sa structure PDIn/PDout de sécurité mise en œuvre par la signature FSP_IO_StructCRC fournie par le FSP_VerifyRecord.

10.4.3.6 Enregistrement de vérification

Le Maître FS reprend le FSP_VerifyRecord du service SMI_PortConfiguration (voir 10.3.3 et 11.7.4).

10.5 Echange des données de processus (PDE)

L'échange des données de processus (PDE) de l'application du Maître FS fournit des fonctionnalités supplémentaires appelées "séparateur" et "compositeur".

La Figure 37 représente le mécanisme de séparation entre la partie SPDUIIn et les données d'entrée d'une part et les données SR et NSR complètes d'autre part. La partie SR est stockée dans un FSInBuffer et la partie NSR dans l'InBuffer déjà spécifié dans l'IEC 61131-9.

En cas de PortConfiguration "OSSDE", l'état de signal C/Q est stocké comme "OSSDe1" dans le bit "0" de l'octet "0", et l'état de signal I/Q est stocké comme "OSSDe2" dans le bit "1" de l'octet "0".

Voir l'IEC 61131-9 pour les définitions de "PQI" et "PQ". Un PDIn valide/non valide fait uniquement référence aux données non relatives à la sécurité.

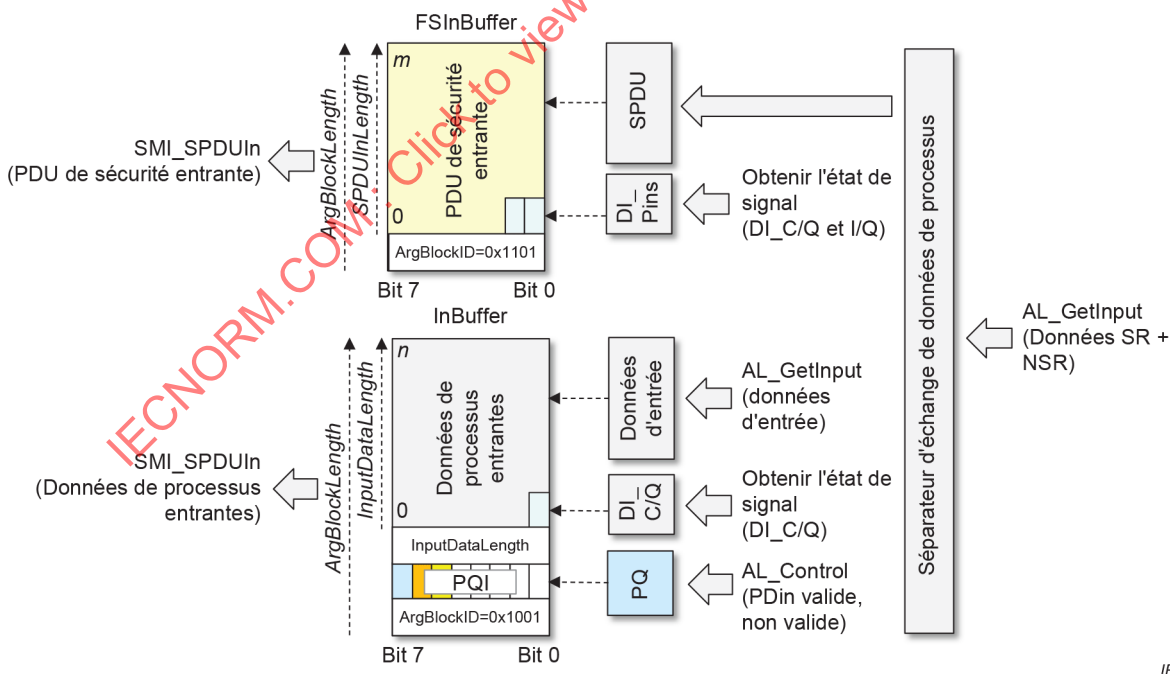


Figure 37 – Séparateur PDE

La Figure 38 représente le mécanisme de composition des données SR et NSR complètes. Celles-ci sont préparées pour le service AL_SetOutput à partir de la partie SPDUIOut du FSOutBuffer et de Process Data Out de l'OutBuffer déjà spécifié dans l'IEC 61131-9.

Voir l'IEC 61131-9 pour les définitions de "OE" et "détection OE".

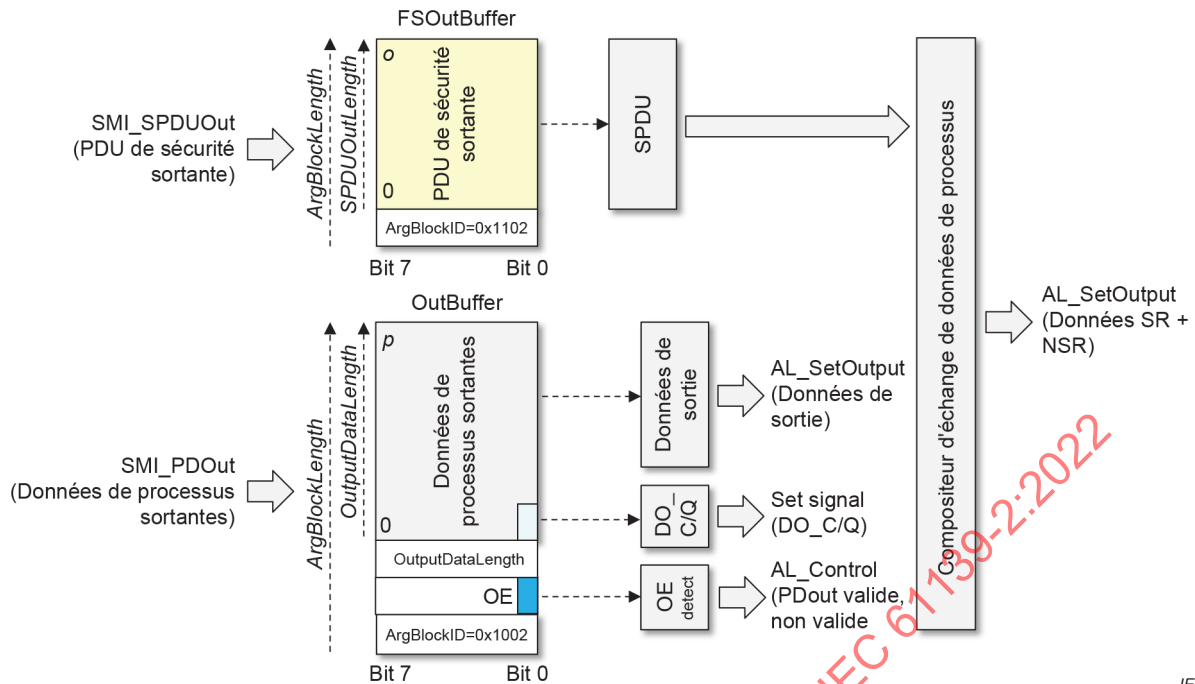


Figure 38 – Compositeur PDE

10.6 Stockage de données (DS)

Le stockage de données aux fins de la sauvegarde des paramètres et du remplacement de l'Appareil/Appareil FS est spécifié dans l'IEC 61131-9. Une brève description et les adaptations pour SDCI-FS peuvent être consultées en 9.4.

11 Couche de communication de sécurité (SCL)

11.1 Exigences fonctionnelles

Les exigences fonctionnelles pour SDCI-FS sont établies en [10]. Le champ d'application principal est la "sécurité des machines". En général, cela signifie que la machine fait l'objet d'un arrêt opérationnel jusqu'à la résolution ou la réparation, et que le redémarrage survient uniquement après l'acquiescement de l'opérateur. L'IEC 62061 et l'ISO 13849 s'appliquent en priorité.

D'autres exigences importantes concernant la compatibilité avec les fonctions de sécurité jusqu'à SIL3/PLe, la passivation spécifique du port et le paramétrage à l'aide d'outils dédiés. Les mesures de sécurité et les taux d'erreurs résiduelles pour l'opportunité, l'authenticité et l'intégrité des données ("TADI", *Timeliness, Authenticity and Data Integrity*) des messages de sécurité (PDU de sécurité) doivent être conformes à l'IEC 61784-3:2021.

11.2 Erreurs de communication et mesures de sécurité

Les fondements de la communication point à point de la SDCI admettent un type de protocole très affiné et une pile de couches de communication de sécurité indépendante du matériel avec une faible empreinte mémoire. Le Tableau 27 indique les erreurs de communication à prendre en compte ainsi que les mesures de sécurité choisies:

- compteur (séquentiel)/compteur inversé;
- temporisateur de chien de garde et messages de réception;
- validation de la connexion à la mise en service, au démarrage et à la réparation; et
- contrôle de redondance cyclique pour l'intégrité des données.

Tableau 27 – Erreurs de communication et mesures de sécurité

Erreur de communication	Mesures de sécurité de protocole			
	Compteur/Compteur inversé	Temporisation avec réception	PortNum + Validation de la connexion ^a	Contrôle de redondance cyclique (CRC)
Corruption	–	–	–	X
Répétition non prévue	X	X	–	–
Séquence incorrecte	X	–	–	–
Perte	X	X	–	–
Retard inacceptable	–	X	–	–
Insertion	X	–	–	–
Déguisement	–	–	X	X
Adressage	–	–	X	–
Bouclage des messages	X	–	X	–
^a La validation de la connexion comprend l'authenticité FSCP (voir A.2.1) et le numéro de port du Maître FS.				

Il est admis par hypothèse qu'il n'existe pas d'éléments de stockage dans le chemin de communication SDCI entre le Maître FS et l'Appareil FS. Ainsi, un compteur à trois bits suffit comme mesure de sécurité. La valeur 0b000 de ce compteur du côté du Maître FS indique une position de départ ou de réinitialisation du compteur. En mode cyclique, il compte jusqu'à 0b111 et revient à 0b001.

Le concept de messages "envoyer" et "recevoir" de la SDCI permet un concept de mesure de sécurité simple pour le temporisateur du chien de garde et la réception des messages, qui correspond au principe de "mise hors tension pour déclenchement".

Il est admis par hypothèse qu'un Maître FS est le propriétaire d'un ID de connexion de sécurité fonctionnelle du système de communication FSCP de niveau supérieur (authenticité FSCP) similaire à un module FS-DI dans un système d'E/S déporté. Il est exigé que le client effectue une procédure de validation, chaque fois qu'une modification survient sur les appareils de sécurité connectés. La SDCI-FS repose sur un tel concept. En outre, en raison du mécanisme de "stockage de données" normalisé de la SDCI et de la nature de sécurité fonctionnelle du Maître FS, il est possible de fournir un mécanisme plus pratique.

Une signature CRC est utilisée pour vérifier l'intégrité des données des PDU de sécurité transmises. Deux options peuvent être configurées. Une signature CRC 16 bits pour les données d'E/S de sécurité jusqu'à 3 octets ou une signature CRC 32 bits pour les données d'E/S de sécurité jusqu'à 25 octets peut être choisie.

11.3 Services SCL

11.3.1 Positionnement des couches de communication de sécurité (SCL)

La Figure 39 indique le positionnement de la couche de communication de sécurité (SCL) SDCI-FS.

Pour chaque port avec un Appareil FS connecté, une instance de la SCL est exigée. Les SCL échangent des PDU de sécurité qui comprennent des données de processus de sortie (PDout) avec code de sécurité vers l'Appareil FS et des données de processus d'entrée (Pdin) avec code de sécurité depuis l'Appareil FS. Les SCL utilisent la communication SDCI normalisée comme "canal noir".

Une disponibilité suffisante, par exemple par le biais d'installations correctes, des alimentations à faible bruit et de faibles interférences sont des conditions préalables pour ce "canal noir", afin

d'éviter les déclenchements dits de nuisance/parasites. Ces déclenchements provoquent des arrêts de production et peuvent, par la suite, amener la direction à retirer l'équipement de sécurité.

Le présent document ne spécifie pas les mesures de sécurité relatives à la mise en œuvre, telles que les microcontrôleurs redondants, les essais de RAM, etc. Il incombe au fabricant/fournisseur de prendre les mesures appropriées contre les défaillances ou les erreurs de composants selon l'IEC 61508 (toutes les parties).

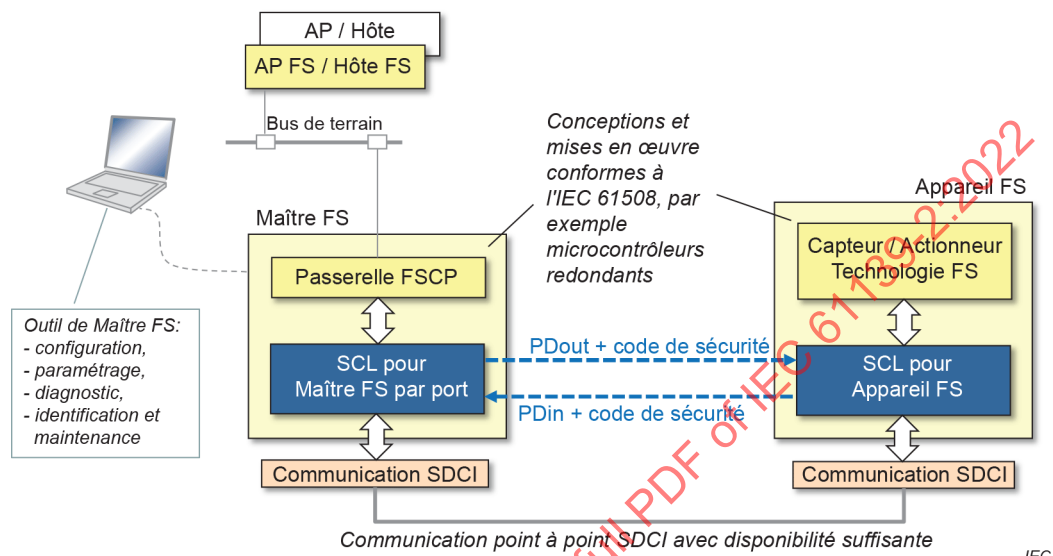
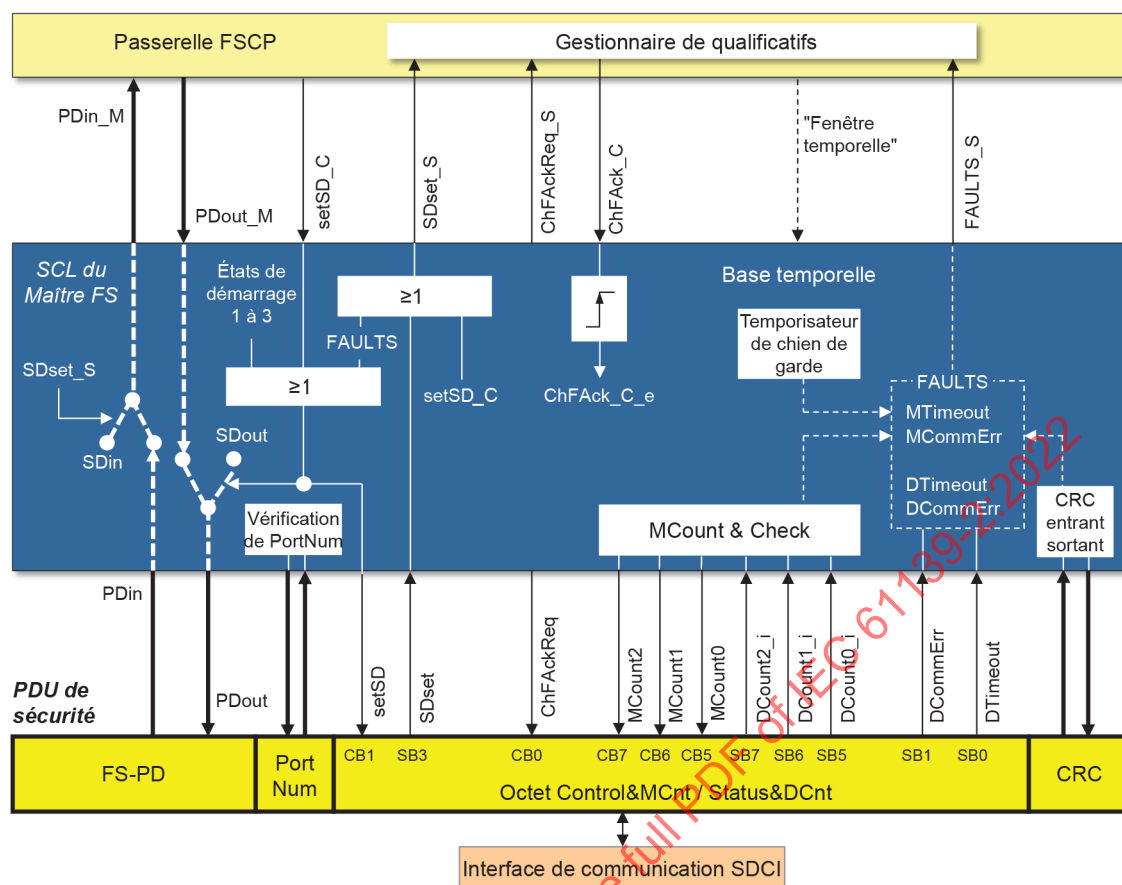


Figure 39 – Positionnement de la couche de communication de sécurité SDCI-FS (SCL)

11.3.2 Services SCL du Maître FS

Les applications SDCI-FS comprennent (entre autres) des connexions aux systèmes de bus de terrain FSCP de niveau supérieur. En général, les FSCP fournissent également des codes de sécurité et des services de contrôle/surveillance (signaux).

La Figure 40 représente les signaux de la couche de communication de sécurité du Maître FS (services), qui sont représentés par des flèches dans la partie supérieure de la figure. Pour chaque FSCP auquel se connecter, un mapping ou une émulation des services SCL correspondants est exigé(e).



IEC

Figure 40 – Services de la couche de communication de sécurité du Maître FS

Un nom de service porte soit une extension "_C" (Control, contrôle) s'il contrôle les activités de communication de sécurité, soit une extension "_S" (Status, état) s'il rend compte des activités.

Certains noms de service correspondent aux noms de signal de l'octet Control ou de l'octet Status (voir la partie inférieure de la figure et 11.4.5). Cela signifie qu'ils sont corrélés, mais qu'il existe une logique de contrôle de la SCL entre eux. Cette logique de contrôle est discrète dans le temps et discontinue, même si elle est décrite comme la boîte logique OR ("≥"). Les diagrammes d'états et les tableaux de transition d'états de la SCL sont définitifs (voir 11.5.2).

Les services suivants du Tableau 28 doivent être disponibles pour la passerelle FSCP ou pour un programmeur d'un système de Maître FS.

Tableau 28 – Services SCL du Maître FS

Service/Signal	Définition
PDin_M, PDout_M	Ces services transportent les valeurs de données de processus réelles, à la fois SDin (tous les bits "0") et SDout (tous les bits "0") en cas d'état de sécurité, ou les valeurs de processus réelles depuis ou vers l'Appareil FS.
SDin, SDout	Ces services transportent des valeurs de données de processus toutes nulles.
setSD_C	En cas d'urgence, les programmes de contrôle de sécurité définissent généralement les données de processus de sortie (PDout_M) d'un actionneur sur "0". Toutefois, dans certains cas (ventilateurs de brûleur, par exemple), l'arrêt peut ne pas être un état de sécurité. S'il est défini sur "1", ce service constitue une information supplémentaire qui permet à l'Appareil FS d'établir un état de sécurité, quelles que soient les valeurs des données de processus. Indépendant de PDout_M, ce service force la SCL à envoyer les valeurs SDout à l'Appareil FS et à envoyer SDin à la passerelle FSCP (PDin_M) par l'intermédiaire de SDset_S.
SDset_S	S'il est défini sur "1", ce service force le gestionnaire de qualificatifs à définir le bit qualificatif pour les données de processus de l'Appareil FS connecté (voir 11.11.4). De plus, il force la SCL à envoyer SDin à la passerelle FSCP (PDin_M).
ChFAckReq_S	La SCL du Maître FS définit ce service sur "1" en cas de disparition de FAULTS ou d'expirations du Maître FS. Il doit être propagé par FSCP et indiqué à l'opérateur.
ChFAck_C	Après une vérification et/ou une réparation, il est demandé à l'opérateur d'acquiescer un service "ChFAckReq_S" avec un "1". Il s'agit d'une condition préalable pour que la SCL reprenne un fonctionnement normal après 3 cycles de transmission avec les valeurs SDin et SDout. L'opérateur doit débloquer le bouton d'acquiescement enclenché pour permettre d'autres acquiescements. Voir "Eléments internes" dans le Tableau 38.
FAULTS_S	Toute erreur de communication (non-concordance de compteur ou erreur de signature CRC) et/ou toute expiration forcent le gestionnaire de qualificatifs à définir le bit qualificatif pour les données de processus de l'Appareil FS connecté (voir 11.11.4).

