IEC 61784-3-13

Edition 1.0 2010-06

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Industrial communication networks – Profiles – Part 3-13: Functional safety fieldbuses – Additional specifications for CPF 13

Réseaux de communication industriels – Profils – Partie 3-13: Bus de terrain de sécurité fonctionnelle – Spécifications supplémentaires pour CPF 13

IEC 61784-3-13:2010

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

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
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 corrigenda or an amendment might have been published.

Useful links:

IEC publications search - www.iec.ch/searchpub

The advanced search enables you 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It also gives information on projects, replaced and withdrawn publications.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 on-line and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary (IEV) on-line.

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: csc@iec.ch.

A propos de la CEI

La Commission Electrotechnique Internationale (CEI) 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 CEI

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

Liens utiles:

Recherche de publications CEI - www.iec.ch/searchpub

La recherche avancée vous permet de trouver des publications CEI 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.

Just Published CEI - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications de la CEI. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (VEI) en ligne.

Service Clients - webstore.iec.ch/csc

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

IEC 61784-3-13

Edition 1.0 2010-06

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Industrial communication networks – Profiles – Part 3-13: Functional safety fieldbuses – Additional specifications for CPF 13

Réseaux de communication industriels – Profils – Partie 3-13: Bus de terrain de sécurité fonctionnelle – Spécifications supplémentaires pour CPF 13

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

PRICE CODE XH
CODE PRIX

ICS 25.040.40, 35.100.05

ISBN 978-2-88912-945-4

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

INDUSTRIAL COMMUNICATION NETWORKS – PROFILES –

Part 3-13: Functional safety fieldbuses – Additional specifications for CPF 13

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.

International Standard IEC 61784-3-13 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial process measurement, control and automation.

This bilingual version (2012-02) corresponds to the monolingual English version, published in 2010-06.

The text of this standard is based on the following documents:

FDIS	Report on voting
65C/591A/FDIS	65C/603/RVD

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

The French version of this standard has not been voted upon.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts of the IEC 61784-3 series, published under the general title *Industrial communication networks – Profiles – Functional safety fieldbuses*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

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

IMPORTANT – The 'colour inside' logo on the cover page of this publication 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.