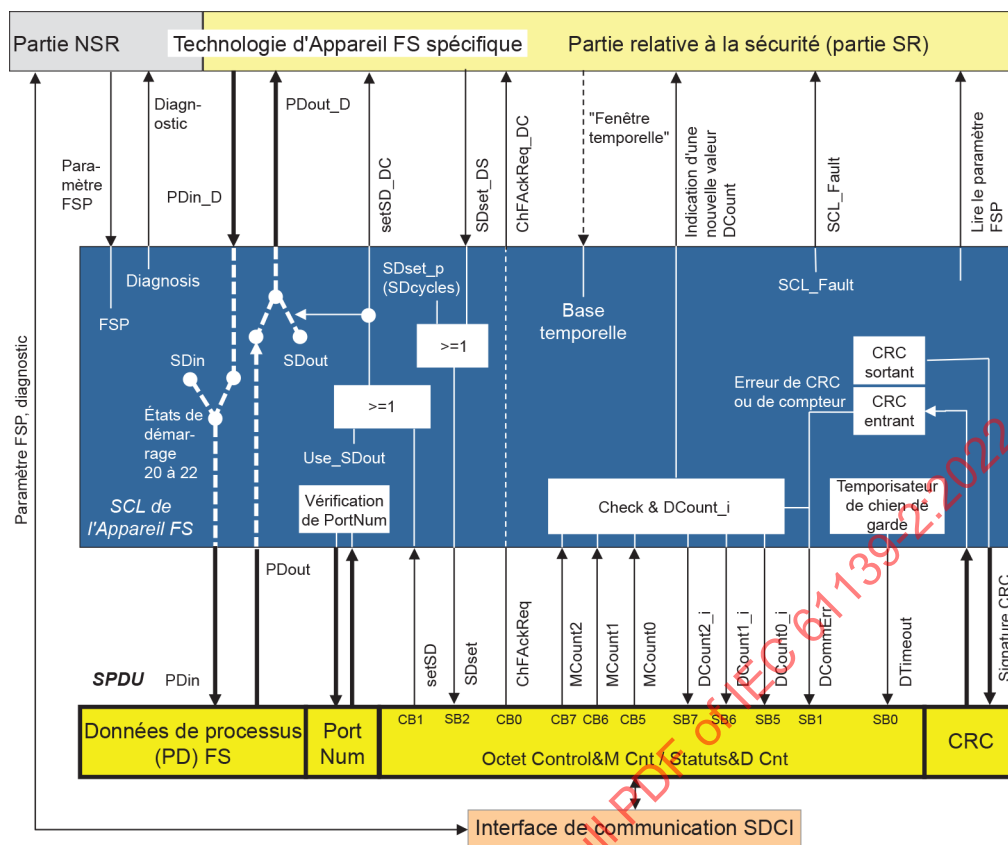
La partie inférieure de la figure représente une PDU de sécurité d'entrée et de sortie combinée, spécifiée en 11.4.3 et en 11.4.5.

11.3.3 Services SCL de l'Appareil FS

La Figure 41 représente les services de la couche de communication de sécurité de l'Appareil FS, qui sont représentés par des flèches dans la partie supérieure de la figure.

Un nom de service porte soit une extension "_DC" (Device Control, contrôle de l'Appareil) s'il contrôle la technologie de l'Appareil FS, soit une extension "_DS" (Device Status, état de l'Appareil) s'il rend compte de son état.

Certains noms de service correspondent aux noms de signal de l'octet Control ou de l'octet Status (voir la partie inférieure de la figure et 11.4.5). Cela signifie qu'ils sont corrélés, mais qu'il existe une logique de contrôle de la SCL entre eux. Cette logique de contrôle est discrète dans le temps et discontinue, même si elle est décrite comme la boîte logique OR ("≥"). Les diagrammes d'états et les tableaux de transition d'états de la SCL sont définitifs (voir 11.5.3).



IEC

Figure 41 – Services de la couche de communication de sécurité de l'Appareil FS

Les services suivants du Tableau 29 doivent être disponibles pour la partie relative à la sécurité de la technologie de l'Appareil FS. Certains services ne sont pas relatifs à la sécurité et doivent être disponibles pour la partie non relative à la sécurité de l'Appareil FS.

Tableau 29 – Services SCL de l'Appareil FS

Service/Signal	Définition
PDin_D, PDout_D	Ces services transportent les valeurs de données de processus réelles, les valeurs de processus réelles depuis l'Appareil FS et SDout (tous les bits "0") en cas d'état de sécurité, ou les valeurs de processus réelles vers l'Appareil FS.
SDin, SDout	Ces services transportent des valeurs de données de processus toutes nulles. Le signal Use_SD indique l'utilisation de données de processus toutes nulles.
setSD_DC	En cas d'urgence, les programmes de contrôle de sécurité définissent généralement les données de processus de sortie (PDout) d'un actionneur sur "0". Toutefois, dans certains cas (ventilateurs de brûleur, par exemple), l'arrêt peut ne pas être un état de sécurité. S'il est défini sur "1", ce service constitue une information supplémentaire qui permet à l'Appareil FS d'établir un état de sécurité, quelles que soient les valeurs des données de processus. Indépendant de PDout, ce service force la SCL à envoyer les valeurs SDout à l'Appareil FS.
SDset_DS	S'il est défini sur "1", ce service indique que l'Appareil FS réagit à un setSD_DC = "1" lorsque l'état de sécurité est établi, ou qu'il a été forcé d'établir l'état de sécurité en raison d'une erreur ou d'une défaillance, et qu'il fournit des valeurs de données de processus d'entrée "0" (PDin_D).
ChFAckReq_DC	S'il est défini sur "1", ce service indique un acquittement de l'opérateur en attente. Ce signal n'est pas relatif à la sécurité et peut être utilisé pour contrôler un indicateur, par exemple une LED (diode électroluminescente).
Fenêtre temporelle	La SCL peut être conçue comme totalement indépendante du matériel si un appel de service périodique contrôle une base temporelle dans la SCL.
Indication de la nouvelle valeur DCount	Les demandes courtes des Appareils FS peuvent ne pas déclencher une fonction de sécurité en raison de leur chaîne de cycles de communication indépendants à travers le réseau. Par conséquent, une demande doit durer au moins deux cycles SCL. Ce service fournit les informations nécessaires pour mettre en œuvre l'extension de la demande, si nécessaire.
SCL_Fault	Ce service indique les anomalies (erreurs) du logiciel SCL.
Read_FSP_Parameter	Ce service permet la technologie de l'Appareil FS pour lire le paramètre FSP (protocole) actuel.
Services non relatifs à la sécurité:	
FSP_Parameter	A chaque démarrage pendant PREOPERATE, le Maître FS transmet l'enregistrement des paramètres FSP (bloc) à l'Appareil FS. Ces paramètres sont propagés à la SCL en utilisant ce service.
Diagnostic	Les informations de diagnostic SCL peuvent être propagées au système d'Événements SDCl en utilisant ce service.

La partie inférieure de la Figure 41 représente une PDU de sécurité d'entrée et de sortie combinée, spécifiée en 11.4.3 et en 11.4.5.

11.4 Protocole SCL

11.4.1 Phases de protocole à prendre en compte

La Figure 42 représente les phases de protocole de principe à prendre en compte pour la conception selon l'IEC 61784-3:2021.

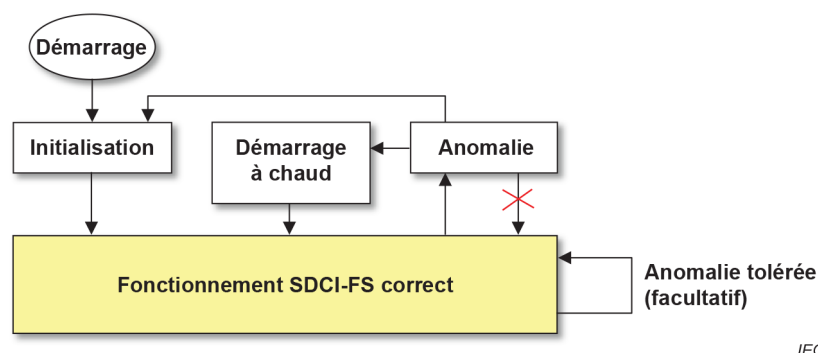


Figure 42 – Phases de protocole à prendre en compte

Les phases de protocole de principe et les exigences correspondantes sont répertoriées dans le Tableau 30.

Tableau 30 – Phases de protocole à prendre en compte

Phase	Activités	Exigences
Initialisation	Etablissement de la communication, transfert du paramètre FSP à l'Appareil FS, cycles SD	– L'actionneur doit être mis hors tension – Les valeurs SDout doivent être utilisées pendant les trois premiers cycles de communication SCL
Configuration ou modification	Mise en service, sauvegarde des paramètres FST	– Tant que le FSP_TechParCRC est défini sur "0", l'échange de données cyclique de valeurs PD est activé
Utilisation	Echange de données de processus, mise hors tension de l'Appareil FS	– La détection des sous-tensions et la définition des valeurs SD relève de la technologie de l'Appareil FS
Redémarrage après transition à la suite d'une anomalie	Temporisation, acquittement de l'opérateur	– L'acquittement de l'opérateur est exigé avant un redémarrage – MCounter est réinitialisé (resynchronisation) – Les valeurs SDout doivent être utilisées pendant les trois premiers cycles de communication SCL
Démarrage à chaud après transition à la suite d'une anomalie	Erreur CRC ou erreur compteur, acquittement de l'opérateur	– L'acquittement de l'opérateur est exigé avant un redémarrage – La communication SCL n'est pas réinitialisée – Les valeurs SDout doivent être utilisées pendant les trois premiers cycles de communication SCL
Arrêt	Rebondissement de contact, creux/variations de tension CEM	– La détection des sous-tensions et la définition des valeurs SD relève de la technologie de l'Appareil FS

11.4.2 Anomalies de l'Appareil FS

Le protocole SCL traite les anomalies qui surviennent lors de la transmission de PDU de sécurité, telles que les erreurs CRC ou les expirations. Il incombe au concepteur de l'Appareil FS de gérer les anomalies de l'Appareil FS et de s'assurer que les mesures de sécurité fonctionnelle nécessaires sont mises en place, par exemple la définition des données de processus de sécurité et le service SDset_DS.

11.4.3 PDU de sécurité (SPDU)

La Figure 43 représente la structure des SPDU du Maître FS et de l'Appareil FS, ainsi que les données d'entrée et de sortie normalisées. La conception suit le concept de transmission explicite des mesures de sécurité pour l'opportunité et l'authenticité, conformément à l'IEC 61784-3:2021, par opposition à la transmission implicite par inclusion dans le calcul global de la signature CRC.

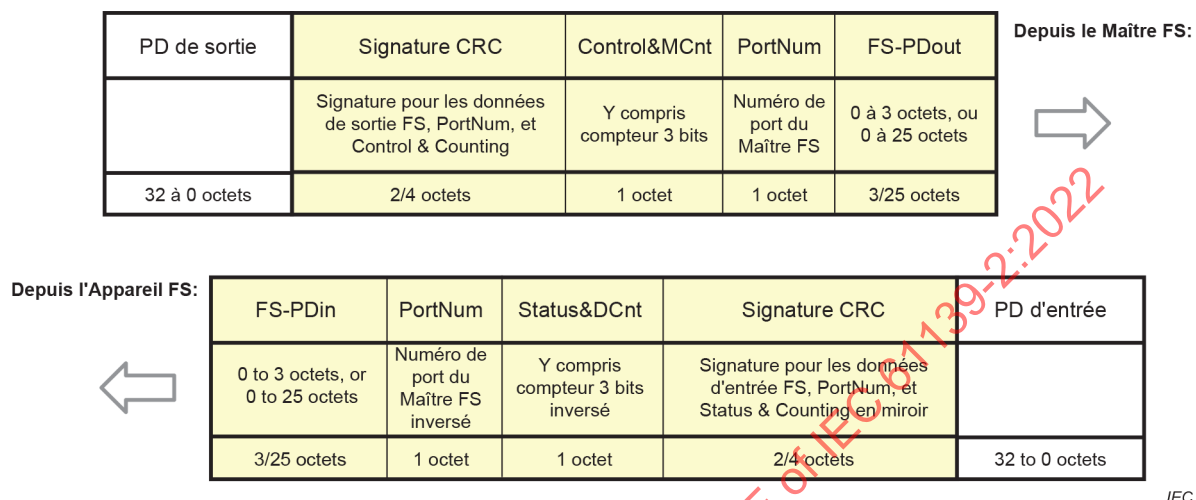


Figure 43 – PDU de sécurité du Maître FS et de l'Appareil FS

La mesure d'opportunité est représentée par un compteur 3 bits dans les octets de gestion de protocole (voir 11.4.6).

En matière d'authenticité, seul le numéro de port du Maître FS est inclus dans la vérification cyclique, en raison de l'utilisation demandée des mises en œuvre SDCI non modifiées ("canal noir"). Cependant, une vérification complète de l'authenticité est effectuée pendant la mise en service et au démarrage.

La conception constitue une optimisation du principe original de "mise hors tension pour déclenchement". En cas d'expiration, d'erreur CRC, d'erreur de compteur ou d'erreur PortNum, le bit qualificatif associé est défini. Il n'est libéré qu'après un acquittement explicite de l'opérateur du côté du Maître FS. Après une erreur CRC, un démarrage à chaud est possible.

11.4.4 Données d'entrée FS et de sortie FS

La taille maximale possible des données d'entrée FS et de sortie FS est de 0 à 25 octets, selon le volume de données SDCI normalisées exigé. Voir 11.4.7 pour les questions d'optimisation et de compromis. Les types de données possibles sont répertoriés dans le Tableau 34.

NOTE Actuellement, le code de sécurité se compose de 4 ou 6 octets uniquement; en théorie, 26 octets peuvent être disponibles.

11.4.5 Numéro de port

Un octet porte le numéro de port du Maître FS ou la valeur de FSP_Port, respectivement (voir A.2.2). L'Appareil FS renvoie la valeur inversée du numéro de port.

11.4.6 Etat et contrôle

Un octet est utilisé dans les deux sens de transmission pour le flux de protocole de la SDCI-FS. Le Tableau 31 décrit les signaux pour contrôler la couche de protocole d'un Appareil FS et une valeur de compteur pour vérifier l'opportunité, ainsi qu'un temporisateur de chien de garde local ajusté par le paramètre "FSP_Watchdog" (voir A.2.6).

Tableau 31 – Contrôle et comptage (Control&MCnt)

CB7	CB6	CB5	CB4	CB3	CB2	CB1	CB0
Compteur séquentiel; bit 2	Compteur séquentiel; bit 1	Compteur séquentiel; bit 0	Réservé ("0")	Réservé ("0")	Réservé ("0")	Activation de l'état de sécurité	Demande d'acquittement d'anomalie de canal (indication)
MCount2	MCount1	MCount0	–	–	–	SetSD	ChFAckReq

Le Tableau 32 décrit la réaction de la couche de protocole d'un Appareil FS et la valeur de compteur inversée pour vérifier l'opportunité. Les valeurs de compteur sont inversées pour empêcher les erreurs de bouclage non détectées.

Tableau 32 – Etat et comptage en miroir (Status&DCnt)

SB7	SB6	SB5	SB4	SB3	SB2	SB1	SB0
Compteur séquentiel; bit 2; inversé	Compteur séquentiel; bit 1; inversé	Compteur séquentiel; bit 0; inversé	Réservé ("0")	Réservé ("0")	Etat de sécurité activé	Erreur de communication: CRC ou compteur/port incorrect	Anomalie de communication: Expiration
DCount2_i	DCount1_i	DCount0_i	–	–	SDset	DCommErr	DTimeout

Le Tableau 33 indique les valeurs de MCount et DCount_i pendant l'utilisation du protocole.

Tableau 33 – Valeurs MCount et DCount_i

Phase	MCount		DCount_i	
	Déc.	Bin.	Déc.	Bin.
Initiale ou après expiration	0	000	7	111
Cyclique	1	001	6	110
	2	010	5	101
	3	011	4	100
	4	100	3	011
	5	101	2	010
	6	110	1	001
	7	111	0	000

11.4.7 Signature CRC

Pour la conception du mécanisme CRC et le calcul de la probabilité/du taux d'erreurs résiduelles, plusieurs paramètres et hypothèses sont exigés:

- absence de transmission multipoint, multicanal ou chiffrée dans la SDCI. Le modèle de canal symétrique binaire (BSC, *Binary Symmetric Channel*) peut donc être appliqué;
- transmission explicite de mesures de sécurité par opposition à une transmission implicite. Dans ce cas, les formules sont disponibles dans l'IEC 61784-3:2021;
- il est admis par hypothèse que le taux d'échantillonnage maximal des PDU de sécurité est de 1 000 PDU de sécurité échantillonnées par seconde;

- les périodes de surveillance des erreurs dans les PDU de sécurité sont indiquées dans le Tableau 41. Toute erreur CRC détectée dans la couche de communication de sécurité doit déclencher la fonction de sécurité correspondante (état de sécurité). Pendant la période de surveillance, seul un déclenchement de nuisance est autorisé. Une maintenance est exigée;
- les polynômes générateurs utilisés doivent être démontrés comme appropriés dans la plage SPDU.
- La valeur de départ à utiliser pour le calcul de la signature CRC est "1" (voir D.3.6).
- Si le résultat du calcul de la signature CRC conduit à un "0", un "1" doit être envoyé et évalué en conséquence du côté récepteur.
- La probabilité d'erreurs sur les éléments binaires admise par hypothèse pour les calculs est de 10^{-2} .

La Figure 44 représente la règle du taux de 1 % de l'IEC 61784-3:2021. Pour les SDCI-FS, cela signifie que le taux d'erreurs résiduelles d'une connexion logique SDCI-FS ne doit pas dépasser 1 % de la fréquence moyenne de défaillance dangereuse (PFH) de la fonction de sécurité avec le SIL le plus élevé pour lequel la communication de sécurité est conçue, à savoir SIL3. Cette valeur est de $10^{-9}/h$.

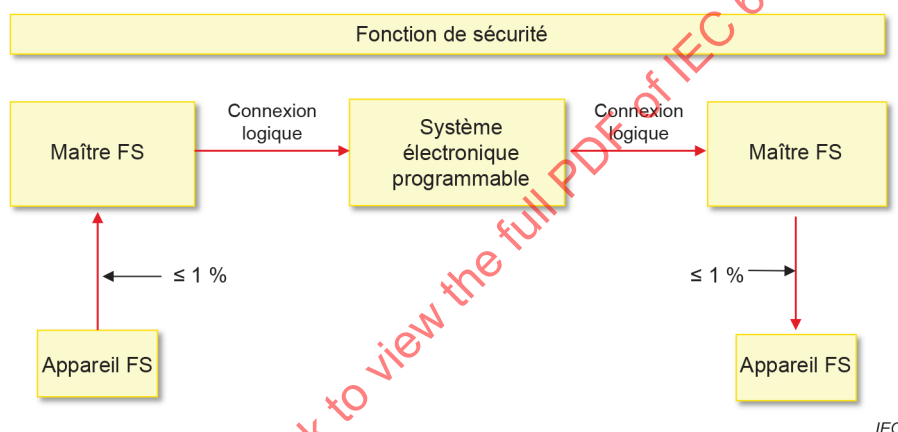


Figure 44 – Règle du taux de 1 % de l'IEC 61784-3:2021

Les calculs effectués dans les conditions ci-dessus ont révélé les possibilités suivantes (voir Annex D):

- pour un polynôme CRC-16 approprié ($0x4EAB$), 3 octets de données de processus (longueur de PDU de sécurité = 7 octets);
- pour un polynôme CRC-32 approprié ($0xF4ACFB13$), 25 octets de données de processus (longueur de PDU de sécurité = 31 octets).

Ainsi, deux variantes sont prises en charge: CRC-16 avec jusqu'à 3 octets de données d'E/S de sécurité et CRC-32 avec jusqu'à 25 octets de données d'E/S de sécurité.

11.4.8 Considérations de sécurité relatives à l'opportunité, l'authenticité et l'intégrité des données ("TADI") (informative)

Afin d'assurer la conformité du protocole SCL à l'IEC 61784-3:2021 (voir 11.1), des considérations doivent être prises et des calculs être effectués pour démontrer que la PFH globale concernant

- l'opportunité,
- l'authenticité (déguisement, bouclage), et
- l'intégrité des données,

est inférieure à $10^{-9}/h$ (Figure 44).

Les mesures pour l'opportunité sont spécifiées en 11.4.6 et consistent en un compteur 3 bits combiné à un temporisateur de chien de garde à l'intérieur de la SCL. Ceci ajoute une mesure supplémentaire pour la vérification de l'intégrité des données et des états de la SCL. Cependant, conformément à l'IEC 61784-3:2021, 5.8.8, en raison de la nature point à point de la SDCI-FS et du fait qu'aucun élément de stockage n'est admis, le taux d'erreurs résiduelles pour l'opportunité, RR_T est de 0.

La mesure pour la vérification de l'authenticité et du déguisement est fournie par la transmission explicite d'un numéro de port dans la SPDU, lequel est vérifié par le point d'extrémité SCL récepteur. Ceci ajoute une mesure supplémentaire pour la vérification de l'intégrité des données. Cependant, conformément à l'IEC 61784-3:2021, 5.8.7, en raison de la nature point à point de la SDCI-FS, le taux d'occurrence des SPDU mal acheminées (R_A) est de 0 et, par conséquent, le taux d'erreurs résiduelles pour l'opportunité, RR_A est de 0. De la même manière, le taux d'occurrence des SPDU déguisées (R_M) est de 0 et, par conséquent, le taux d'erreurs résiduelles pour le déguisement, RR_M est de 0.

Les mesures pour l'intégrité des données sont fournies par la signature CRC sur l'ensemble des données de processus de sécurité, le numéro de port et le Control&MCnt ou le Status&DCnt, respectivement, comme indiqué en 11.4.3. Le calcul de la probabilité d'erreurs résiduelles peut être effectué à une probabilité d'erreurs sur les éléments binaires de 10^{-2} à l'aide des informations données dans l'Annex D du présent document. Les concepteurs doivent être attentifs à la fréquence d'échantillonnage maximale spécifiée en 11.4.7. L'appareil de surveillance PFH limite le nombre maximal de SPDU corrompues qui sont détectées pendant un intervalle de temps donné (voir 11.5.5).

11.4.9 Types de données pour SDCI-FS

11.4.9.1 Généralités

Les structures de données de sécurité fonctionnelle échangées de manière cyclique entre un Appareil FS et un Maître FS comprennent les données d'E/S de processus FS et la queue de protocole SDCI-FS. Elles sont transmises dans les PDU de sécurité.

Les structures de données de sécurité fonctionnelle échangées de manière acyclique sont transmises dans des conteneurs de données à la demande (OD) SDCI, soit à partir d'un outil dédié, soit à partir du programme utilisateur d'un AP FS. Dans ce cas, des mécanismes de sécurisation supplémentaires (signature CRC, par exemple) sont exigés à chaque transfert ou après un bloc de paramètres.

11.4.9.2 Données d'E/S de processus FS (PDin et PDout)

Pour les données d'E/S de processus FS, un ensemble bien défini de types de données et une description correspondante sont spécifiés pour l'Appareil FS et le Maître FS, pour un traitement et un mapping corrects vers les FSCP de niveau supérieur. Le Tableau 34 répertorie les trois types de données admis (voir Annex C).

Tableau 34 – Types de données d'E/S de processus FS

Type de données	Codage	Longueur	Voir IEC 61131-9:—	Exemple d'Appareil
BooleanT/bit	BooleanT ("forme encapsulée" à des fins d'efficacité, aucune structure WORD); l'affectation de noms de signal aux bits est possible.	1 bit	F.2.2, Tableau F.22 et Figure F.9	Commutateur de proximité
IntegerT(16)	IntegerT (énuméré ou signé)	2 octets	F.2.4; Tableau F.4, Tableau F.7 et Figure F.3	Champs de protection du lecteur laser
IntegerT(32)	IntegerT (énuméré ou signé)	4 octets	F.2.4, Tableau F.4, Tableau F.6 et Figure F.3	Codeur ou mesure de longueur ($\approx \pm 2$ km, résolution 1 μ m)

11.4.9.3 Qualificatif

Normalement, les Appareils FS n'exigent pas de qualificatifs (voir 11.11.2). Les bits qualificatifs sont configurés avec les données de processus (ou données de sécurité = SD) lors du mapping vers le système FSCP de niveau supérieur. Les structures de données dépendent des règles de ces systèmes FSCP.

Dans le cas des bornes FS (voir 11.11.3), les règles du Tableau 35 pour la disposition des données binaires et numériques et de leurs bits qualificatifs s'appliquent.

Tableau 35 – Règles de disposition des valeurs et des qualificatifs

N°	Règle
1	Seuls les types de données Boolean (DI, DO) et IntegerT(16) ou IntegerT(32) (AI, AO) doivent être utilisés. Toute valeur doit être affectée à l'une de ces catégories.
2	Les valeurs Boolean précèdent les valeurs Integer
3	Les valeurs IntegerT(16) précèdent les valeurs IntegerT(32)
4	Les valeurs précèdent les qualificatifs au niveau de l'octet
5	Les qualificatifs suivent directement les valeurs d'entrée. En l'absence de valeurs d'entrée, seuls les qualificatifs des valeurs de sortie sont placés.
6	Les qualificatifs des valeurs d'entrée précèdent les qualificatifs des valeurs de sortie
7	Les qualificatifs de chaque catégorie (DI, DO, AI, AO) sont encapsulés séparément au niveau de l'octet
8	Si des types de données sont manquants, les types de données restants sont récupérés

Le Tableau 36 indique l'ordre des valeurs et des qualificatifs.

Tableau 36 – Ordre des valeurs et des qualificatifs

Ordre	Vers le Maître FS	Vers l'Appareil FS
1	Valeur DI	Valeur DO
2	Valeur AI	Valeur AO
3	Qualificatif DI	–
4	Qualificatif AI	–
5	Qualificatif DO	–
6	Qualificatif AO	–

11.4.9.4 Queue de protocole SDCI-FS

Les types de données pour la queue de protocole ("code de sécurité") sont spécifiés en C.5.

11.4.9.5 Paramètres FSP et FST

Aucune définition de type de données particulière n'est exigée.

11.5 Comportement SCL

11.5.1 Généralités

Les diagrammes d'états de la couche de communication de sécurité du Maître FS et de l'Appareil FS sont conçus en utilisant les mesures de sécurité choisies dans le Tableau 27 et les signaux de protocole en 11.4.6.

11.5.2 Diagramme d'états SCL du Maître FS

La Figure 45 représente le diagramme d'états du Maître FS pour la communication point à point d'une SDCI câblée.

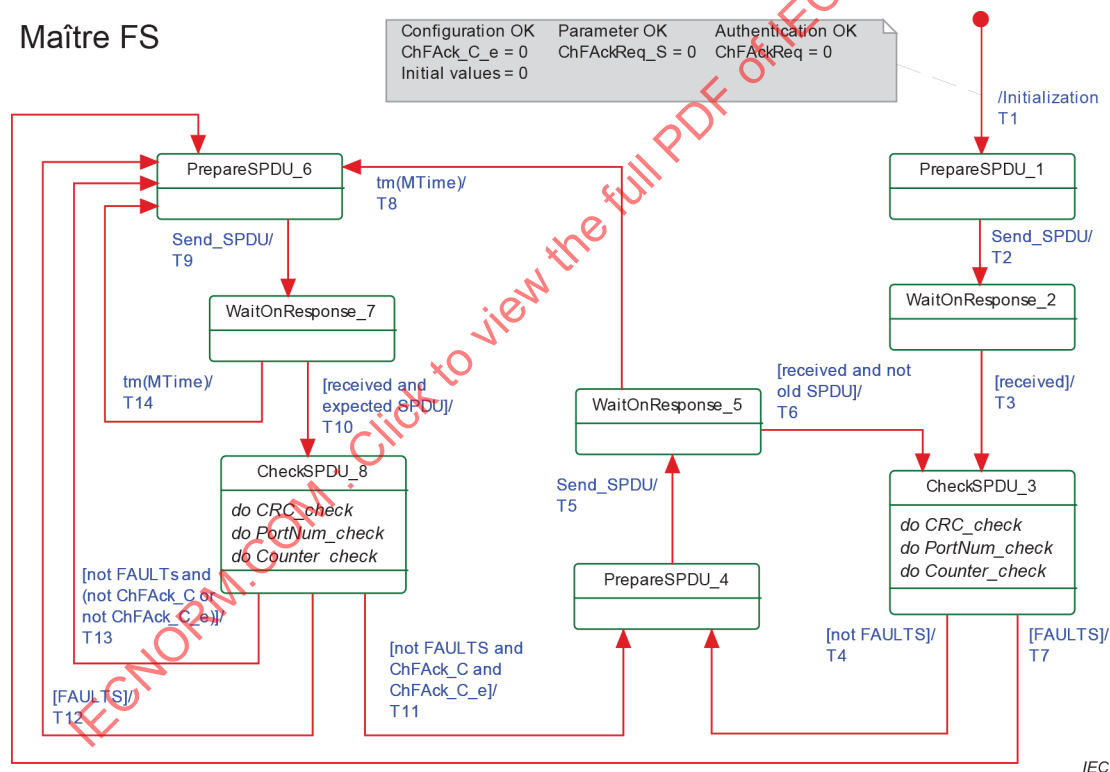


Figure 45 – Diagramme d'états SCL du Maître FS

Les termes utilisés dans la Figure 45 sont définis dans le Tableau 37.

Tableau 37 – Définition des termes utilisés dans le diagramme d'états SCL du Maître FS

Terme	Définition
ChFack_C	Acquittement de l'opérateur pour la fonction de sécurité au travers de la passerelle FS
FAULTS	MTimeout: expiration du Maître FS lors de l'attente d'une réponse SPDU de l'Appareil FS, ou MCommErr: le Maître FS détecte une réponse SPDU de l'Appareil FS corrompue (y compris une erreur de compteur/port), ou DTimeout: l'Appareil FS a signalé une expiration de sa SCL au travers de l'octet Status&DCnt, ou DCommErr: l'Appareil FS a signalé un CRC (y compris une erreur de compteur/port) par sa SCL au travers de l'octet Status&DCnt

Tableau 38 – Etats et transitions SCL du Maître FS

NOM D'ÉTAT	DESCRIPTION DE L'ÉTAT		
Initialisation	Etat initial de l'instance SCL du Maître FS à la mise sous tension (un par port).		
1 PrepareSPDU	Préparation d'une SPDU (<i>normale</i>) pour l'Appareil FS. Envoi de la SPDU lorsqu'elle a été préparée.		
2 WaitOnResponse	La SCL attend une SPDU de l'Appareil FS. Une SPDU avec tous les octets "0" doit être ignorée.		
3 CheckSPDU	Vérification de l'UDSPD reçue pour contrôler l'absence de FAULTS (→ T4). En cas de FAULTS: erreurs dans l'octet Status&DCnt (DCommErr, DTimeout, SDset) → T7.		
4 PrepareSPDU	Préparation d'une SPDU (<i>normale</i>) pour l'Appareil FS. Envoi de la SPDU lorsqu'elle a été préparée.		
5 WaitOnResponse	La SCL attend la prochaine SPDU de l'Appareil FS qui ne transporte pas le DCount_i précédent. Une SPDU avec tous les octets "0" doit être ignorée.		
6 PrepareSPDU	Préparation d'une SPDU (<i>exceptionnelle</i>) pour l'Appareil FS (en raison de MTimeout, d'OpAck manquant ou de FAULTS).		
7 WaitOnResponse	La SCL attend la prochaine SPDU de l'Appareil FS qui ne transporte pas le DCount_i précédent. A réception → T10, après MTimeout → T14.		
8 CheckSPDU	Vérification de la SPDU reçue pour contrôler l'absence d'erreur CRC (MCommErr) et d'anomalies potentielles de l'Appareil FS dans l'octet Status&DCnt (DTimeout, DCommErr). Lorsqu'une anomalie est survenue, aucun redémarrage automatique d'une fonction de sécurité n'est admis tant qu'un signal d'acquittement de l'opérateur (ChFack_C) n'est pas arrivé (voir Figure 40). Conseil: un délai peut être exigé pour éviter l'impact d'un arrêt occasionnel du système.		

TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T1	0	1	utiliser SD, setSD = 1, SDset_S = 1 MCount = 0
T2	1	2	–
T3	2	3	–
T4	3	4	MCount = MCount + 1 si MCount = 8 alors MCount = 1 si SDset = 1 ou setSD_C = 1 alors utiliser SDin, SDset_S = 1 sinon utiliser PDin, SDset_S = 0 si setSD_C = 1 alors utiliser SDout, setSD = 1 sinon utiliser PDout, setSD = 0
T5	4	5	redémarrer MTimer
T6	5	3	–
T7	3	6	utiliser SD, setSD = 1, SDset_S = 1 MCount = MCount + 1 si MCount = 8 alors MCount = 1
T8	5	6	utiliser SD, setSD = 1, SDset_S = 1 MCount = 0

TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T9	6	7	redémarrer MTimer
T10	7	8	–
T11	8	4	ChFAckReq =0, ChFAckReq_S =0, ChFAck_C_e =0, MCount = MCount + 1 si MCount = 8 alors MCount = 1 si SDset =1 ou setSD_C =1 alors utiliser SDin, SDset_S =1 sinon utiliser PDin, SDset_S =0 si setSD_C =1 alors utiliser SDout, setSD =1 sinon utiliser PDout, setSD =0
T12	8	6	ChFAckReq =0, ChFAckReq_S =0, ChFAck_C_e =0, utiliser SD, setSD =1, SDset_S =1 MCount = MCount + 1 si MCount = 8 alors MCount = 1
T13	8	6	ChFAckReq =1, ChFAckReq_S =1, /*définir la demande de qualificatif/d'acquittement*/ si ChFAck_C = 0 alors ChFAck_C_e =1 utiliser SD, setSD =1, SDset_S =1 MCount = MCount + 1 si MCount = 8 alors MCount = 1
T14	7	6	ChFAckReq =0, ChFAckReq_S =0, ChFAck_C_e =0, utiliser SD, setSD =1, SDset_S =1 MCount = 0

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
MTimer	Temporisateur	Ce temporisateur vérifie l'arrivée de la prochaine SPDU valide depuis l'Appareil FS dans les temps. L'outil de Maître FS est chargé de définir ce temps de fonctionnement du chien de garde. La plage de valeurs est comprise entre 1 et 65 535 ms.
ChFAck_C_e	Drapeau	Cette variable (bit) auxiliaire assure le maintien de l'état de sécurité jusqu'à une modification de signal ChFAck_C de 0 → 1 (front). Sans ce mécanisme, un opérateur peut annuler des états de sécurité en activant définitivement le signal ChFAck_C.
FAULTS	Drapeaux	Le stockage permanent des erreurs ou défaillances suivantes peut être omis dans le Maître FS, s'il peut être admis par hypothèse que le système FSCP de niveau supérieur empêche le redémarrage automatique des fonctions de sécurité (aucune persistance de l'Appareil FS): - MCommErr ou MTimeout - DCommErr, y compris les erreurs de compteur/port (Status&DCnt bit 1 et PortNum) - DTimeout (Status&DCnt bit 0)
SPDU prévue	Protecteur	Compteur inversé en miroir (DCount_i = MCount inversé)
Aucune ancienne SPDU	Protecteur	Valeur de compteur ≠ valeur de la précédente SPDU
lancer CRC_check	Activité	La SCL calcule la signature CRC sur la SPDU reçue lorsque la valeur de signature = "0" et la compare avec la signature CRC reçue
lancer PortNum_check	Activité	La SCL vérifie si PortNum comporte le numéro de port correct du Maître FS
lancer Counter_check	Activité	La SCL vérifie si DCount comporte une valeur attendue (miroir)
NOTE Les variables sous ACTION sont définies en 11.3.		

11.5.3 Diagramme d'états SCL de l'Appareil FS

La Figure 46 représente le diagramme d'états de l'Appareil FS correspondant.

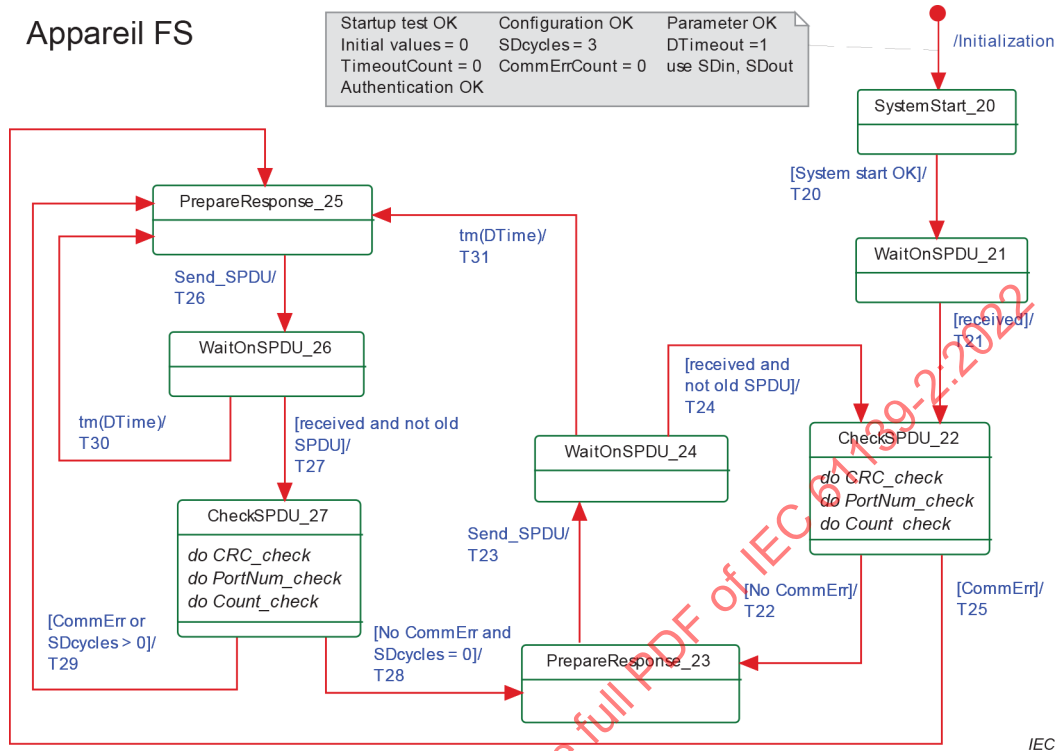


Figure 46 – Diagramme d'états SCL de l'Appareil FS

Les termes utilisés dans la Figure 46 sont définis dans le Tableau 39.

Tableau 39 – Définition des termes utilisés dans le diagramme d'états SCL de l'Appareil FS

Terme	Définition
CommErr	La SCL de l'Appareil FS a détecté une erreur CRC ou une erreur de compteur/port dans la SPDU reçue
CommErrCount	Voir ÉLÉMENTS INTERNES dans le Tableau 40
SDcycles	Voir ÉLÉMENTS INTERNES dans le Tableau 40
DTimeout	FSP_WatchdogTime expiré
TimeoutCount	Voir ÉLÉMENTS INTERNES dans le Tableau 40

Tableau 40 – Etats et transitions SCL de l'Appareil FS

NOM D'ÉTAT	DESCRIPTION DE L'ÉTAT		
Initialisation	Initialisation de l'Appareil FS à la mise sous tension. A la mise sous tension, l'Appareil FS (actionneur) définit le PDout sur "0". A la mise sous tension, l'Appareil FS (capteur) envoie "0".		
20 SystemStart	Immédiatement après le paramétrage FSP, l'Appareil FS définit PDout sur les valeurs SDout. Immédiatement après le paramétrage FSP, il envoie les données de processus (PD).		
21 WaitOnSPDU	La SCL attend la prochaine SPDU du Maître FS. Une SPDU avec tous les octets "0" doit être ignorée.		
22 CheckSPDU	Vérification de la SPDU reçue du Maître FS pour contrôler l'absence d'erreurs CRC; définition de ChFackReq_DC = ChFackReq. Lorsque le protecteur "No CommErr" = true → T22. Lorsque le protecteur "CommErr" = true → T25.		
23 PrepareResponse	Préparation d'une réponse SPDU (<i>normale</i>) pour le Maître FS (message de réponse)		
24 WaitOnSPDU	La SCL attend la prochaine SPDU (<i>normale</i>) du Maître FS qui ne transporte pas le MCount précédent. Après expiration de FSP_WatchdogTime → T31. Lorsque la SPDU a été reçue et que le protecteur "MCounter_incremented" = true → T24 (cycle <i>normal</i>).		
25 PrepareResponse	Préparation d'une réponse SPDU (<i>exceptionnelle</i>) pour le Maître FS (en raison de DTimeout ou de DCommErr = bits de rapport d'erreur dans l'octet Status&DCnt).		
26 WaitOnSPDU	La SCL attend la prochaine SPDU du Maître FS qui ne transporte pas le MCount précédent. Une SPDU avec tous les octets "0" doit être ignorée. Après expiration de FSP_WatchdogTime → T30. Lorsque la SPDU a été reçue et que le protecteur "MCounter_incremented" = true → T27.		
27 CheckSPDU	Vérification de la SPDU reçue du Maître FS pour contrôler l'absence d'erreurs CRC; définition de ChFackReq_DC = ChFackReq. Lorsque le protecteur "No CommErr and SDcycles =0" = true → T28. Lorsque le protecteur "CommErr or SDcycles >0" = true → T29.		
TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T20	20	21	–
T21	21	22	–
T22	22	23	utiliser PDin_D, DCommErr = 0, /*Status&DCnt, bit 1*/ DTimeout = 0, /*Status&DCnt, bit 0*/ DCount_i = MCount_inv, redémarrer DTimer si SDcycles <> 0 alors utiliser SDout, setSD_DC=1, SDset =1, /*pendant SDcycles: SDset_p =1*/ SDcycles = SDcycles – 1 alors utiliser PDout, setSD_DC=0, SDset = 0 si setSD =1 /*use_SD =1*/ alors utiliser SDout, setSD_DC=1,
T23	23	24	si SDset_DS = 1 /*anomalie de l'Appareil FS*/ alors SDset = 1
T24	24	22	–
T25	22	25	utiliser PDin_D, utiliser SDout, setSD_DC=1, SDset = 1, DCommErr =1, /*Status&DCnt, bit 1*/ CommErrCount = 1, DCount_i = MCount_inv, SDcycles = 3, redémarrer DTimer
T26	25	26	–
T27	26	27	–
T28	27	23	utiliser PDin_D, utiliser SDout, setSD_DC=0, SDset = 0, DCount_i = MCount_inv, DCommErr =0, /*Status&DCnt, bit 1*/ DTimeout =0, /*Status&DCnt, bit 0*/ redémarrer DTimer,

TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T29	27	25	utiliser PDin_D, utiliser SDout, setSD_DC=1, SDset = 1, DCount_i = MCount_inv, redémarrer DTimer si CommErr alors DCommErr = 1, /*Status&DCnt, bit 1*/ CommErrCount = 1, SDcycles = 3, alors SDcycles = SDcycles -1 si CommErrCount = 1 alors DCommErr = 1, /*Status&DCnt, bit 1*/ CommErrCount = 0 alors DCommErr = 0 /*Status&DCnt, bit 1*/ si TimeoutCount = 1 alors DTimeout = 1, /*Status&DCnt, bit 0*/ TimeoutCount = 0 alors DTimeout = 0 /*Status&DCnt, bit 0*/
T30	26	25	utiliser PDin_D, utiliser SDout, setSD_DC=1, SDset =1, DTimeout =1, /*Status&DCnt, bit 0*/ TimeoutCount =1, SDcycles = 3, redémarrer DTimer, DCount_i = MCount_inv
T31	24	25	utiliser PDin_D, utiliser SDout, setSD_DC=1, SDset =1, DTimeout =1, /*Status&DCnt, bit 0*/ TimeoutCount =1, SDcycles = 3, redémarrer DTimer, DCount_i = MCount_inv

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
MCount_inv	Variable	Valeur inverse de la valeur MCount actuelle
SDcycles	Compteur	Ce compteur décrémental est utilisé pour provoquer le paramétrage SCL SDout et SDset pendant au moins 3 cycles au démarrage et après une anomalie. La plage de valeurs est comprise entre 3 et 0.
CommErrCount	Compteur	Ce compteur décrémental permet de s'assurer que le bit "DCommErr" de l'octet Status&DCnt est défini pour au moins 1 cycle ou au plus 2 cycles. La plage de valeurs est comprise entre 1 et 0.
TimeoutCount	Compteur	Ce compteur décrémental permet de s'assurer que le bit "DTimeout" de l'octet Status&DCnt est défini pour au moins 1 cycle ou au plus 2 cycles. La plage de valeurs est comprise entre 1 et 0.
lancer CRC_check	Activité	La SCL calcule la signature CRC sur la SPDU reçue lorsque la valeur de signature = "0" et la compare avec la signature CRC reçue
lancer PortNum_check	Activité	La SCL vérifie si PortNum comporte le numéro de port correct du Maître FS
lancer Counter_check	Activité	La SCL vérifie si MCount comporte "0" (première SPDU), ou "0" ou la valeur suivante attendue (toutes les autres SPDU)
NOTE Les variables sous ACTION sont définies en 11.3.		