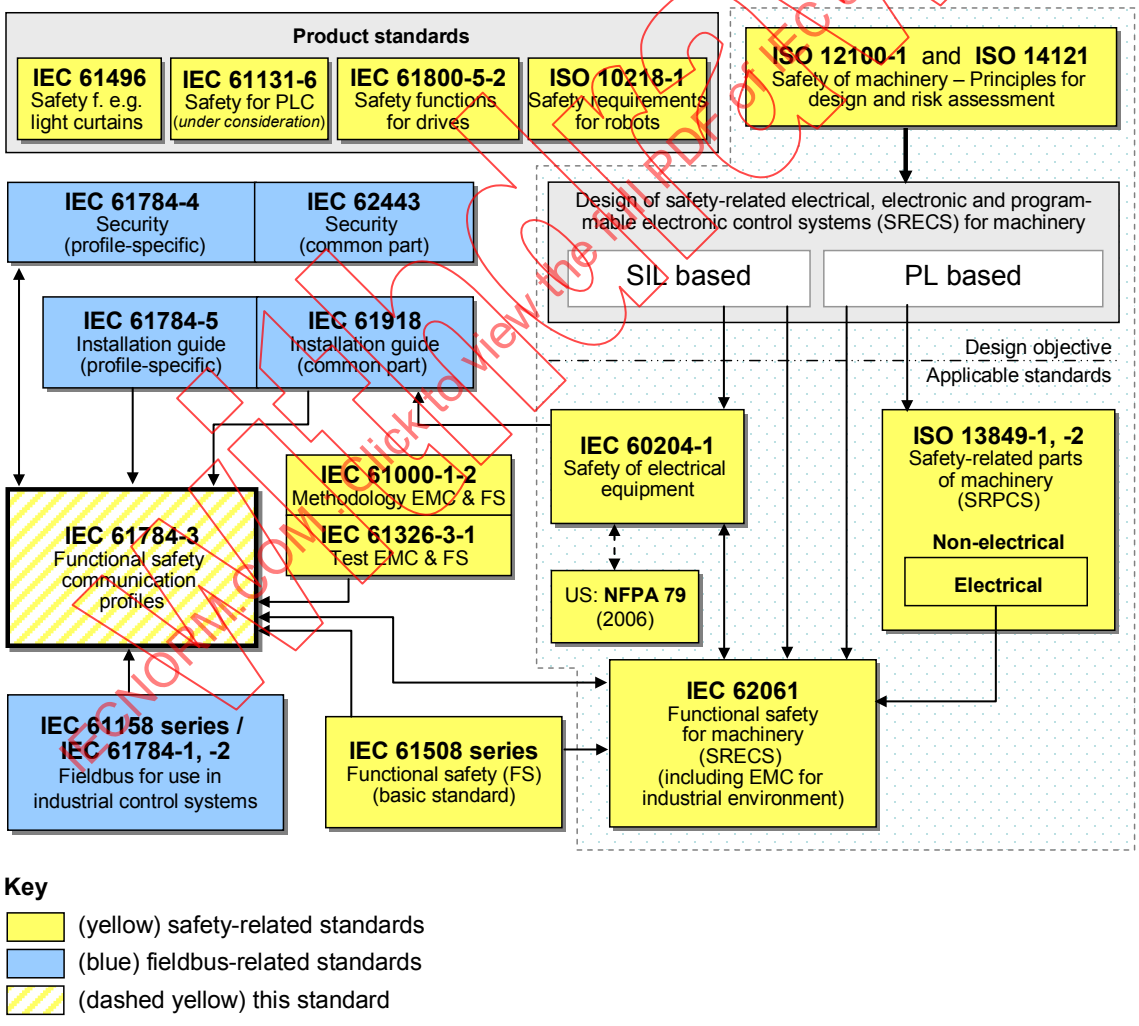
0 Introduction

0.1 General

The IEC 61158 fieldbus standard together with its companion standards IEC 61784-1 and IEC 61784-2 defines a set of communication protocols that enable distributed control of automation applications. Fieldbus technology is now considered well accepted and well proven. Thus many fieldbus enhancements are emerging, addressing not yet standardized areas such as real time, safety-related and security-related applications.

This standard explains the relevant principles for functional safety communications with reference to IEC 61508 series and specifies several safety communication layers (profiles and corresponding protocols) based on the communication profiles and protocol layers of IEC 61784-1, IEC 61784-2 and the IEC 61158 series. It does not cover electrical safety and intrinsic safety aspects.