Il est très peu probable qu'un Appareil FS reçoive des SPDU avec tous les octets sur "0". La SCL de l'Appareil FS doit ignorer de telles SPDU. Normalement, au moins la signature CRC est à "1" si les données de processus et l'octet Control sont à "0", conformément aux règles données en 11.4.7.

11.5.4 Diagrammes séquentiels pour plusieurs cas d'utilisation

11.5.4.1 Maître FS et Appareil FS tous deux sous tension (ON)

La Figure 47 représente le diagramme séquentiel d'un démarrage normal du Maître FS et de l'Appareil FS.



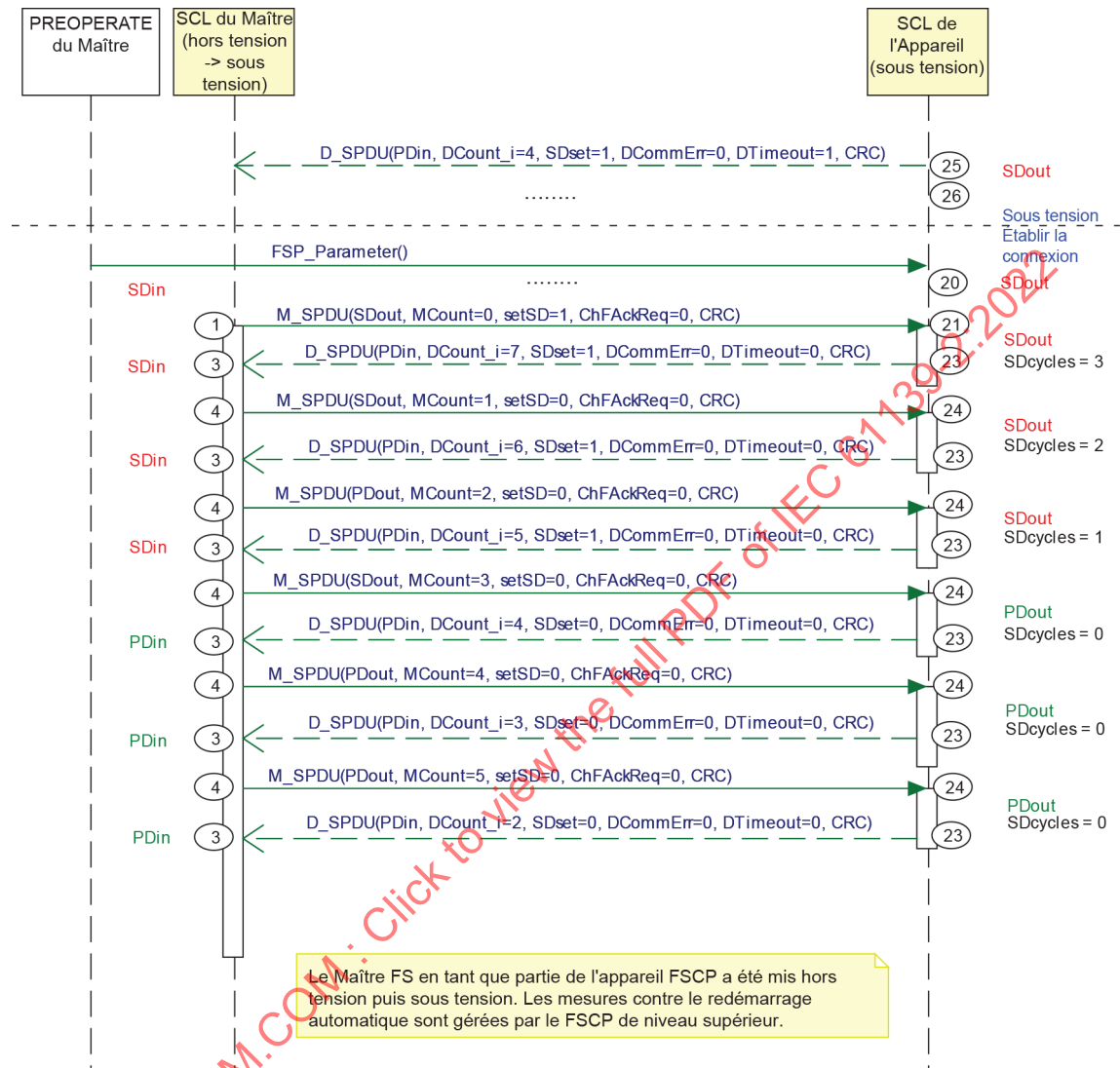
IEC

Figure 47 – Maître FS et Appareil FS tous deux sous tension (ON)

A la mise sous tension, le Maître FS et l'Appareil FS fournissent respectivement SDin (PDin = "0") et SDout (PDout = "0"). Tous deux gardent ces valeurs pendant 3 cycles de communication (SDcycles) avant de basculer en mode normal, où seules les valeurs MCounter et DCounter varient.

11.5.4.2 Maître FS hors tension puis sous tension (OFF → ON)

La Figure 48 représente le diagramme séquentiel du fonctionnement normal lorsque le Maître FS est mis hors tension puis à nouveau sous tension.



IEC

Figure 48 – Maître FS hors tension puis sous tension (OFF → ON)

La partie communication de l'Appareil FS est toujours alimentée par le Maître FS. Par conséquent, si le Maître FS est mis hors tension puis à nouveau sous tension, l'Appareil FS se contente de suivre, et un démarrage normal se produit. Etant donné que le Maître FS fait partie d'un système FSCP de niveau supérieur, le système FSCP est chargé d'empêcher le redémarrage automatique des fonctions de sécurité dans ce cas.

11.5.4.3 Appareil FS avec démarrage SCL différé

La Figure 49 représente le diagramme séquentiel lorsque le démarrage SCL de l'Appareil FS est différé.

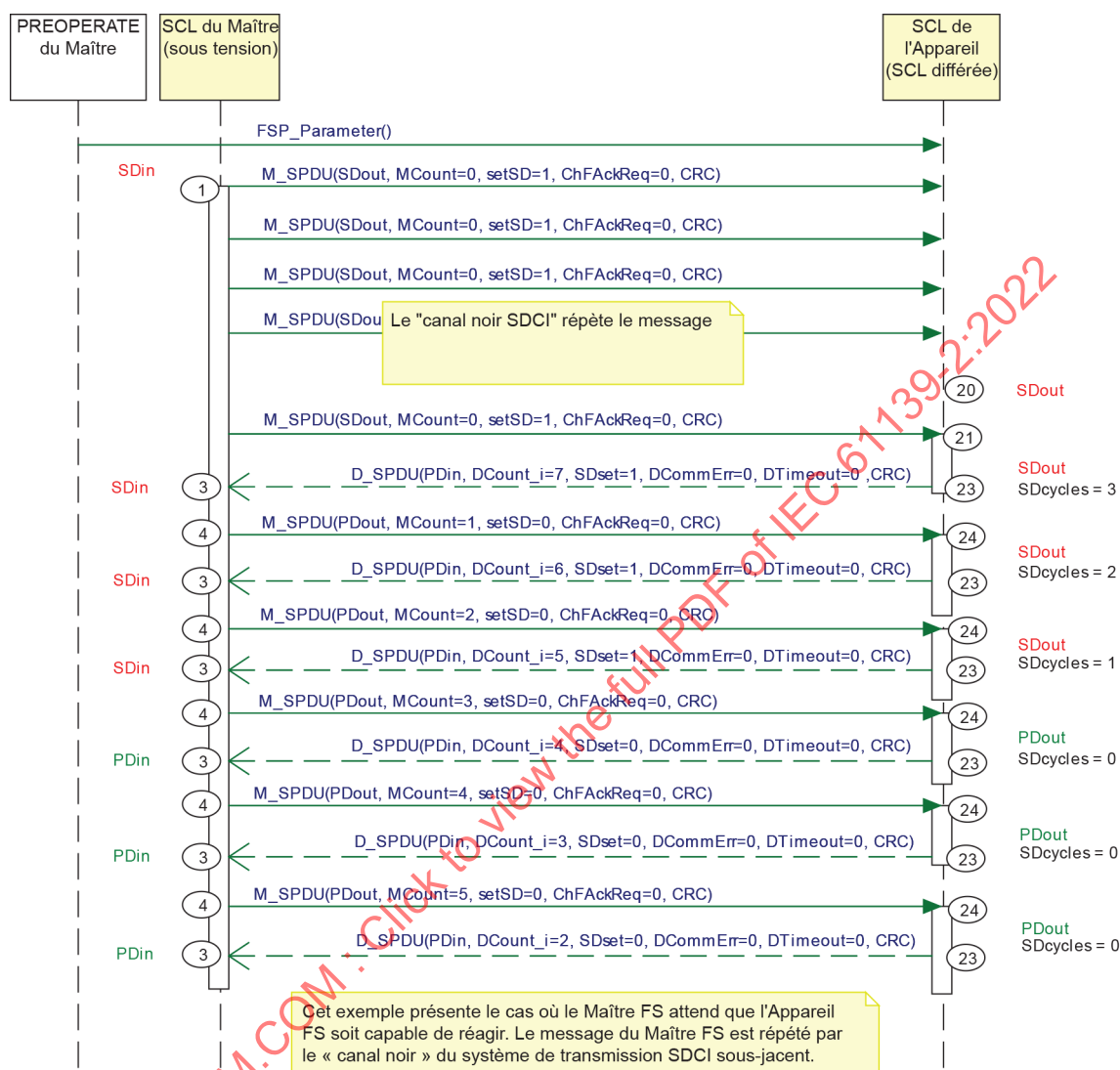


Figure 49 – Appareil FS avec démarrage SCL différé

Ce diagramme décrit comment la SCL d'un Maître FS attend la SCL de l'Appareil FS en cas de retard. La SPDU initiale du Maître FS est répétée par le système de transmission SDCl (canal noir) jusqu'à ce que la SCL de l'Appareil FS soit prête à passer à l'état 21.

Tant que la SCL de l'Appareil FS n'est pas prête, la SPDU de réponse contient tous les "0" et la SCL du Maître FS ignore cette SPDU. Le PDvalid/invalid de la SDCl est réservé à la partie non relative à la sécurité de l'ensemble du message.

11.5.4.4 Appareil FS hors tension puis sous tension (OFF -> ON)

La Figure 50 représente le diagramme séquentiel lorsque l'Appareil FS est mis hors tension puis à nouveau sous tension.

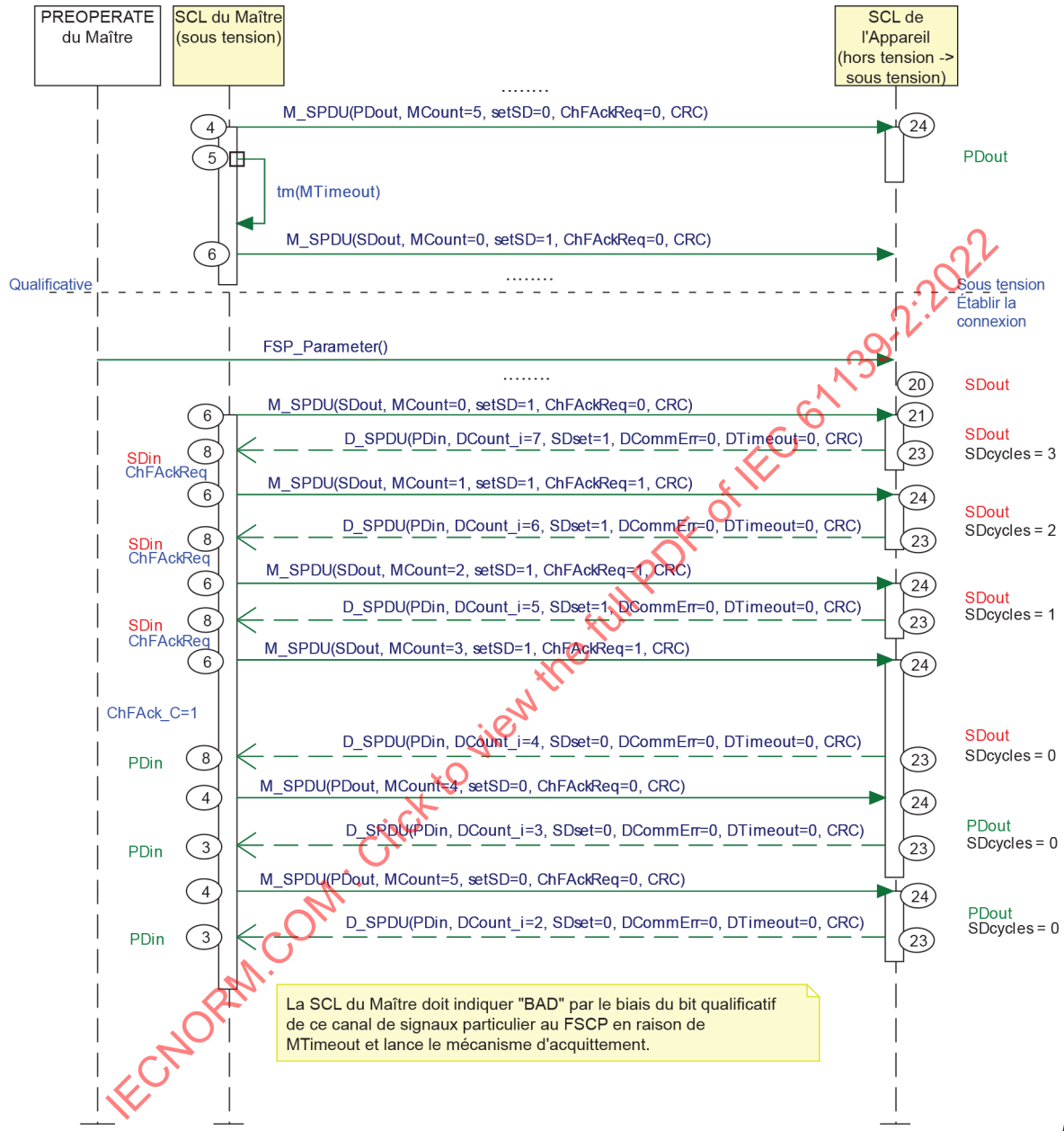


Figure 50 – Appareil FS hors tension puis sous tension (OFF -> ON)

Ce cas prend par exemple pour hypothèse un bref débranchement puis le rebranchement de l'Appareil FS, ce qui provoque une FAULT (MTimeout) du côté du Maître FS. Cette FAULT entraîne la définition d'un bit qualificatif qui exige, par ChFackReq (=1), un acquittement par ChFack_C (=1). Le Maître FS et l'Appareil FS conservent SDin et SDout jusqu'à l'arrivée de l'acquittement.

11.5.4.5 Détection d'une erreur de signature CRC par le Maître FS

La Figure 51 représente le diagramme séquentiel lorsque le Maître FS détecte une erreur de signature CRC.

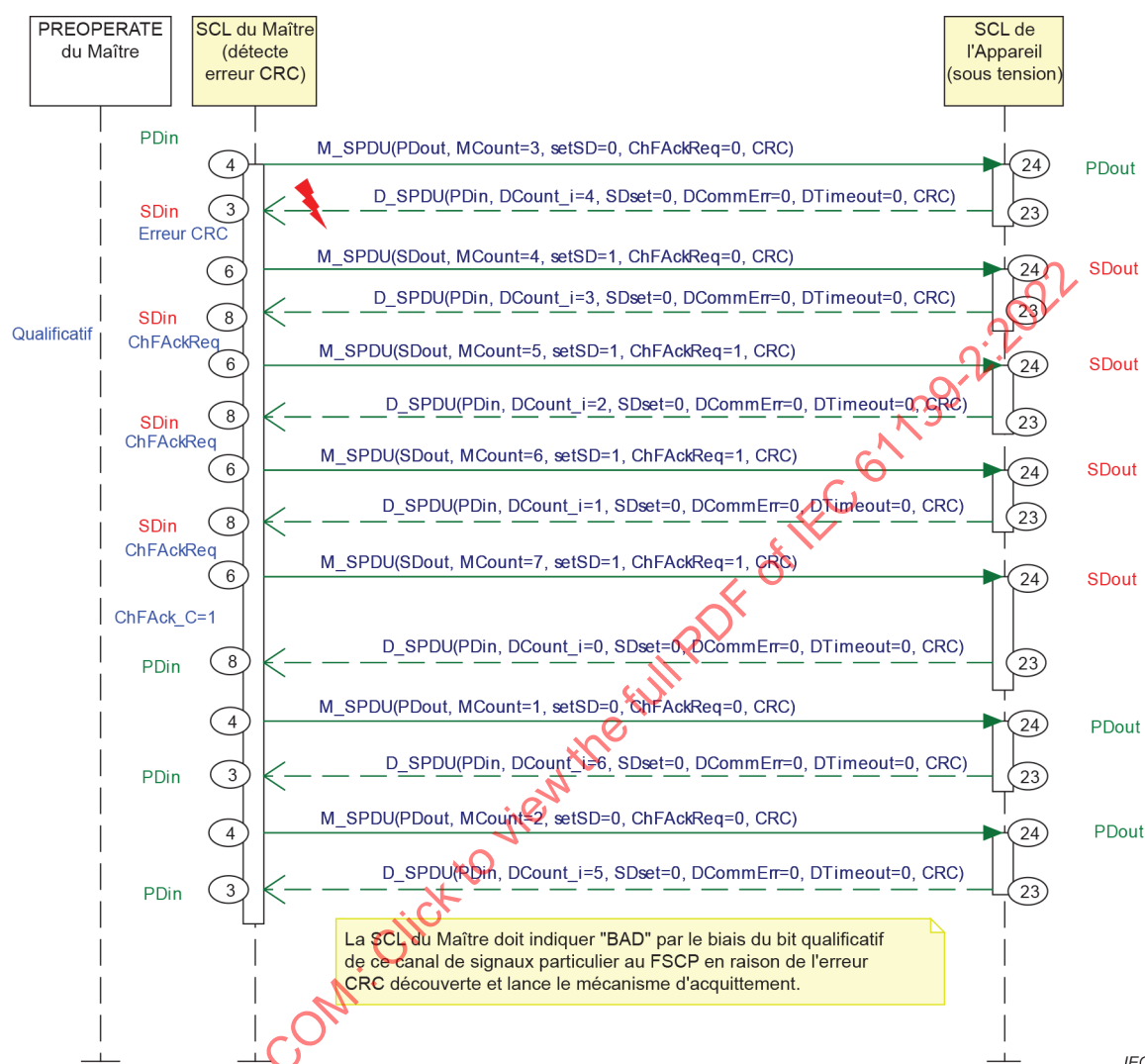


Figure 51 – Détection d'une erreur de signature CRC par le Maître FS

Le Maître FS a reçu une SPDU avec des données falsifiées ou une signature CRC falsifiée, ce qui donne lieu à une "erreur CRC" (MCommErr). Le Maître FS et l'Appareil FS basculent respectivement sur SDin et SDout, et le Maître/la passerelle FS crée un bit qualificatif et indique un signal ChAckReq. Ce signal est également indiqué à l'Appareil FS par ChAckReq (= 1) pour une indication par LED (diode électroluminescente) à l'utilisateur.

Le Maître FS et l'Appareil FS conservent SDin et SDout jusqu'à l'arrivée de l'acquittement ChAck_C (=1).

11.5.4.6 Détection d'une erreur de signature CRC par l'Appareil FS

La Figure 52 représente le diagramme séquentiel lorsque l'Appareil FS détecte une erreur de signature CRC.

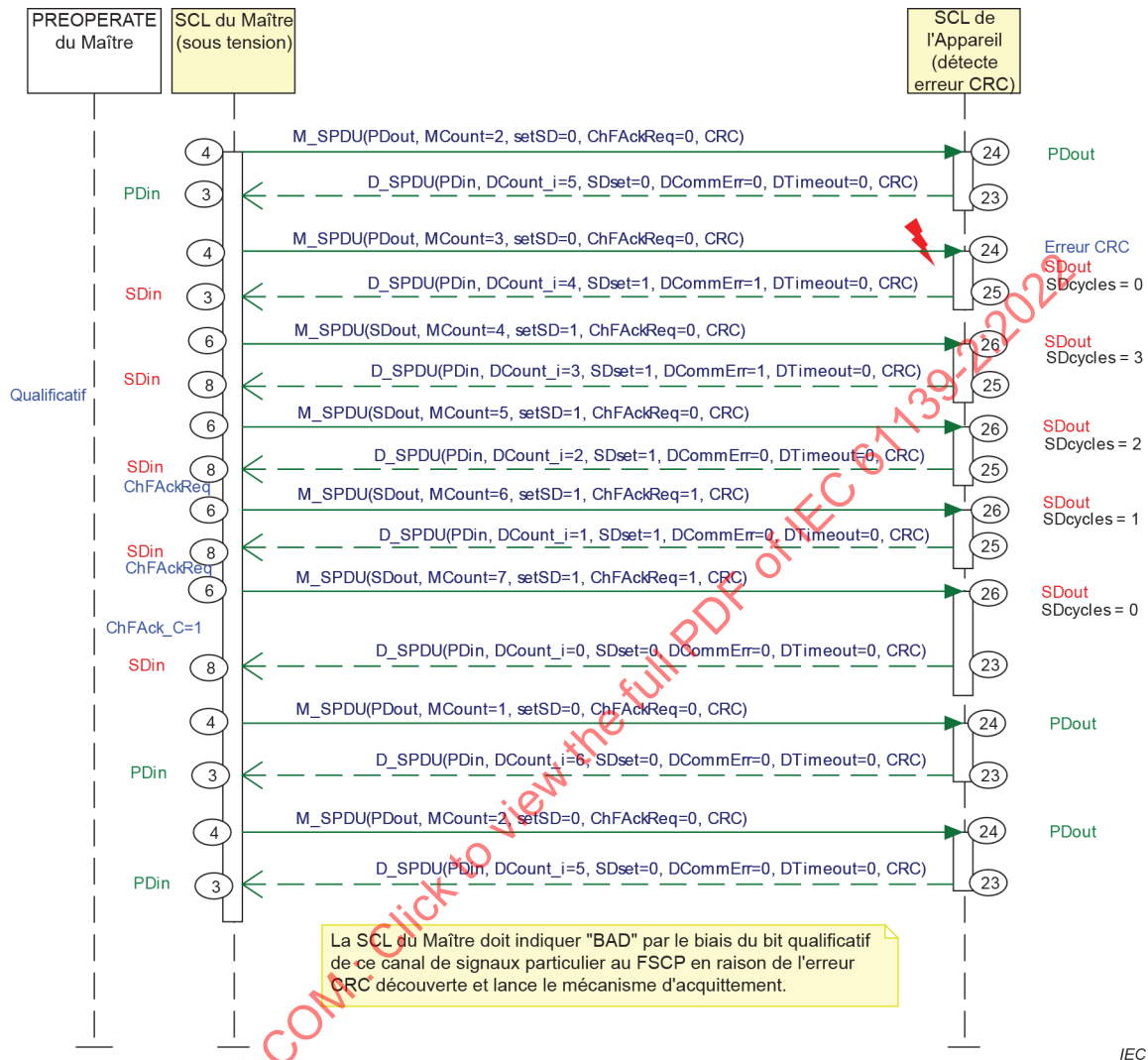


Figure 52 – Détection d'une erreur de signature CRC par l'Appareil FS

L'Appareil FS a reçu une SPDU avec des données falsifiées ou une signature CRC falsifiée, ce qui donne lieu à une "erreur CRC" (DCommErr). Le Maître FS et l'Appareil FS basculent respectivement sur SDin et SDout en raison des informations de l'octet Status de l'Appareil FS (SDset=1 et DCommErr=1). Le Maître/la passerelle FS crée un bit qualificatif et indique un signal ChFAckReq. Ce signal est également indiqué à l'Appareil FS par ChFAckReq (= 1) pour une indication par LED (diode électroluminescente) à l'utilisateur.