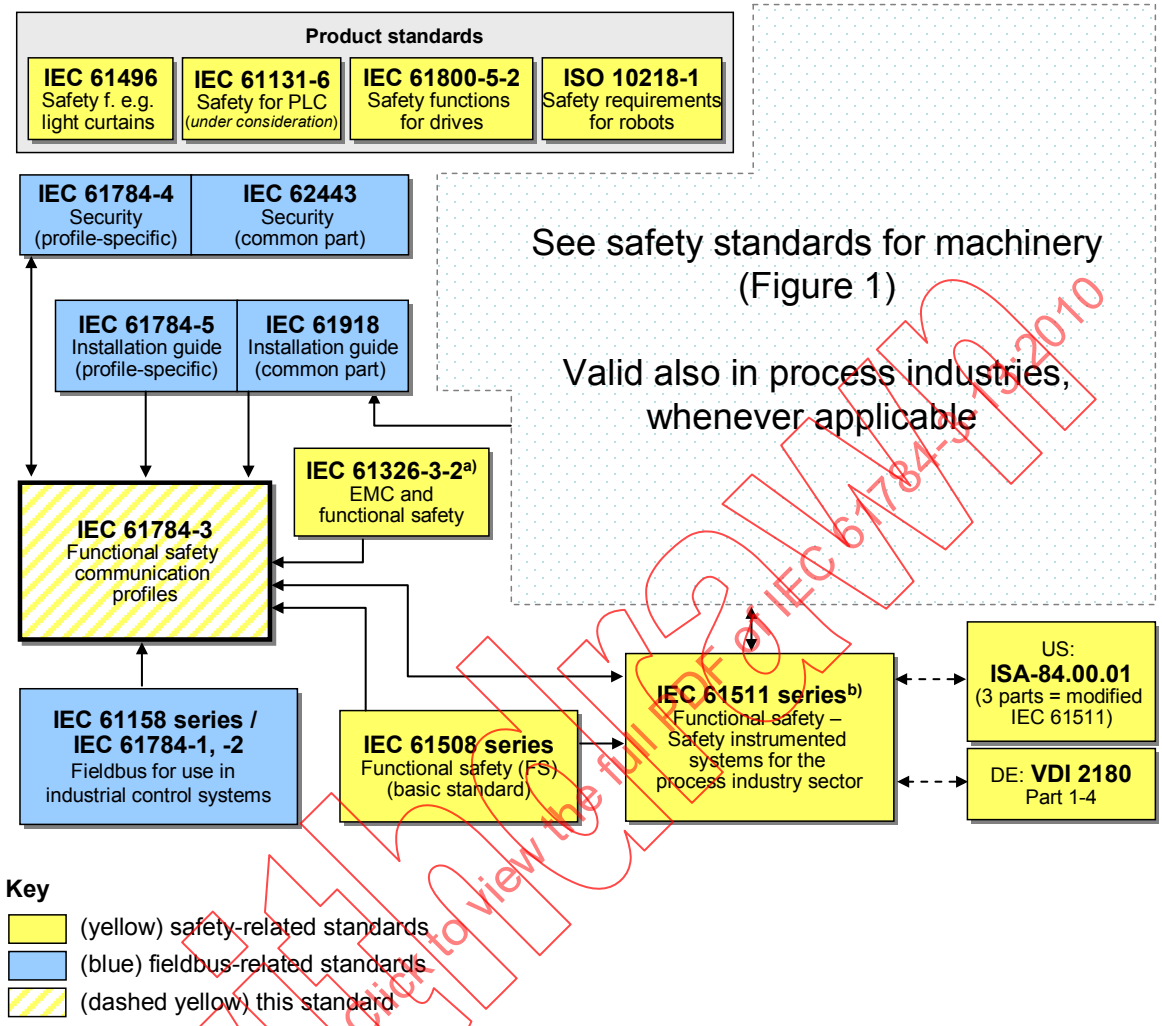
Figure 1 shows the relationships between this standard and relevant safety and fieldbus standards in a machinery environment.



NOTE Subclauses 6.7.6.4 (high complexity) and 6.7.8.1.6 (low complexity) of IEC 62061 specify the relationship between PL (Category) and SIL.

Figure 1 – Relationships of IEC 61784-3 with other standards (machinery)

Figure 2 shows the relationships between this standard and relevant safety and fieldbus standards in a process environment.



[a] For specified electromagnetic environments; otherwise IEC 61326-3-1.

[b] EN ratified.

Figure 2 – Relationships of IEC 61784-3 with other standards (process)

Safety communication layers which are implemented as parts of safety-related systems according to IEC 61508 series provide the necessary confidence in the transportation of messages (information) between two or more participants on a fieldbus in a safety-related system, or sufficient confidence of safe behaviour in the event of fieldbus errors or failures.

Safety communication layers specified in this standard do this in such a way that a fieldbus can be used for applications requiring functional safety up to the Safety Integrity Level (SIL) specified by its corresponding functional safety communication profile.

The resulting SIL claim of a system depends on the implementation of the selected functional safety communication profile within this system – implementation of a functional safety communication profile in a standard device is not sufficient to qualify it as a safety device.

This standard describes:

— basic principles for implementing the requirements of IEC 61508 series for safety-related data communications, including possible transmission faults, remedial measures and considerations affecting data integrity;

— individual description of functional safety profiles for several communication profile families in IEC 61784-1 and IEC 61784-2;

— safety layer extensions to the communication service and protocols sections of the IEC 61158 series.

0.2 Patent declaration

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents concerning the functional safety communication profiles for family 13 as follows, where the [xx] notation indicates the holder of the patent right:

AT 31/2007	[BR]	Anordnung und ein Verfahren zur sicheren Datenkommunikation über ein nicht sicheres Netzwerk
DE 102004055978.3	[BR]	Verfahren zur Zeitsynchronisation innerhalb eines sicherheitsgerichteten Netzwerkes
DE 102004055685.7	[BR]	Verfahren zur Abgrenzung eines sicheren Netzwerkes
DE 102004055684.9	[BR]	Verfahren zur Absicherung des Datentransfers in einem sicheren Netzwerk mit CRC's variabler Länge
EP 08150038	[BR]	Arrangement and a method for safe data communication via a non-safe network
US 11/970178	[BR]	Arrangement and a method for safe data communication via a non-safe network

IEC takes no position concerning the evidence, validity and scope of these patent rights.

The holders of these patents rights have assured the IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holders of these patent rights are registered with IEC.

Information may be obtained from:

[BR] Bernecker + Rainer
Industrie-Elektronik Ges.m.b.H.
B&R Strassse 1
5142 Eggelsberg
AUSTRIA

Tel.: +43 7748 6586– 0
Fax.: +43 7748 6586 – 26

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

INDUSTRIAL COMMUNICATION NETWORKS – PROFILES –

Part 3-13: Functional safety fieldbuses – Additional specifications for CPF 13

1 Scope

This part of the IEC 61784-3 series specifies a safety communication layer (services and protocol) based on CPF 13 of IEC 61784-2 and IEC 61158 Type 13. It identifies the principles for functional safety communications defined in IEC 61784-3 that are relevant for this safety communication layer.

NOTE 1 It does not cover electrical safety and intrinsic safety aspects. Electrical safety relates to hazards such as electrical shock. Intrinsic safety relates to hazards associated with potentially explosive atmospheres.

This part[1] defines mechanisms for the transmission of safety-relevant messages among participants within a distributed network using fieldbus technology in accordance with the requirements of IEC 61508 series[2] for functional safety. These mechanisms may be used in various industrial applications such as process control, manufacturing automation and machinery.

This part provides guidelines for both developers and assessors of compliant devices and systems.

NOTE 2 The resulting SIL claim of a system depends on the implementation of the selected functional safety communication profile within this system – implementation of a functional safety communication profile according to this part in a standard device is not sufficient to qualify it as a safety device.

2 Normative references

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

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

IEC 61158-3-13, *Industrial communication networks – Fieldbus specifications – Part 3-13: Data-link layer service definition – Type 13 elements*

IEC 61158-4-13, *Industrial communication networks – Fieldbus specifications – Part 4-13: Data-link layer protocol specification – Type 13 elements*

IEC 61158-5-13, *Industrial communication networks – Fieldbus specifications – Part 5-13: Application layer service definition – Type 13 elements*

IEC 61158-6-13, *Industrial communication networks – Fieldbus specifications – Part 6-13: Application layer protocol specification – Type 13 elements*

---

1 In the following pages of this standard, “this part” will be used for “this part of the IEC 61784-3 series”.

2 In the following pages of this standard, “IEC 61508” will be used for “IEC 61508 series”.

IEC 61508 (all parts), *Functional safety of electrical/electronic/programmable electronic safety-related systems*

IEC 61784-2, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3*

IEC 61784-3:2010[3], *Industrial communication networks – Profiles – Part 3: Functional safety fieldbuses – General rules and profile definitions*

IEC 61918, *Industrial communication networks – Installation of communication networks in industrial premises*

ISO/IEC 19501, *Information technology – Open Distributed Processing – Unified Modeling Language (UML) Version 1.4.2*