L'Appareil FS exécute 3 Sdcycles, puis le Maître FS et l'Appareil FS conservent SDin et SDout jusqu'à l'arrivée de l'acquittement ChFAck_C (=1).

11.5.5 Surveillance des temps de sécurité

La Figure 53 représente les temps SDCI et les temps de sécurité.

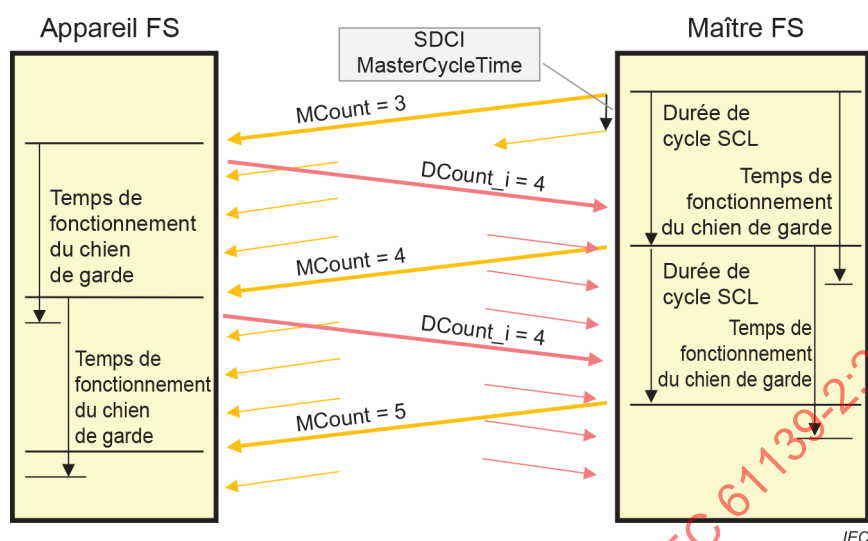


Figure 53 – Surveillance de la durée de cycle SCL

Le système SDCI de base ("canal noir") transmet les SPDU dans la MasterCycleTime (voir IEC 61131-9), du Maître FS à l'Appareil FS et inversement. Une même SPDU, par exemple avec MCount = 3, peut être envoyée à plusieurs reprises avant que la couche de communication de sécurité (SCL) ne lance le cycle SCL suivant, avec MCount = 4. En parallèle, le Maître FS a reçu la réponse SPDU de l'Appareil FS avec DCount_i = 4.

Le Tableau 41 indique les contraintes de temporisation.

Tableau 41 – Contraintes de temporisation

Élément	Contraintes	
Synchronisation	A chaque démarrage et après une expiration du Maître FS, la SCL du Maître FS utilise MCount = 0.	
Durée de cycle SCL	La durée de cycle SCL comprend le temps de transmission de la SPDU du Maître FS, le temps de traitement de l'Appareil FS, le temps de transmission de la SPDU de réponse de l'Appareil FS et le temps de traitement du Maître FS jusqu'à la prochaine SPDU du Maître FS (voir Figure 53)	
Temps de fonctionnement du chien de garde	Toute la durée de cycle SCL est surveillée par le temporisateur de chien de garde, dont la valeur temporelle est définie par le paramètre FSP_Watchdog (voir A.2.6).	
Vérification du compteur	Les valeurs du compteur sont incluses dans le calcul de la signature CRC cyclique. Une valeur de signature CRC incorrecte entraîne d'emblée un état de sécurité. La vérification immédiate du compteur dans certains états est utilisée pour éliminer les "SPDU obsolètes".	
Répétition	Répétition lorsque la détection des signatures CRC incorrectes n'est pas fournie.	
Appareil de surveillance PFH	Le Maître FS contient les informations relatives à la fiabilité des transmissions SPDU depuis et vers l'Appareil FS (voir Tableau 32, bit 1). Par conséquent, le Maître FS surveille la fréquence moyenne de défaillance dangereuse dans un délai donné (temps de surveillance PFH). Le diagramme d'états du Maître FS est conçu de sorte que toute SPDU corrompue entraîne toujours un état de sécurité. Dans le cas peu probable de détection d'une SPDU corrompue pendant la production ou l'exploitation, l'opérateur responsable joue le rôle d'appareil de surveillance PFH et peut tolérer et acquiescer l'indication. Dans le cas d'indications fréquentes (plus d'une fois par temps de surveillance PFH), il convient de vérifier la qualité de l'installation ou de la transmission (voir H.6).	
Temps de surveillance PFH (h)	10	FSP_ProtMode = 0x01; CRC-16, voir A.2.5
	10	FSP_ProtMode = 0x02; CRC-32, voir A.2.5

11.5.6 Réaction en cas de dysfonctionnement

11.5.6.1 Généralités

Les erreurs de communication potentielles sont spécifiées du 11.5.6.2 au 11.5.6.10. Ces paragraphes sont issus du 5.3 de l'IEC 61784-3:2021 et renvoient au Tableau 27 du présent document. Des notes supplémentaires sont fournies pour indiquer le comportement classique du canal noir SDCI.

11.5.6.2 Corruption

Les messages peuvent être corrompus par des erreurs internes à un élément de la communication, des erreurs sur le support de transmission ou des perturbations entre les messages.

NOTE 1 La falsification de bits dans les messages pendant le transfert est un phénomène normal pour tout système de communication normalisé. De telles erreurs sont détectées sur les récepteurs qui présentent une probabilité élevée en utilisant une fonction de hachage, ou une somme de contrôle avec une SDCI (CKT ou CKS), et le message est ignoré (IEC 61131-9:—, A.1). Après deux nouvelles tentatives, le Maître lance un redémarrage complet avec réactivation.

NOTE 2 Si les procédures de rétablissement ou de répétition durent plus longtemps qu'un délai spécifié, un message est classé comme un "retard inacceptable" (voir 11.5.6.6).

Contre-mesures:

La signature CRC spécifiée en 11.4.7 détecte les erreurs sur les éléments binaires dans les messages entre le Maître FS et l'Appareil FS dans la mesure exigée par les applications SIL3. La signature CRC est générée sur l'ensemble de la SPDU, y compris les données PD ou SD, le numéro de port et l'octet Control&MCnt ou Status&DCnt pour la communication cyclique.

Au démarrage, les paramètres FSP sont envoyés une fois à l'Appareil FS par le biais des services ISDU. Ils sont sécurisés par la signature FSP_ProtParCRC 16 bits. Par hypothèse, la fréquence de cette occurrence est de 1/jour en tant que paramètre pour le calcul du taux d'erreurs résiduelles.

11.5.6.3 Répétition non prévue

Les messages sont répétés à la suite d'une erreur, d'une anomalie ou d'une perturbation.

NOTE La répétition par l'émetteur constitue une procédure normale lorsqu'aucun acquittement /aucune réponse attendu(e) n'est reçu(e) depuis la station cible ou lorsqu'une station réceptrice détecte l'absence d'un message et demande sa retransmission.

Contre-mesures:

Les données dans le canal noir sont transférées de manière cyclique. Par conséquent, un message/une SPDU incorrect(e) avec la dernière valeur de compteur reçue qui est inséré(e) une fois est ignoré(e). Le retard possible d'une demande peut donc être un DTime ou un MTime, respectivement.

11.5.6.4 Séquence incorrecte

La séquence prédéfinie (par exemple, nombres naturels, références temporelles) associée aux messages d'une source particulière est incorrecte en raison d'une erreur, d'une anomalie ou d'une perturbation.

NOTE Dans une SDCI, une seule séquence est active à partir d'une source, le gestionnaire de messages.

Contre-mesures:

Le récepteur détecte toute séquence incorrecte en raison de l'attente séquentielle rigoureuse des valeurs MCount et DCount.

11.5.6.5 Perte

Un message ou un acquittement n'est pas reçu en raison d'une erreur, d'une anomalie ou d'une perturbation.

Contre-mesures:

Les informations perdues sont détectées en modifiant et en examinant rigoureusement MCount/DCount et/ou MTime/DTime dans la couche de communication de sécurité du récepteur respectif.

11.5.6.6 Retard inacceptable

Les messages peuvent être retardés au-delà de leur fenêtre temporelle d'arrivée admise, en raison, par exemple, de falsifications de bits sur le support de transmission, de lignes de transmission encombrées, de perturbations ou de l'envoi de messages par des éléments de la communication de nature à retarder ou à refuser les services (FIFO au niveau des commutateurs, ponts, routeurs, par exemple).

NOTE 1 La SDCl fournit une interface de communication point à point avec des séquences de messages définies et, par conséquent, la probabilité de congestion et de stockage des messages est extrêmement faible.

Contre-mesures:

Un compteur consécutif dans chaque message (MCount/DCount) ainsi qu'un temporisateur de chien de garde (MTime/DTime) détectent les retards inacceptables.

11.5.6.7 Insertion

Un message est reçu, qui se rapporte à une entité source imprévue ou inconnue, en raison d'une anomalie ou d'une perturbation.

NOTE 1 Ces messages s'ajoutent au flux de messages attendu. Ils ne peuvent pas être classés comme "corrects", "répétition non prévue" ou "séquence incorrecte" dans la mesure où ils ne comportent pas de source attendue.

NOTE 2 La SDCl fournit une interface de communication point à point (port) et, par conséquent, la probabilité d'insertion des messages est extrêmement faible.

Contre-mesures:

Le récepteur détecte toute séquence incorrecte en raison de l'attente séquentielle rigoureuse des valeurs MCount et DCount.

11.5.6.8 Déguisement

Une anomalie ou une perturbation provoque l'insertion d'un message associé à une entité source apparemment valide. Un message non relatif à la sécurité mal acheminé peut alors être reçu par un élément relatif à la sécurité, qui le traite alors comme un message relatif à la sécurité correct.

NOTE 1 Les systèmes de communication utilisés pour les applications relatives à la sécurité peuvent recourir à des contrôles supplémentaires pour détecter le déguisement, par exemple les identités de sources autorisées, les expressions d'adaptation ou la cryptographie.

NOTE 2 La SDCl fournit une interface de communication point à point (port) et, par conséquent, la probabilité d'insertion des messages est extrêmement faible.

Contre-mesures:

Dans le cas de données NSR à la place d'une SPDU normale, le mécanisme de signature CRC de la SCL détecte cet incident.

Après une modification de câblage, l'Appareil FS peut détecter les erreurs de connexion par les paramètres FSP_Authenticity1/2 et FSP_Port (voir A.2.1 et A.2.2) au démarrage.

11.5.6.9 Adressage

En raison d'une anomalie ou d'une perturbation, un message relatif à la sécurité est délivré au participant relatif à la sécurité inapproprié, qui traite alors le message reçu comme un message correct. Cela inclut le cas appelé erreur de bouclage, où l'émetteur du message reçoit en retour son propre message.

NOTE La probabilité de ne pas détecter un message non relatif à la sécurité mal acheminé est inférieure à la probabilité de ne pas détecter un message relatif à la sécurité mal acheminé, puisque les structures SPDU sont semblables en raison des procédures de protocole partagées.

La SDCI fournit une interface de communication point à point (port) et, par conséquent, la probabilité d'insertion des messages est extrêmement faible. Toutefois, le Maître FS peut utiliser des mécanismes de bus internes pour l'adressage des ports.

Contre-mesures:

Les erreurs d'adressage de port peuvent être détectées par le biais du numéro de port (PortNum) dans la SPDU.

Après une modification de câblage, l'Appareil FS peut détecter les erreurs de connexion par les paramètres FSP_Authenticity1/2 et FSP_Port (voir A.2.1 et A.2.2) au démarrage.

11.5.6.10 Bouclage

Le cas appelé erreur de bouclage, où l'émetteur du message reçoit en retour son propre message, constitue une erreur d'adressage spéciale.

Contre-mesures:

La SDCI-FS fournit des valeurs inversées de MCount en tant que DCount, et des valeurs inversées du numéro de port (PortNum) que renvoie l'Appareil FS.

11.5.7 Démarrage (communication)

Un Appareil FS démarre toujours après un Maître FS, car le Maître FS doit être le seul mettre sous tension au moins la partie communication de l'Appareil FS. Ces deux appareils exigent généralement un temps pour des autovérifications de sécurité, qui peuvent dépasser les temporisations normalisées définies dans l'IEC 61131-9.

En raison du comportement initial d'un Appareil FS en tant qu'OSSDe, le démarrage est coordonné et spécifié en 5.7, en 7.2 et en 7.3.

Le démarrage du système de communication SDCI sous-jacent est spécifié dans l'IEC 61131-9 et représenté de manière abstraite et simplifiée à la Figure 57 pour une meilleure compréhension. Tout écart de la signature CRC des paramètres de protocole ou d'authenticité FSP doit entraîner un état de sécurité du port du Maître FS particulier et empêcher le démarrage de la SCL.

11.6 Gestion de la SCL

11.6.1 Vue d'ensemble des paramètres (FSP et FST)

L'Annex A spécifie plusieurs paramètres de sécurité fonctionnelle pour les services de protocole de communication (FSP) ainsi qu'aux fins du traitement et de l'intégrité des paramètres de technologie de l'Appareil FS (FST).

Les paramètres sont divisés en 4 groupes:

- authenticité;
- communication de sécurité;
- description de la structure d'E/S FS;
- paramètres auxiliaires.

Les paramètres d'authenticité combinent l'ID de connexion de sécurité ("code A") du Maître FS (affecté par le système FSCP de niveau supérieur) avec le numéro de port de l'Appareil FS connecté. En raison de la nature point à point de la communication de l'Appareil FS avec son Maître, une vérification ponctuelle au démarrage suffit pour assurer l'authenticité (voir 11.7.4).

Les couches de communication de sécurité (SCL) exigent des paramètres pour les versions de protocole, les modes de protocole (comme CRC-16 ou CRC-32), un chien de garde pour l'opportunité, une signature CRC pour sécuriser les paramètres de technologie et une signature CRC pour sécuriser les paramètres de communication de sécurité.

Le groupe suivant contient une description de la structure de données d'E/S FS, que l'Appareil FS souhaite échanger avec l'hôte FSCP. Cette description simplifie le mapping vers la description que certains systèmes FSCP exigent pour la configuration. Pendant le processus de mapping, l'outil de Maître FS annexe les bits qualificatifs nécessaires à la passivation sélective du port.

Les paramètres auxiliaires sont spécifiés à plusieurs fins. Par exemple, pour sécuriser la description des paramètres de sécurité fonctionnelle dans l'IODD, pour prendre en charge le calcul automatique des valeurs approximatives des temps de réponse des fonctions de sécurité et pour donner des informations sur le temps d'autovérification au démarrage d'un Appareil FS jusqu'à l'apparition de l'impulsion Ready.

La Figure 54 donne une vue d'ensemble des composants et des activités autour du paramétrage.

Un Maître FS en tant que passerelle est fourni avec un fichier de description associé du système de niveau supérieur (FSCP). Avec l'aide d'un outil de développement et de ces paramètres, le Maître FS reçoit, lors de la mise en service, son ID de connexion FSCP ("code A" pour authenticité) et le temps de fonctionnement du chien de garde FSCP ("code T" pour timeliness, opportunité), par exemple. Ainsi, les cycles de communication FSCP sont indépendants des cycles de communication SDCI-FS entre le Maître FS et l'Appareil FS.

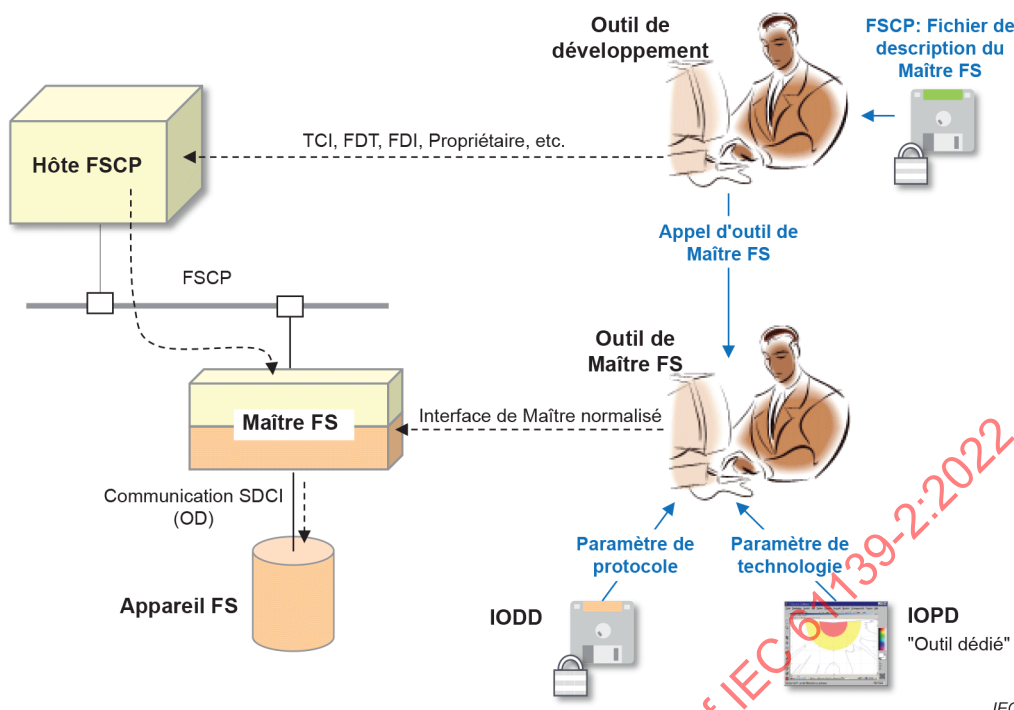


Figure 54 – Types de paramètres et affectations

Le Maître FS et son côté SDCI peuvent être configurés et paramétrés à l'aide de l'outil du Maître FS. L'IODD d'un Appareil FS contient, en plus des paramètres non relatifs à la sécurité, la section relative à la sécurité avec les paramètres de l'Annex A. Les paramètres peuvent être configurés hors site ou en ligne, de la même manière qu'avec un système non relatif à la sécurité, lors de "l'essai de mise en service" (voir Tableau G.1). Le paramètre d'authenticité FSCP peut être copié depuis l'affichage de l'outil de développement vers l'affichage de l'outil de Maître FS (voir A.2.1).

Il est possible de décrire un jeu réduit de paramètres de technologie (FST) de manière non relative à la sécurité, ce qui permet d'utiliser le mécanisme de stockage de données normalisé SDCI décrit en 9.4.

Toutefois, un outil dédié distinct (IOPD), développé conformément à l'IEC 61508-3, doit être utilisé pour calculer une signature CRC dans les instances des paramètres FST. Cette signature CRC doit être intégrée au paramètre FSP correspondant (voir A.2.8).

L'outil IOPD utilise une nouvelle interface de communication IOPD normalisée (DTI, voir Annex F) et le même chemin vers l'Appareil FS que vers l'outil de Maître FS.

11.6.2 Approches de paramétrage

11.6.2.1 Centré sur le Maître FS

La configuration et le paramétrage d'un système SDCI-FS autonome correspondent principalement à l'approche décrite en 11.6.1. Dans ce cas, l'authenticité fait appel à une valeur par défaut (voir A.2.1).

La Figure 54 représente un système grossièrement couplé, où le paramétrage est effectué dans la partie SDCI-FS. Dans le système FSCP, des structures de données d'E/S FS prédéfinies sont disponibles et peuvent être choisies lors de la mise en service.

11.6.2.2 Centré sur l'hôte FSCP

Certains champs d'application d'automatisation privilégient une approche centrée sur l'hôte FSCP. Dans ce cas, il est attendu que l'ensemble des paramètres soient stockés dans l'hôte FSCP et téléchargés au démarrage dans le Maître FS (sous-système FSCP), puis dans l'Appareil FS.