3 Terms, definitions, symbols, abbreviated terms and conventions

3.1 Terms and definitions

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

3.1.1 Common terms and definitions

3.1.1.1
availability
probability for an automated system that for a given period of time there are no unsatisfactory system conditions such as loss of production

3.1.1.2
black channel
communication channel without available evidence of design or validation according to IEC 61508

3.1.1.3
communication channel
logical connection between two end-points within a *communication system*

3.1.1.4
communication system
arrangement of hardware, software and propagation media to allow the transfer of *messages* (ISO/IEC 7498 application layer) from one application to another

3.1.1.5
connection
logical binding between two application objects within the same or different devices

3.1.1.6
Cyclic Redundancy Check (CRC)
<value> redundant data derived from, and stored or transmitted together with, a block of data in order to detect data corruption

<method> procedure used to calculate the redundant data

NOTE 1 Terms “CRC code” and "CRC signature", and labels such as CRC1, CRC2, may also be used in this standard to refer to the redundant data.

---

3 In preparation.

NOTE 2 See also [37], [38][4].

3.1.1.7
error
discrepancy between a computed, observed or measured value or condition and the true, discrepancy between a computed, observed or measured value or condition and the true, specified or theoretically correct value or condition

[IEC 61508-4:2010[5]], [IEC 61158]

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

NOTE 2 Errors do not necessarily result in a *failure* or a *fault*.

3.1.1.8
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 The definition in IEC 61508-4 is the same, with additional notes.

[IEC 61508-4:2010, modified], [ISO/IEC 2382-14.01.11, modified]

NOTE 2 Failure may be due to an *error* (for example, problem with hardware/software design or message disruption)

3.1.1.9
fault
abnormal condition that may cause a reduction in, or loss of, the capability of a functional unit to perform a required function

NOTE IEV 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.

[IEC 61508-4:2010, modified], [ISO/IEC 2382-14.01.10, modified]

3.1.1.10
fieldbus
communication system based on serial data transfer and used in industrial automation or process control applications

3.1.1.11
fieldbus system
system using a *fieldbus* with connected devices

3.1.1.12
frame
denigrated synonym for DLPDU

3.1.1.13
hash function
(mathematical) function that maps values from a (possibly very) large set of values into a (usually) smaller range of values

NOTE 1 Hash functions can be used to detect data corruption.

NOTE 2 Common hash functions include parity, checksum or CRC.

---

4 Figures in square brackets refer to the bibliography.

5 To be published.

[IEC/TR 62210, modified]

3.1.1.14
hazard
state or set of conditions of a system that, together with other related conditions will inevitably lead to harm to persons, property or environment

3.1.1.15
message
ordered series of octets intended to convey information
[ISO/IEC 2382-16.02.01, modified]

3.1.1.16
performance level (PL)
discrete level used to specify the ability of safety-related parts of control systems to perform a safety function under foreseeable conditions
[ISO 13849-1]

3.1.1.17
redundancy
existence of means, in addition to the means which would be sufficient for a functional unit to perform a required function or for data to represent information

NOTE The definition in IEC 61508-4 is the same, with additional example and notes.

[IEC 61508-4:2010, modified], [ISO/IEC 2382-14.01.12, modified]

3.1.1.18
reliability
probability that an automated system can perform a required function under given conditions for a given time interval (t1,t2)

NOTE 1 It is generally assumed that the automated system is in a state to perform this required function at the beginning of the time interval.

NOTE 2 The term "reliability" is also used to denote the reliability performance quantified by this probability.

NOTE 3 Within the MTBF or MTTF period of time, the probability that an automated system will perform a required function under given conditions is decreasing.

NOTE 4 Reliability differs from availability.

[IEC 62059-11, modified]

3.1.1.19
risk
combination of the probability of occurrence of harm and the severity of that harm

NOTE For more discussion on this concept see Annex A of IEC 61508-5:2010[6].