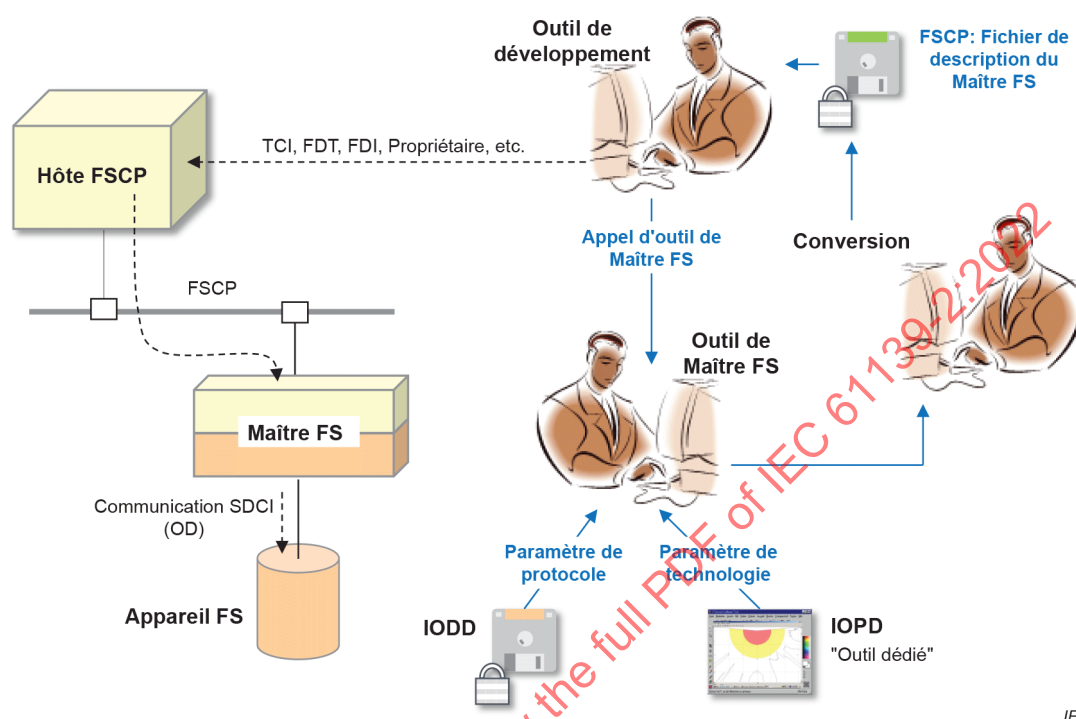


Figure 55 – Système centré sur l'hôte FSCP

En raison de la conception indépendante du bus de terrain de la SDCI et de la SDCI-FS, tous les paramètres peuvent être convertis dans le fichier de description d'appareil de bus de terrain, par exemple. L'un des objectifs de la SDCI-FS est d'optimiser la conception des paramètres de sécurité, afin de permettre une conversion efficace.

11.7 Mesures d'intégrité

11.7.1 Intégrité de l'IODD

Les paramètres spécifiés à l'Annex A sont codés dans un fichier IODD par XML. Afin de protéger la description des paramètres de sécurité dans ce fichier, une signature CRC ("FSP_ParamDescCRC") doit être calculée sur l'ensemble des parties relatives à la sécurité correspondantes (voir Annex D et E.5.6). En général, le fichier IODD se transmet de plusieurs façons et la signature CRC contribue à détecter les bits potentiellement brouillés.

11.7.2 Intégrité des outils

A l'ouverture de l'IODD, l'outil de Maître FS (interpréteur du fichier IODD) doit calculer la signature CRC pour les parties relatives à la sécurité et compare le résultat au paramètre "FSP_ParamDescCRC".

Au cours des manipulations des données dans l'outil de Maître FS et dans les outils de l'Appareil/IOPD ("outils dédiés"), telles que l'affichage, les modifications prévues, le stockage/l'extraction et le téléchargement, les valeurs des paramètres peuvent devenir incorrectes. Il incombe au concepteur de développer les outils en fonction du niveau de sécurité du logiciel demandé par l'ISO 13849-1 ou l'IEC 61508-3.

11.7.3 Intégrité de la transmission

Etant donné que la communication entre l'outil de Maître FS et l'Appareil FS est propriétaire, il incombe à l'outil de Maître FS d'assurer l'intégrité et l'authenticité de la transmission, par exemple au moyen de signatures CRC et/ou de relectures (voir Tableau 27 et D.3.1).

11.7.4 Enregistrement de vérification

Dans l'approche centrée sur le Maître FS ou dans l'approche centrée sur l'hôte FSCP, un FSP_VerifyRecord des données de paramétrage est stocké dans le Maître FS par port/Appareil FS, comme cela est indiqué à la Figure 56.

FSP_VerifyRecord	En-tête spécifique à l'Appareil FS	
Paramètre FSP	0	UIntegerT (32)
	1 FSP_Authenticity1 (Dépendant du FSCP)	
	2	
	3	
	4	UIntegerT (32)
	5 FSP_Authenticity2 (extension) (Dépendant du FSCP)	
	6	
	7	
	8 FSP_Port	UIntegerT (8)
	9 FSP_AuthentCRC	UIntegerT (16)
	10	
	11 FSP_ProtVersion	UIntegerT (8)
	12 FSP_ProtMode	UIntegerT (8)
	13 FSP_WatchdogTime	UIntegerT (16)
	14	
	15 FSP_IO_StructCRC	UIntegerT (16)
	16	
	17 FSP_TechParCRC	UIntegerT (32)
	18	
	19	
	20	
	21 FSP_ProtParCRC	UIntegerT (16)
	22	
Fin du FSP_VerifyRecord		

IEC

Figure 56 – Structure de FSP_VerifyRecord

Les paramètres d'authenticité sont sécurisés par FSP_AuthentCRC pour la transmission de l'outil de Maître FS vers le Maître FS, puis vers l'Appareil FS. La procédure d'acquisition des paramètres d'authenticité FSCP depuis la passerelle FSCP et la gestion ultérieure de l'enregistrement d'authenticité FSP est décrite en 10.4.3.3. FSP_ProtParCRC sécurise les paramètres de protocole, comme cela est décrit en 10.4.3.4.

11.7.5 Authentification

Le SLM du Maître FS utilise le FSP_VerifyRecord reçu du gestionnaire de configuration. Ainsi, les codes FSP_Authenticity de l'enregistrement peuvent être comparés aux valeurs d'authenticité FSCP réelles de la partie relative à la sécurité de la passerelle.

11.7.6 Intégrité du stockage

Les deux enregistrements (authenticité et protocole) de la Figure 56 sont stockés à la fois dans le Maître FS et l'Appareil FS, et peuvent échouer au fil du temps (voir aussi Tableau A.1).

A chaque démarrage normal, le gestionnaire de configuration transfère le FSP_VerifyRecord vers l'Appareil FS pendant PREOPERATE, comme cela est indiqué à la Figure 57 et décrit en 10.4.3.1 et en A.2.10. La Figure 57 est issue du gestionnaire de messages du Maître de l'IEC 61131-9 et fournit une image abstraite et simplifiée des états et transitions essentiels.

L'Appareil FS détecte une non-concordance potentielle entre le jeu de paramètres d'authenticité téléchargé et les valeurs déjà stockées dans l'Appareil FS, si un Appareil FS est connecté de manière erronée à un port différent ou même à un Maître FS différent, par exemple (voir Figure 36).

Les paramètres de protocole sont propagés à la couche de communication de sécurité à chaque démarrage.

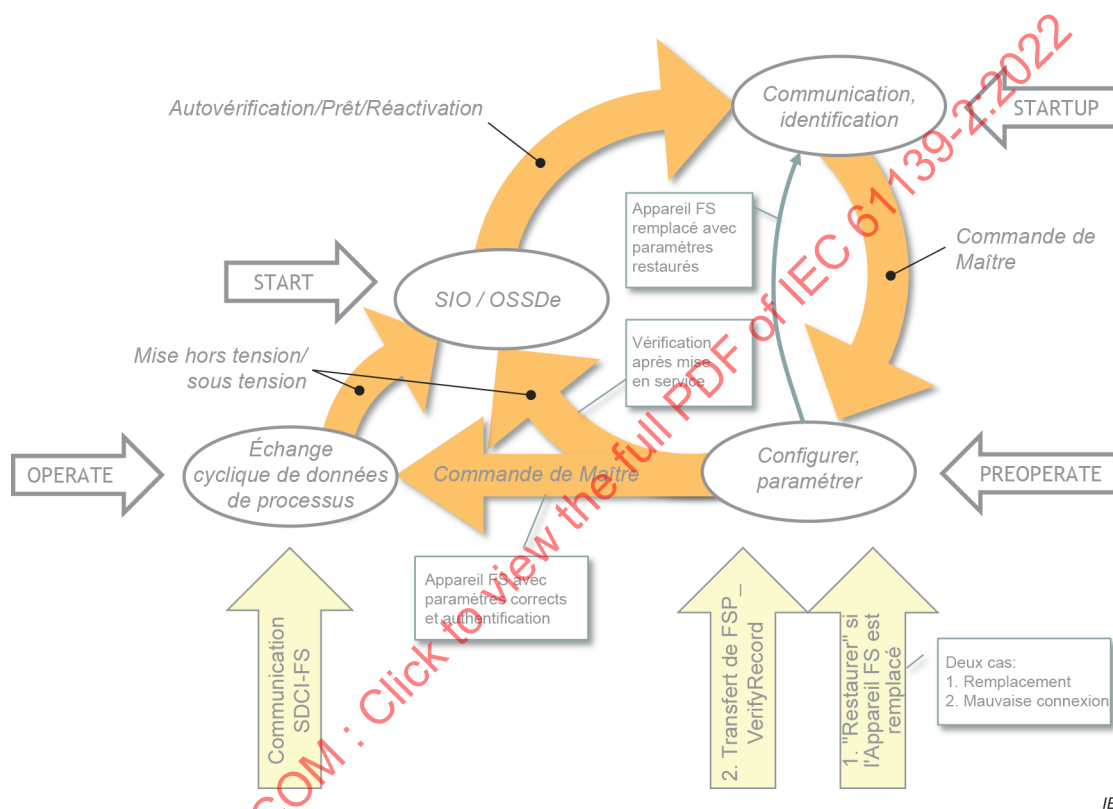


Figure 57 – Démarrage de la SDCI-FS

Dans le cas où l'Appareil FS a été remplacé en raison d'une défaillance, les paramètres spécifiques à la technologie (FST) et les paramètres FSP sont "restaurés" à partir du stockage de données, si l'Appareil FS contient tous les paramètres d'authenticité = "0". Si les paramètres d'authenticité ne sont pas à "0", l'Appareil FS doit les ignorer et conserver les éléments existants (voir 9.4, E.5.7, et étape 1 de la Figure 57). Dans ce cas, une erreur de connexion peut être admise par hypothèse, ou l'Appareil FS a déjà été utilisé et exige des essais et une réinitialisation des paramètres d'authenticité (voir Annex G).

11.7.7 Intégrité de la structure de données d'E/S FS

Il convient de mapper l'ensemble des données d'E/S des Appareils FS de manière efficace vers les données d'E/S FSCP, comme cela est indiqué en 12.1.

En raison des bits qualificatifs supplémentaires exigés pour la passivation sélective du port, la structure de données spécifique à l'Appareil FS d'origine n'est pas directement visible dans la structure de données d'E/S FSCP échangée avec l'hôte FSCP.

L'interpréteur relatif à la sécurité de l'outil de Maître FS transfère l'ensemble des données d'instance ainsi que la signature CRC au mappeur de données d'E/S FS, comme cela est indiqué en 10.4.3.1 (voir aussi A.2.7).

11.7.8 Paramètres de technologie (FST) fondés sur l'IODD

L'un des objectifs de la SDCI-FS est de remplacer l'Appareil FS sans outils, en utilisant le mécanisme de stockage de données d'origine de la SDCI. A titre de condition préalable, la description des paramètres FST est exigée dans l'IODD (voir E.5.7).

Dans ce cas, les paramètres FST s'affichent dans l'outil de Maître FS (voir Figure 58, partie sur les paramètres FST). Les valeurs peuvent être affectées de la même manière que les paramètres non relatifs à la sécurité, lors de "l'essai de mise en service" uniquement (voir Tableau G.1).

L'utilisateur est responsable de l'exactitude des valeurs au sein de l'Appareil FS, en utilisant des procédures de validation adéquates. L'outil de Maître FS peut aider, par la lecture et l'affichage des paramètres, par exemple.

La sécurisation des paramètres FST par signature ne doit pas être effectuée par l'outil de Maître FS. Un "outil dédié" distinct (outil d'Appareil) fourni par le fabricant de l'Appareil FS doit être utilisé à la place, comme cela est indiqué à la Figure 59 et expliqué ci-après.

Topologie

- Niveau supérieur
- ... Maître
- Port 1: Appareil aa
- Port 2: Appareil b
- ...
- Port n: Appareil xxx

Identification	Paramètres	Observation	Diagnostic
Paramètres de protocole SDCI-FS (FSP)			
FSP_Watchdog			
FSP_Protocol			
FSP_Portmode			
FSP_Safety-Level			
FSP_TechParCRC	0x3AF2		
Device Tool	THC26		
Paramètres de technologie (FST)			
Filtre	26		
Discordance	5		
Redondance	yes		
Cycle d'essai	3		

Bibliothèque d'Appareils

- Fournisseur 1
 - Appareil a V1.03
 - Appareil b V1.23
 - Appareil c V2.00
 - ...
- Fournisseur 2
 - Appareil aa V0.99
 - Appareil bb V1.1.2
 - ...

Commande à deux mains THC 26

Filtre	26		
Discordance	5		
Redondance	yes		
Cycle d'essai	3		
Signature CRC	0x3AF2		

Confirmer

Étapes de sécurisation :

- ① Menu de l'outil d'Appareil (PID)
- ② Transférer la valeur d'instance par TPF
- ③ Confirmer les valeurs
- ④ Transférer la valeur de signature CRC par presse papiers ou TBF

Figure 58 – Sécurisation des paramètres FST avec un outil dédié

Après essai et validation, l'outil d'Appareil est invoqué par le biais d'un menu (étape ①). Les valeurs d'instance sont transférées par TPF (étape ②) et affichées à nouveau. L'utilisateur compare les valeurs d'instance et confirme leur exactitude avec le bouton "Confirmer" (étape ③). L'outil de l'Appareil calcule ensuite la signature CRC pour les données d'instance des paramètres FST (voir "signature CRC" à la Figure 58), qui peut être copiée et collée, ou transférée par TBF dans le champ "FSP_TechParCRC" des paramètres FSP (étape ④).

Puisque ce paramètre fait partie du bloc de paramètres FSP, l'Appareil FS peut vérifier l'intégrité de ces paramètres FST avec les paramètres de protocole.

11.7.9 Paramètres de technologie (FST) fondés sur un outil dédié existant (IOPD)

Dans les cas où les appareils de sécurité existants ont déjà leur programme PC avec protection par mot de passe, assistants, fonctions d'autoapprentissage, instructions de vérification, surveillance en ligne, diagnostic, accès spécial à l'historique des appareils pour le fabricant, etc., la description des paramètres FST peut ne pas être disponible. La Figure 59 donne un exemple.

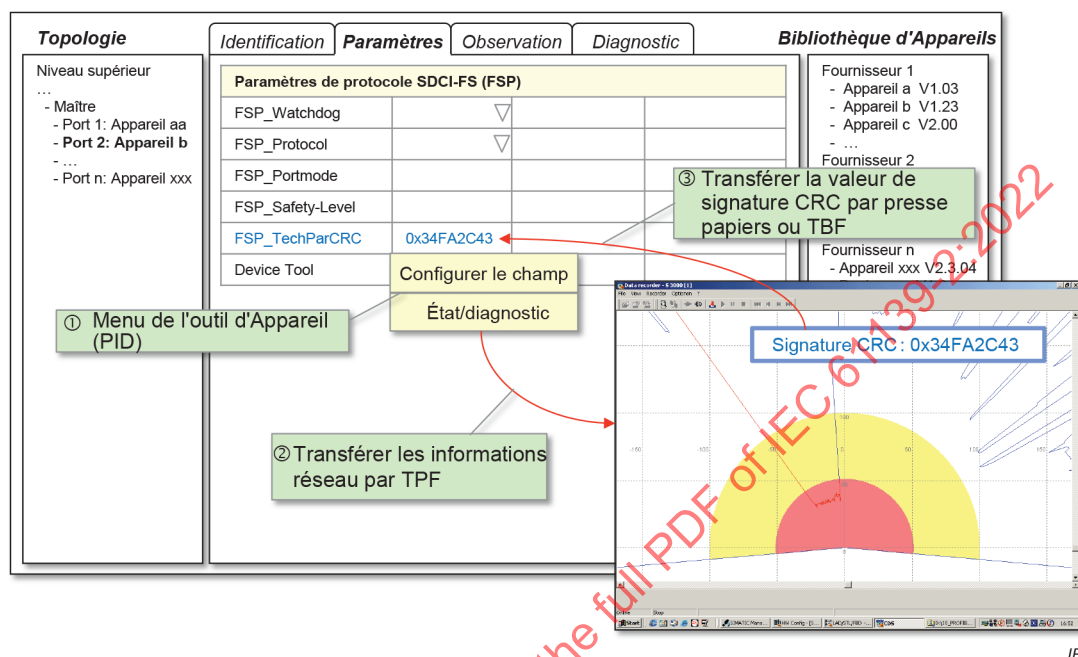


Figure 59 – Modification des paramètres FST avec un outil d'Appareil

Un outil d'Appareil exige une communication avec son Appareil FS particulier et, par conséquent, l'accès à un serveur de communication (voir Article F.5). Il peut être invoqué par des entrées de menu (étape ①) et ainsi accéder directement à des fonctions de configuration ou d'état/de diagnostic, par exemple. Les informations réseau sont transférées par TPF (étape ②). Après essai et validation, il doit fournir un affichage de la signature CRC calculée pour les données d'instance, qui peut être copiée et collée dans le champ "FSP_TechParCRC" des paramètres FSP (étape ③).

Ces Appareils FS doivent être pris en charge par le mécanisme de stockage de données de la SDCI et le remplacement d'un Appareil FS sans outils est possible.

La limite de stockage de données par Appareil FS est de 2 048 octets selon l'IEC 61131-9.

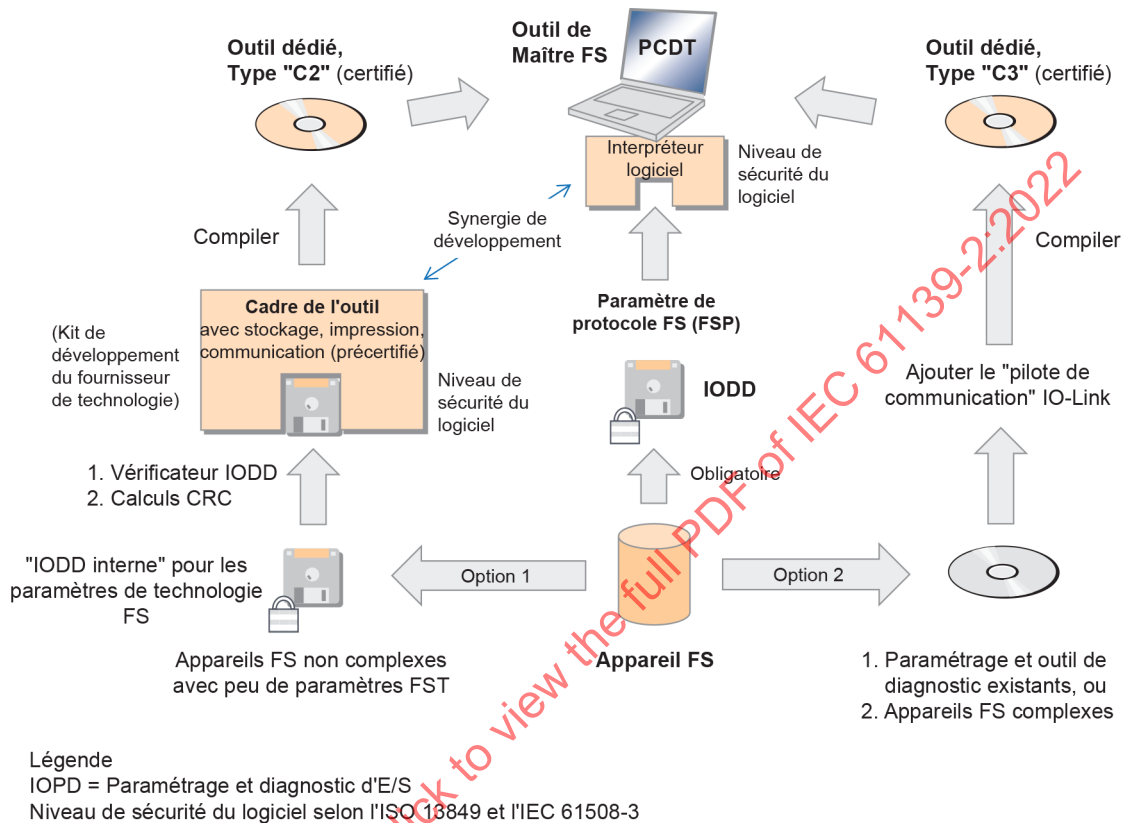
11.8 Création des paramètres FSP et FST

Les normes de "sécurité des machines" (ISO 13849-1 et IEC 62061, par exemple) exigent des "outils dédiés" pour le paramétrage des appareils de sécurité. Pour simplifier le développement et la logistique des outils logiciels, il est recommandé d'utiliser le processus décrit à la Figure 60.

Pour les Appareils FS avec quelques paramètres FST seulement et sans logique applicative, ni assistant ni système d'aide, un "cadre d'outil dédié" particulier peut être développé de manière sécurisée selon l'IEC 61508-3 et équipé des interfaces de communication nécessaires. Le fournisseur de technologie peut fournir un tel cadre pour les paramètres FST d'un Appareil FS particulier (option 1 de la Figure 60). Les développeurs d'Appareils FS peuvent personnaliser le cadre en utilisant le nom de la marque, le nom de l'entreprise et les identifiants

de l'Appareil FS pour un outil dédié unique (IOPD). Le logiciel d'un outil dédié exécutable peut être certifié par des organismes d'évaluation.

Pour les Appareils FS avec des paramètres FST plus complexes, l'IOPD peut être développé sur une base individuelle, ou les outils existants peuvent être utilisés. Dans les deux cas, les outils peuvent être équipés des interfaces de communication nécessaires (option 2 de la Figure 60).



IEC

Figure 60 – Création des paramètres FSP et FST

Dans tous les cas, l'outil dédié (IOPD) doit calculer et afficher la signature CRC pour tous les paramètres FST. La signature peut être copiée dans le champ d'entrée du paramètre FSP "FSP_TechParCRC", de sorte qu'un Appareil FS peut vérifier l'exactitude des paramètres FST stockés en local après le démarrage et le téléchargement des paramètres FSP configurés sur l'Appareil FS.

Pour chaque Appareil FS, le même jeu de paramètres FSP (protocole) doit être créé dans une IODD étendue pour SDCI-FS. L'IODD est obligatoire et contient les paramètres conventionnels habituels ainsi que les paramètres FSP.

11.9 Intégration des outils dédiés (IOPD)

11.9.1 Interface IOPD

En général, un outil de Maître (PCDT) assure un soutien technique pour un Maître et ses Appareils au moyen de descriptions d'Appareils sous la forme de fichiers XML (IODD). En principe, le concept est le même pour un Maître FS et un Appareil FS. Pour la sécurité fonctionnelle, en plus d'une IODD étendue, il est nécessaire qu'un fournisseur d'Appareils FS fournisse un outil dédié supplémentaire (IOPD), comme cela est indiqué à la Figure 60.

Pour que l'IOPD communique avec l'Appareil FS correspondant, une nouvelle interface de communication normalisée est exigée.

11.9.2 Interfaces normalisées

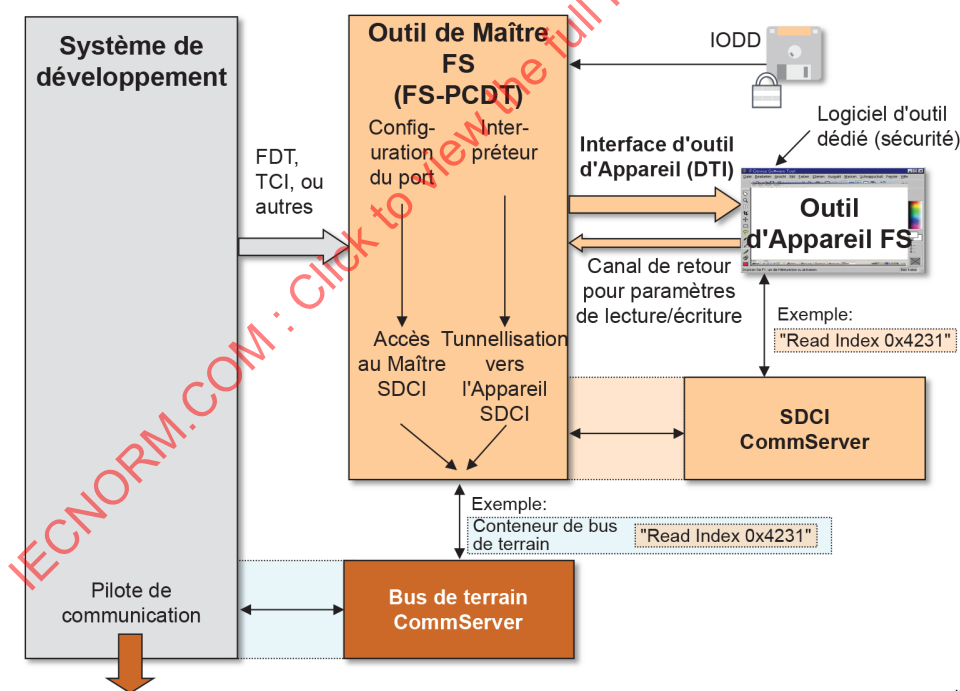
En général, les outils de Maître sont intégrés en utilisant des normes existantes telles que le FDT, la FDI (voir [11]) ou des solutions propriétaires. Une telle diversité n'est pas acceptable pour les Appareils FS et, par conséquent, une technologie simple et éprouvée a été choisie et adoptée pour la SDCI-FS. Il s'agit de l'"interface d'outil d'Appareil" (DTI).

L'Annex F fournit la spécification de cette interface.

La Figure 61 représente la hiérarchie de communication du FDT et des autres normes pour les bus de terrain, ainsi que la connexion par le biais de l'"interface d'outil d'Appareil" et la communication SDCI sous-jacente.

L'outil d'Appareil FS (IOPD) n'a pas à connaître l'environnement du bus de terrain auquel il est connecté. L'exemple de la Figure 61 représente comment il envoie un service "Read Index 0x4231" et comment l'outil de Maître FS compresse ce service dans un conteneur de bus de terrain et le transmet au serveur de communication du bus de terrain.

Le Maître FS visé par l'adressage est connecté au bus de terrain et reçoit ce conteneur. Il décompresse le service de lecture SDCI et l'exécute avec l'Appareil FS visé par l'adressage connecté à un port.



IEC

Figure 61 – Exemple de hiérarchie de communication

11.9.3 Canal de retour

Un fournisseur de Maître FS ne sait pas à l'avance quels Appareils FS le client souhaite connecter aux ports du Maître FS. Par conséquent, la description d'appareil du bus de terrain du Maître FS ne peut fournir que des "conteneurs" prédéfinis pour la structure de données d'E/S obtenue des ensembles d'Appareils FS connectés aux ports. En matière de sécurité fonctionnelle, cela est encore plus compliqué, puisque la description des structures de données doit être codée et sécurisée.

En général, en raison de la diversité de configurations et de paramétrages différents d'un Appareil FS particulier, celui-ci:

- exige par exemple différentes structures de données d'E/S pour échanger avec les AP ou les hôtes;
- présente par exemple des temps de réaction différents en raison des résolutions configurées (élevées ou faibles); et
- peut par exemple avoir différentes valeurs de SIL, PL, catégorie ou PFH qui influent sur le niveau de sécurité global d'une fonction de sécurité.

La "description d'appareil de bus de terrain" classique pour informer un système de développement n'est pas flexible à cet égard. Son avantage réside dans l'aptitude à l'essai et la certification de son interopérabilité avec des outils de développement.

Néanmoins, un "canal de retour" au sein des interfaces d'outil permet aujourd'hui une fabrication flexible et une simplicité de développement et de mise en service. Un exemple est spécifié en 4.15.5 de [12].

Les F.3.5 et F.9.4 spécifient les fonctionnalités de ce "canal de retour".

11.10 Validation

Il incombe au concepteur de l'Appareil FS de spécifier les étapes de vérification et de validation nécessaires (essais, par exemple; voir H.6) dans le manuel de sécurité/d'utilisation et/ou dans l'"outil dédié" (IOPD).

11.11 Passivation

11.11.1 Motifs et moyens

La Figure 62 représente les motifs de la passivation sélective du port. Habituellement, pour des raisons d'efficacité, les signaux 0 à 7 des Appareils FS connectés à des ports ne sont pas mappés individuellement vers une SPDU FSCP, mais plutôt encapsulés dans une SPDU FSCP. Chacun de ces signaux peut être affecté à une fonction de sécurité distincte n à $n+7$. Si une anomalie survient dans l'un des canaux de signaux, une passivation collective pour toute la SPDU FSCP est nécessaire, ce qui force le déclenchement de toutes les fonctions de sécurité.

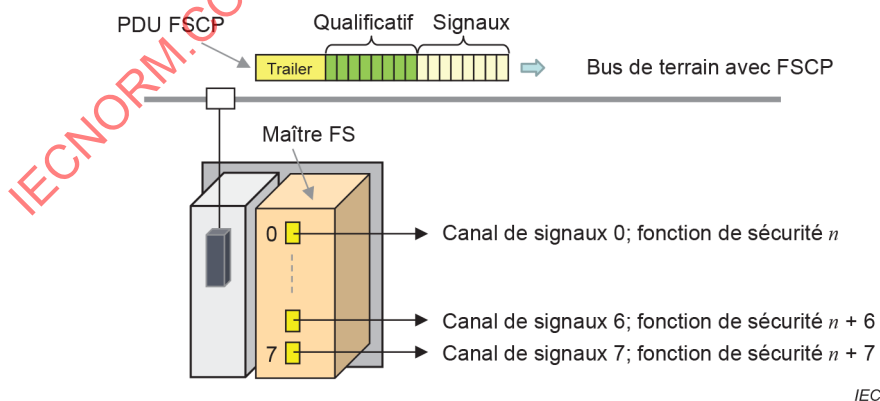


Figure 62 – Motifs de la passivation sélective du port

En général, les FSCP fournissent des bits dits qualificatifs associés aux données de processus de signalisation, qui permettent la passivation sélective de ce canal de signaux particulier et de la fonction de sécurité associée.

En général, la sécurité des machines exige un acquittement de l'opérateur après la réparation d'un canal de signaux défectueux, afin d'empêcher le redémarrage automatique de la machine.

Il n'est pas nécessaire de fournir l'acquittement pour chaque canal de signaux; il peut s'agir d'un acquittement pour tous les canaux.

11.11.2 Passivation sélective du port (Maître FS)

Le 11.11.1 décrit un cas d'utilisation où le canal de signaux correspond directement à un Appareil FS particulier. Le mécanisme de qualification et d'acquittement doit être mis en œuvre au sein du Maître FS, conformément aux spécifications du FSCP particulier.

Il peut être utile pour l'utilisateur d'indiquer sur chaque Appareil FS qu'un acquittement de l'opérateur est exigé avant le redémarrage d'une fonction de sécurité. CB0 (ChAckReq) dans l'octet Control&MCnt (voir Tableau 31) doit être utilisé à cette fin. Il n'est pas relatif à la sécurité.

En option, en cas de FS_PortMode "OSSDe" (voir 10.4.2), le signal ChAckReq peut être connecté séparément à l'indication correspondante de l'Appareil FS (voir H.1).

11.11.3 Passivation sélective du signal (borne FS)

La Figure 13 représente le cas d'utilisation d'une borne FS où un Appareil FS fournit plusieurs canaux de signaux aux appareils de commutation, tels que les boutons d'arrêt d'urgence.

Pour ces Appareils FS, les règles de conception données en 11.4.9.3 s'appliquent. Les mécanismes d'acquittement doivent être mis en œuvre dans les données de processus de sécurité.

11.11.4 Réglages des qualificatifs en cas de communication

La Figure 63 représente l'intégration du gestionnaire de qualificatifs en cas de FS_PortModes "SafetyCom" et "MixedSafetyCom" (voir 10.4.2). Les services/signaux "FAULT_S", "SDset_S", "ChAckReq_S" et "ChAck_C" sont spécifiés en 11.3.2 et en 11.5.2.

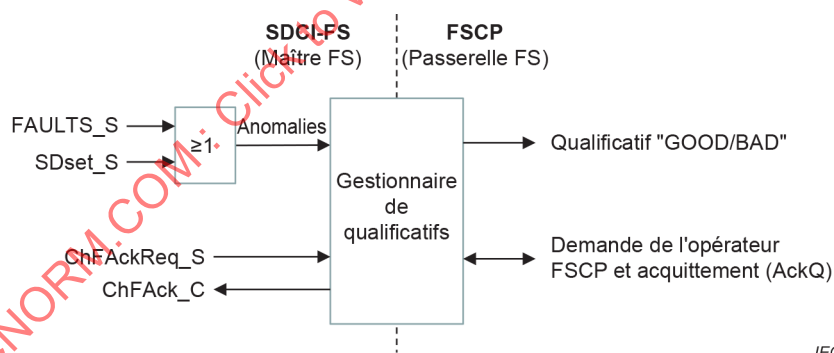


Figure 63 – Gestionnaire de qualificatifs (communication)

Les bits qualificatifs "GOOD/BAD" doivent être définis conformément aux définitions du Tableau 42 lors de la procédure de mapping FSCP.

Tableau 42 – Bits qualificatifs "GOOD/BAD"

Anomalies	Qualificatif	Etat du signal
0	GOOD	1
1	BAD	0

11.11.5 Gestion des qualificatifs en cas d'OSSDe

La Figure 64 représente l'intégration du gestionnaire de qualificatifs en cas de FS_PortModes "OSSDe" (voir 10.4.2). Les définitions du Tableau 42 s'appliquent.

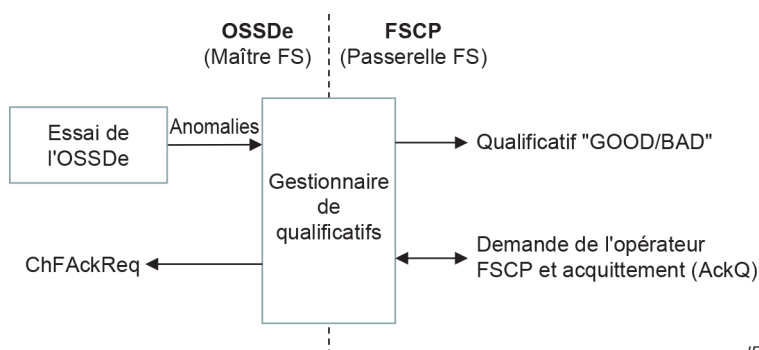


Figure 64 – Gestionnaire de qualificatifs (OSSDe)

La Figure 65 représente le diagramme d'états pour le comportement du gestionnaire de qualificatifs (OSSDe).

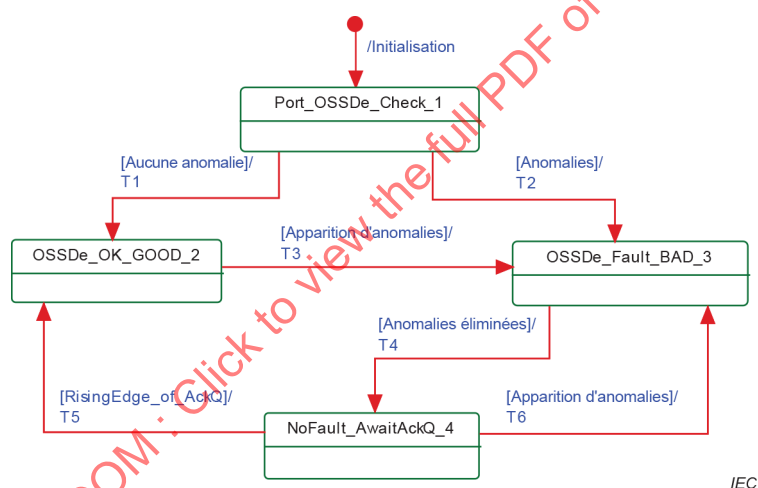


Figure 65 – Comportement des qualificatifs par port du Maître FS

Le Tableau 43 représente les tableaux de transition d'états pour le comportement des qualificatifs.

Tableau 43 – Tableaux de transition d'états pour le comportement des qualificatifs

NOM D'ÉTAT	DESCRIPTION DE L'ÉTAT
Initialisation	Utiliser SD, Qualifier = BAD, ChFAckReq =0
1 Port_OSSDe_Check	Effectuer un diagnostic du port pour détecter les anomalies
2 OSSDe_OK_GOOD	Effectuer un diagnostic cyclique du port pour détecter les anomalies
3 OSSDe_Fault_BAD	Effectuer un diagnostic cyclique du port pour détecter les anomalies
4 NoFault_AwaitAckQ	Attendre le front montant du signal AckQ

TRANSITION	ÉTAT SOURCE	ÉTAT CIBLE	ACTION
T1	1	2	Utiliser PD, Qualifier = GOOD, AckQ = 0, ChFAckReq =0
T2	1	3	Utiliser SD, Qualifier = BAD, AckQ = 0, ChFAckReq =0
T3	2	3	Utiliser SD, Qualifier = BAD, AckQ = 0, ChFAckReq =0
T4	3	4	Utiliser SD, Qualifier = BAD, AckQ = 0, ChFAckReq =1
T5	4	2	Utiliser PD, Qualifier = GOOD, AckQ = 1, ChFAckReq =0
T6	4	3	Utiliser SD, Qualifier = BAD, AckQ = 0, ChFAckReq =0

ÉLÉMENTS INTERNES	TYPE	DÉFINITION
RisingEdge_of_AckQ	Drapeau	Moyen d'empêcher l'activation permanente du signal AckQ.
AckQ	Drapeau	Drapeau qui dépend du système FSCP de niveau supérieur individuel et de son mapping.
Faults	Drapeau	Toute anomalie détectée, comme une rupture de câble ou un court-circuit.
ChFAckReq	Drapeau	Signal défini par le gestionnaire de qualificatifs (voir 11.11.2 et H.1)

11.12 Diagnostic SCL

La couche de communication de sécurité peut créer ses propres EventCodes, comme erreur CRC, erreur de compteur ou expiration, comme cela est indiqué en B.1.

12 Traitement de sécurité fonctionnelle (FS-P)

12.1 Recommandations pour des mappings d'E/S efficaces

La Figure 66 montre comment l'efficacité peut être améliorée par l'encapsulation des données d'E/S des appareils de sécurité connectés dans une seule SPDU FSCP au lieu de plusieurs SPDU FSCP distinctes. Sur la gauche, les bits des appareils de sécurité (OSSD) sont encapsulés dans une SPDU FSCP par le module FS-DI. Sur la droite, les Appareils FS utilisent chacun une SPDU FSCP par l'intermédiaire du Maître/de la passerelle FS pour transmettre un bit. Au milieu, un Maître/une passerelle FS encapsule plusieurs bits dans une SPDU FSCP de manière similaire à un FS-DI.

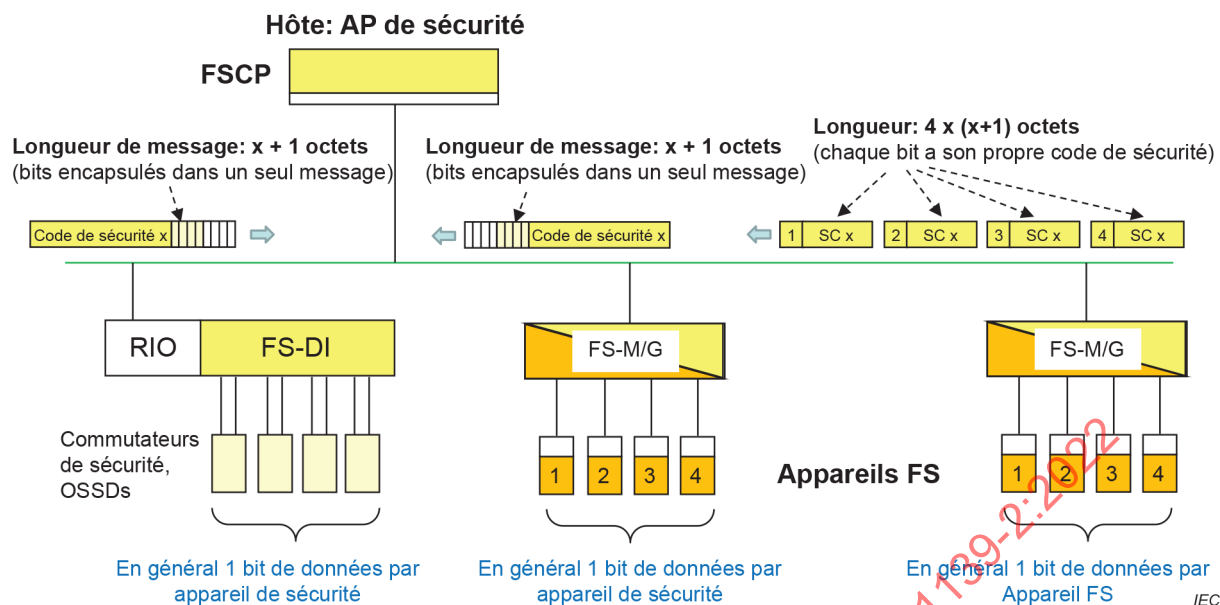


Figure 66 – Problèmes liés à l'efficacité du mapping

La structure de données d'E/S FS doit être créée en tant que multiple de 16 bits.

12.2 Contrôleur FS intégré

La spécification et la mise en œuvre d'un contrôleur FS intégré comme cela est décrit en 4.2 et à la Figure 1 pour assurer des fonctions de sécurité "locales" au niveau de la SDCI-FS relèvent de la responsabilité du fabricant et ne sont pas normalisées.

Annexe A (normative)

Extensions des paramètres

A.1 Index et paramètres pour SDCI-FS

La plage d'index réservée à la SDCI-FS comprend 255 index de 0x4200 à 0x42FF.

Le Tableau A.1 décrit les index spécifiés pour les profils SDCI, les paramètres de protocole (FSP) de la SDCI-FS, qui comprennent l'authenticité, le protocole, les enregistrements de la structure de données d'E/S, et les paramètres auxiliaires, ainsi que la plage totale réservée pour SDCI-FS et la deuxième plage d'index pour les profils SDCI.

Tableau A.1 – Index pour SDCI-FS

Index (déc.)	Sous-index	Nom d'objet	Accès	Longueur	Type de données	M/O/C	Objet/Référence
...							
0x4000 à 0x41FF		Index spécifiques au profil					Par exemple: Capteurs intelligents
Authenticité (11 octets)							
0x4200 (16896)	1	FSP_Authenticity_1	R/W	4 octets	UIntegerT	M	"Code A" du système FSCP de niveau supérieur; voir A.2.1
	2	FSP_Authenticity_2	R/W	4 octets	UIntegerT	M	"Code A" étendu du système FSCP de niveau supérieur
	3	FSP_Port	R/W	1 octet	UIntegerT	M	PortNumber qui identifie l'Appareil FS particulier; voir A.2.2
	4	FSP_AuthentCRC	R/W	2 octets	UIntegerT	M	CRC-16 pour les paramètres d'authenticité; voir A.2.3
Protocole (12 octets)							
0x4201 (16897)	1	FSP_ProtVersion	R/W	1 octet	UIntegerT	M	Version de protocole: 0x01; voir A.2.4
	2	FSP_ProtMode	R/W	1 octet	UIntegerT	M	Modes de protocole, par exemple CRC 16/32 bits; voir A.2.5
	3	FSP_Watchdog	R/W	2 octets	UIntegerT	M	Surveillance de la mise à jour des E/S; 1 à 65 535 ms; voir A.2.6
	4	FSP_IO_StructCRC	R/W	2 octets	UIntegerT	M	Signature CRC-16 pour le bloc de description de la structure d'E/S; voir A.2.7
	5	FSP_TechParCRC	R/W	4 octets	UIntegerT	M	Sécurisation du code pour FST (paramètre spécifique à la technologie); voir A.2.8
	6	FSP_ProtParCRC	R/W	2 octets	UIntegerT	M	CRC-16 pour les paramètres de protocole; voir A.2.9

Index (déc.)	Sous-index	Nom d'objet	Accès	Longueur	Type de données	M/O/C	Objet/Référence
Enregistrement de vérification (23 octets)							
0x4202 (16898)		FSP_VerifyRecord	W	23 octets	RecordT	M	Le Maître FS envoie cet enregistrement de vérification composé de paramètres d'authenticité et de protocole à PREOPERATE. Cet index est masqué à l'utilisateur; voir A.2.10
Paramètres auxiliaires							
0x4210 (16912)		FSP_TimeToReady	R	2 octets	UIntegerT	M	Délai jusqu'à l'impulsion Ready; 1 à 32 767 (en 10 ms), voir A.2.11
0x4211 (16913)		FSP_MinShutDown Time	R	2 octets	UIntegerT	M	Délai minimal d'arrêt de l'Appareil FS après la mise hors tension du port; 100 à 1 000 (en 10 ms), voir A.2.12
0x4212 (16914)		FSP_ParamDescCRC	R	4 octets	UIntegerT	M	Signature CRC-32 qui sécurise l'authenticité, le protocole et la description de la structure d'E/S FS au sein de l'IODD; voir A.2.15
0x4213 (16915)		FSP_WCDT	R	2 octets	UIntegerT	M	Délai le plus défavorable; 1 à 32 767 (en ms), voir A.2.13 et H.6
0x4214 (16916)		FSP_OFDT	R	2 octets	UIntegerT	M	Délai de première anomalie; 1 à 32 767 (en ms), voir A.2.14 et H.6
0x4215 (16917) à 0x42FF (17151)		Réservé à la SDCI-FS					
0x4300 à 0x4FFF		Index spécifiques au profil					Par exemple: Mise à jour du BLOB et du micrologiciel
...							
Légende							
M = obligatoire (mandatory); O = facultatif (optional); C = conditionnel							

A.2 Détail des paramètres

A.2.1 FSP_Authenticity

Lors de la mise en service hors ligne d'un projet SDCI-FS, la valeur par défaut de ce paramètre est "0". Lors de la mise en service en ligne, l'utilisateur acquiert l'authenticité FSCP ("code A") auprès du Maître FS par le biais du service SMI et la propage à l'Appareil FS dans un enregistrement entier, comme cela est décrit en 10.4.3.1. L'outil de Maître FS ne doit transférer que des blocs d'authenticité entiers à l'Appareil FS avec des valeurs de signature CRC correctes, de sorte que l'Appareil FS peut en vérifier la vraisemblance et l'exactitude (voir A.2.3).

Si le système est armé (SP_TechParCRC ≠ "0"), l'Appareil FS compare à chaque démarrage (DS_Change ou PortPowerOffOn) ses valeurs stockées en local avec les valeurs de FSP_VerifyRecord, afin de détecter toute erreur de connexion (port ou Maître FS incorrect); voir Annex G.

Certains FSCP fournissent une authenticité étendue. Dans ces cas, le code étendu doit être inclus dans ce paramètre.

Les bits et les octets de remplissage doivent être remplis par des "0".

A.2.2 FSP_Port

L'outil de Maître FS identifie le PortNumber de Maître FS de l'Appareil FS connecté et le stocke dans ce paramètre. Le stockage et la vérification du paramètre par l'Appareil FS sont décrits en A.2.1 et en A.2.3. La numérotation commence à "1". Ainsi, l'appareil FS ne doit pas accepter un "0".

Le PortNumber par défaut dans l'IODD est "0", ce qui signifie que le PortNumber d'un Appareil particulier n'a pas encore été affecté.

A.2.3 FSP_AuthentCRC

L'outil de Maître FS ne doit transférer que des blocs d'authenticité entiers à l'Appareil FS, y compris FSP_Authenticity et FSP_Port (voir Tableau A.1).

Pour calculer la signature CRC du bloc d'authenticité entier, le CRC-16 du Tableau D.1 doit être utilisé. Ce polynôme CRC a une distance de Hamming ≥ 6 pour des longueurs ≤ 16 octets. La valeur de départ "0" doit être utilisée (voir D.3.6).

A.2.4 FSP_ProtVersion

Le Tableau A.2 décrit le codage de FSP_ProtVersion.

Tableau A.2 – Codage de la version de protocole

Valeur	Définition
0x00	Non admis
0x01	La présente version de protocole
0x02 à 0xFF	Réservé

A.2.5 FSP_ProtMode

Le Tableau A.3 décrit le codage de FSP_ProtMode. Les appareils d'essai utilisent des "miroirs d'essai".

Tableau A.3 – Codage du mode de protocole

Valeur	Définition
0x00	Non admis
0x01	0 à 3 octets de données de processus d'E/S FS; CRC-16
0x02	0 à 25 octets de données de processus d'E/S FS; CRC-32
0x03 à 0xF8	Réservé
0xF9	Miroir d'essai en cas de CRC-32 (réservé à l'essai)
0xFA	Miroir d'essai en cas de CRC-16 (réservé à l'essai)
0xFB à 0xFF	Réservé

A.2.6 FSP_Watchdog

Le concepteur de l'Appareil FS détermine le temps de mise à jour des E/S et l'utilise comme valeur par défaut de ce paramètre dans l'IODD. Le temps de mise à jour des E/S correspond à la période de temps entre deux PDU de sécurité avec des valeurs de compteur consécutives (échantillons d'E/S), y compris les répétitions possibles dans la couche de communication SDCI (canal noir; voir 11.5.5).

A l'aide de la valeur par défaut du paramètre (temps de mise à jour des E/S), les temps de transmission des PDU de sécurité et les temps de traitement du Maître FS, l'outil de Maître FS peut estimer le temps total et suggérer la valeur du paramètre "FSP_Watchdog".

La plage de valeurs s'étend de 1 à 65 535 (mesurée en ms). La valeur "0" n'est pas admise. La SCL de l'Appareil FS est chargée de vérifier la validité au démarrage et de créer une erreur, le cas échéant (voir Tableau B.1).

A.2.7 FSP_IO_StructCRC

Une visionneuse non relative à la sécurité fondée sur l'IODD peut être utilisée pour calculer la signature CRC 16 bits pour la description de la structure d'E/S FS au sein de l'IODD pendant la phase de développement. L'algorithme de calcul est représenté à l'Annex D. La valeur de départ "0" doit être utilisée (voir D.3.6).

L'interpréteur relatif à la sécurité de l'outil de Maître FS transfère l'ensemble des données d'instance ainsi que la signature CRC au mappeur de données d'E/S FS, comme cela est indiqué en 10.4.3.1.

Le Tableau A.4 représente la version "1" de la description de la structure de données d'E/S FS générique pour Appareils FS. A l'aide de ce tableau, les instances individuelles des données de processus d'E/S FS de l'Appareil FS peuvent être créées par le biais de l'IODD et, entre autres, utilisées pour un mapping automatique des données SDCI-FS vers les données de sécurité FSCP.

Tableau A.4 – Description de la structure de données d'E/S FS générique

Nom de l'élément	Longueur de l'élément	Définition
IO_DescVersion	1 octet	Version de la description de structure générique: 0x01
InputDataRange	1 octet	Longueur en octets de l'ensemble des données de processus d'entrée FS, qui inclut respectivement 4 ou 6 octets pour le code de sécurité (Control/Status, PortNumber et CRC-16/32)
TotalOfInBits	1 octet	Nombre du jeu complet de BooleanT d'entrée (bits)
TotalOfInOctets	1 octet	Nombre d'octets avec BooleanT d'entrée (y compris les octets non remplis)
TotalOfInInt16	1 octet	Nombre d'IntegerT(16) d'entrée
TotalOfInInt32	1 octet	Nombre d'IntegerT(32) d'entrée
OutputDataRange	1 octet	Longueur en octets de l'ensemble des données de processus de sortie FS, qui inclut respectivement 4 ou 6 octets pour le code de sécurité (Control/Status, PortNumber et CRC-16/32)
TotalOfOutBits	1 octet	Nombre du jeu complet de BooleanT de sortie (bits)
TotalOfOutOctets	1 octet	Nombre d'octets avec BooleanT de sortie (y compris les octets non remplis)
TotalOfOutInt16	1 octet	Nombre d'IntegerT(16) de sortie
TotalOfOutInt32	1 octet	Nombre d'IntegerT(32) de sortie
FSP_IO_StructCRC	2 octets	Valeur de la signature CRC-16 pour tous les éléments (voir D.1)

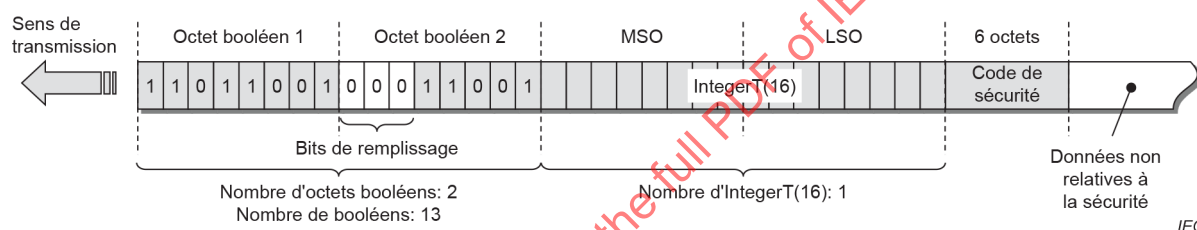
La Figure A.1 représente l'instance de la description des données d'E/S FS de l'exemple de la Figure A.2.

IO_DESCVERSION	01	0x01
INPUT_DATA_RANGE	10	0x0A
TOTAL_OF_INBITS	13	0x0D
TOTAL_OF_INOCTETS	02	0x02
TOTAL_OF_ININT16	01	0x01
TOTAL_OF_ININT32	00	0x00
OUTPUT_DATA_RANGE	06	0x06
TOTAL_OF_OUTBITS	00	0x00
TOTAL_OF_OUTOCTETS	00	0x00
TOTAL_OF_OUTINT16	00	0x00
TOTAL_OF_OUTINT32	00	0x00
FSP_IO_STRUCTCRC	39464	0x9A28

IEC

Figure A.1 – Instance d'une description des données d'E/S FS

La Figure A.2 représente un exemple avec des données de processus d'entrée FS et sans données de processus de sortie FS.



IEC

Figure A.2 – Exemple de structure de données d'E/S FS avec des données non relatives à la sécurité

A.2.8 FSP_TechParCRC

Le présent document spécifie deux méthodes de base pour l'affectation des paramètres spécifiques à la technologie (FST). Le concepteur de l'Appareil FS est chargé de choisir la méthode de sécurisation.

La méthode utilisée en 11.7.8 est fondée sur l'IODD et suggère d'utiliser l'un des polynômes générateurs de CRC du Tableau D.1. Si le calcul de la signature CRC donne la valeur "0", un "1" doit être utilisé.

La méthode donnée en 11.7.9 repose sur un outil d'Appareil FS existant (outil dédié). Quelle que soit la méthode utilisée, l'outil doit afficher un code de sécurisation après vérification et validation, qui peut être copié et collé dans le champ d'entrée du paramètre FSP_TechParCRC.

Lors de la mise en service, une valeur de "0" peut être entrée pour permettre un comportement donné au démarrage de l'Appareil FS (voir 10.4.3.1). Pendant la production, cette valeur doit être ≠ "0".

Pour les transferts de blocs de paramètres spécifiques à la technologie > à 232 octets, le service SMI_PortCmd CMD = "0" (DeviceParBatch) spécifié dans l'IEC 61131-9 peut être utilisé.

A.2.9 FSP_ProtParCRC

L'outil de Maître FS ne doit transférer que des blocs de protocole entiers à l'appareil FS, y compris l'ensemble des paramètres de protocole (voir Tableau A.1). Pour calculer la

signature CRC du bloc de protocole entier, le CRC-16 du Tableau D.1 doit être utilisé. Ce polynôme CRC a une distance de Hamming ≥ 6 pour des longueurs ≤ 16 octets. La valeur de départ "0" doit être utilisée (voir D.3.6).

A.2.10 FSP_VerifyRecord

Un enregistrement composé des paramètres d'authenticité et de protocole est transféré par le biais du service "SMI_PortConfiguration" (voir 10.2.1 et 10.3.3) et stocké dans le gestionnaire de configuration d'un Maître FS. Au démarrage, pendant PREOPERATE, le Maître FS transfère cet enregistrement de vérification en écriture seulement vers un index "masqué" dans l'Appareil FS (voir 11.7.4). L'Appareil FS utilise cet enregistrement géré de manière hétéroclite pour vérifier l'authenticité, le protocole, la structure d'E/S et les paramètres de technologie. Ceci a lieu pendant PREOPERATE, après "DS_Change" (voir Figure 35 et IEC 61131-9), chaque fois qu'un Appareil FS a été remplacé et que les paramètres ont été restaurés par le biais de mécanismes de stockage de données. Cela se produit également après la mise hors tension/sous tension du port pendant la mise en service par SMI_PortPowerOffOn (voir IEC 61131-9:—, 11.2.14 et E.9).

L'enregistrement doit être transféré en tant qu'entité. L'accès au sous-index n'est pas admis. L'index 0x4202 (16898) doit être "masqué" à l'utilisateur, c'est-à-dire qu'il ne doit pas être décrit dans l'IODD.

A.2.11 FSP_TimeToReady

Le concepteur de l'Appareil FS mesure/détermine le laps de temps entre la mise sous tension et l'apparition de l'impulsion Ready (voir 5.3.3 et t_{2R} dans le Tableau 7) et affecte la valeur au FSP_TimeToReady dans l'IODD de l'Appareil FS (voir E.5.8).

NOTE La valeur est liée au paramètre de "retard à la disponibilité" de [9] ou [13].

Une valeur supérieure à 5 s se traduit par un comportement restreint du Maître FS (pas de détection automatique d'Appareil, par exemple).

A.2.12 FSP_MinShutDownTime

Le concepteur de l'Appareil FS mesure/détermine le délai minimal d'arrêt exigé après une mise hors tension du port et avant un redémarrage, et affecte la valeur FSP_MinShutDownTime dans l'IODD de l'Appareil FS (voir E.5.8).

A.2.13 FSP_WCDT

Le concepteur de l'Appareil FS mesure/détermine les valeurs du "délai le plus défavorable" définies en H.6. Plusieurs valeurs différentes sont possibles, puisque l'Appareil FS peut être configuré par des paramètres de technologie (par exemple, une résolution élevée peut entraîner une résolution plus longue et plus faible sur des laps de temps plus courts). Lors de la lecture de ce paramètre, l'Appareil FS fournit une valeur qui correspond à son paramétrage actuel.

A.2.14 FSP_OFDT

Le concepteur de l'Appareil FS mesure/détermine les valeurs du "délai de première anomalie" définies en H.6. Plusieurs valeurs différentes sont possibles. Lors de la lecture de ce paramètre, l'Appareil FS fournit une valeur qui correspond à son paramétrage actuel.

A.2.15 FSP_ParamDescCRC

Ce paramètre a pour objet de sécuriser les descriptions applicables des paramètres de sécurité au sein de l'IODD (voir E.5.6) contre la falsification de données, comme cela est indiqué à la Figure A.3.

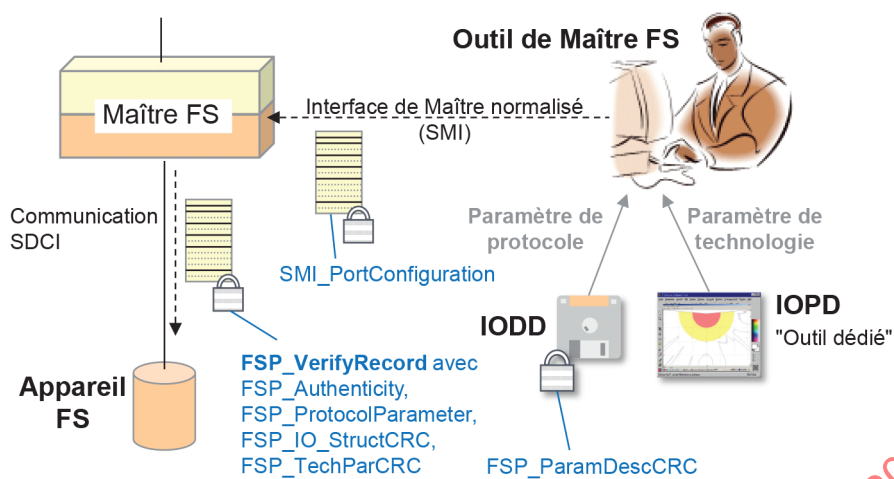


Figure A.3 – Sécurisation des paramètres de sécurité

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

Annexe B (normative)

Extensions des EventCodes

B.1 EventCodes d'Appareil FS supplémentaires

La couche de communication de sécurité (SCL) d'un Appareil FS peut créer ses propres EventCodes, comme cela est indiqué dans le Tableau B.1. Ceux-ci sont transmis par le service SMI_DeviceEvent.

Tableau B.1 – EventCodes spécifiques à la SCL de l'Appareil FS

EventCode	Définition et actions de maintenance recommandées	Valeur d'état de l'Appareil FS	TYPE
0xB000	Erreur de transmission (signature CRC)	2	Notification
0xB001	Erreur de transmission (compteur)	2	Notification
0xB002	Erreur de transmission (temporisation)	3	Erreur
0xB003	Code d'authentification inattendu	3	Erreur
0xB004	Port d'authentification inattendu	3	Erreur
0xB005	FSP_AuthentCRC incorrect	3	Erreur
0xB006	FSP_ProtParCRC incorrect	3	Erreur
0xB007	FSP_TechParCRC incorrect	3	Erreur
0xB008	FSP_IO_StructCRC incorrect	3	Erreur
0xB009	Temps de fonctionnement du chien de garde hors de la spécification ("0", par exemple)	3	Erreur
0xB00A	Aucun FSP_VerifyRecord reçu (déclenché après transition vers OPERATE)	3	Erreur
0xB00B à 0xB0FF	Réservé: ne pas utiliser ces numéros; ne pas évaluer ces numéros	-	-

En général, les erreurs de transmission de "signature CRC" et/ou "compteur" sont provoquées par des messages SDCI avec SPDU gravement falsifiés à la suite d'importantes perturbations. Il n'y a rien à réparer et l'acquittement de l'opérateur suffit. Il convient que cet avertissement très peu probable informe l'opérateur et le gestionnaire de production responsable des possibles modifications d'une machine qui nécessite un examen conformément au manuel de sécurité (voir H.6).

B.2 EventCodes de port supplémentaires

La couche de communication de sécurité (SCL) d'un Maître FS peut créer ses propres EventCodes, comme cela est indiqué dans le Tableau B.2. Ceux-ci sont transmis par le service SMI_PortEvent (voir IEC 61131-9).

Tableau B.2 – EventCodes spécifiques à la SCL du Maître FS

EventCode	Définition et actions de maintenance recommandées	Valeur d'état	TYPE
0x2000	Erreur de transmission (signature CRC)	2	Notification
0x2001	Erreur de transmission (compteur)	2	Notification
0x2002	Erreur de transmission (temporisation)	3	Erreur
0x2003	Code d'authentification inattendu	3	Erreur
0x2004	Réservé		
0x2005	FSP_AuthentCRC incorrect	3	Erreur
0x2006	FSP_ProtParCRC incorrect	3	Erreur
0x2007 à 0x2008	Réservé		
0x2009	Temps de fonctionnement du chien de garde hors de la spécification ("0", par exemple)	3	Erreur
0x200A à 0x20EF	Réservé: ne pas utiliser ces numéros; ne pas évaluer ces numéros	-	-
0x20F0	OSSD verrouillé – signaux au-dessus des temps de discordance spécifiés	3	Erreur
0x20F1 à 0x20FF	Réservé: ne pas utiliser ces numéros; ne pas évaluer ces numéros	-	-

IECNORM.COM : Click to view the full PDF of IEC 61139-2:2022

Annexe C (normative)

Extensions des types de données

C.1 Types de données pour SDCI-FS

Le Tableau C.1 décrit les types de données disponibles dans la SDCI-FS pour l'échange cyclique de données de processus pour les fonctions de sécurité (voir 11.4.9.2).

Tableau C.1 – Types de données pour SDCI-FS

Type de données	Codage	Longueur	Voir IEC 61131-9:—	Exemple d'Appareil
BooleanT/bit	BooleanT ("forme encapsulée" à des fins d'efficacité, aucune structure WORD); l'affectation de noms de signal aux bits est possible.	1 bit	F.2.2, Tableau F.22 et Figure F.9	Commutateur de proximité
IntegerT(16)	IntegerT (énuméré ou signé)	2 octets	F.2.4, Tableau F.4, Tableau F.7 et Figure F.3	Champs de protection du lecteur laser
IntegerT(32)	IntegerT (énuméré ou signé)	4 octets	F.2.4, Tableau F.4, Tableau F.6 et Figure F.3	Codeur ou mesure de longueur ($\approx \pm 2$ km, résolution 1 μ m)

C.2 BooleanT (bit)

Un BooleanT représente un type de données qui ne peut avoir que deux valeurs différentes, à savoir TRUE et FALSE. Ce type de données est spécifié dans le Tableau C.2.

Tableau C.2 – BooleanT pour SDCI-FS

Nom du type de données	Plage de valeurs	Résolution	Longueur
BooleanT	TRUE / FALSE	-	1 bit

La SDCI-FS utilise uniquement le format encapsulé par RecordT, comme cela est indiqué dans le Tableau C.3.

Tableau C.3 – Exemple de BooleanT dans un RecordT

Sous-index	Décalage	Eléments de données	Type de données	Nom/Symbole
1	0	TRUE	BooleanT	Proximity_1
2	1	FALSE	BooleanT	Proximity_2
3	2	FALSE	BooleanT	EmergencyStop_1
4	3	TRUE	BooleanT	EmergencyStop_2
5	4	TRUE	BooleanT	EmergencyStop_3