[IEC 61508-4:2010], [ISO/IEC Guide 51:1999, definition 3.2]

3.1.1.20
safety communication layer (SCL)
communication layer that includes all the necessary measures to ensure safe transmission of data in accordance with the requirements of IEC 61508

6 To be published.

3.1.1.21
safety data
data transmitted across a safety network using a safety protocol

NOTE The Safety Communication Layer does not ensure safety of the data itself, only that the data is transmitted safely.

3.1.1.22
safety device
device designed in accordance with IEC 61508 and which implements the functional safety communication profile

3.1.1.23
safety function
function to be implemented by an E/E/PE safety-related system or other risk reduction measures, that is intended to achieve or maintain a safe state for the EUC, in respect of a specific hazardous event

NOTE The definition in IEC 61508-4 is the same, with an additional example and reference.

[IEC 61508-4:2010, modified]

3.1.1.24
safety function response time
worst case elapsed time following an actuation of a safety sensor connected to a fieldbus, before the corresponding safe state of its safety actuator(s) is achieved in the presence of errors or failures in the safety function channel

NOTE This concept is introduced in IEC 61784-3:2010[7], 5.2.4 and addressed by the functional safety communication profiles defined in this part.

3.1.1.25
safety integrity level (SIL)
discrete level (one out of a possible four), corresponding to a range of safety integrity values, where safety integrity level 4 has the highest level of safety integrity and safety integrity level 1 has the lowest

NOTE 1 The target failure measures (see IEC 61508-4:2010, 3.5.17) for the four safety integrity levels are specified in Tables 2 and 3 of IEC 61508-1:2010[8].

NOTE 2 Safety integrity levels are used for specifying the safety integrity requirements of the safety functions to be allocated to the E/E/PE safety-related systems.

NOTE 3 A safety integrity level (SIL) is not a property of a system, subsystem, element or component. The correct interpretation of the phrase "SILn safety-related system" (where n is 1, 2, 3 or 4) is that the system is potentially capable of supporting safety functions with a safety integrity level up to n.

[IEC 61508-4:2010]

3.1.1.26
safety measure
<this standard> measure to control possible communication *errors* that is designed and implemented in compliance with the requirements of IEC 61508

NOTE 1 In practice, several safety measures are combined to achieve the required safety integrity level.

NOTE 2 Communication *errors* and related safety measures are detailed in IEC 61784-3:2010, 5.3 and 5.4.

---

7 In preparation.

8 To be published.

3.1.1.27
safety-related application
programs designed in accordance with IEC 61508 to meet the SIL requirements of the application

3.1.1.28
safety-related system
system performing *safety functions* according to IEC 61508

3.1.1.29
slave
passive communication entity able to receive messages and send them in response to another communication entity which may be a master or a slave

3.1.1.30
time stamp
time information included in a *message*

3.1.2 CPF 13: Additional terms and definitions

3.1.2.1
bit error rate
likelihood of a bit misinterpretation due to electrical noise

3.1.2.2
Device Vendor Information (DVI)
information for identifying a safety node

3.1.2.3
Safety Address (SADR)
address of a safety node within a safety domain

3.1.2.4
Safety Configuration Manager (SCM)
service which is responsible to manage safety related services

NOTE Examples of managed safety related services are configuration, parameterization.

3.1.2.5
Safety Node (SN)
node with safety function

3.1.2.6
Safety Domain (SD)
address space for up to 1023 safety nodes

3.1.2.7
Safety Domain Gateway (SDG)
gateway to share data between different safety domains

3.1.2.8
Safety Domain Number (SDN)
identification of a safety domain

3.1.2.9
Safety Network Management (SNMT)
services for safety layer management

3.1.2.10
Safety Object Dictionary (SOD)
repository of all data objects accessible using services of the functional safety fieldbus

3.1.2.11
Safety Process Data Object (SPDO)
object for process data exchange between different safety nodes

3.1.2.12
Safety Service Data Object (SSDO)
object for service data exchange between safety nodes

3.1.2.13
initialization
initial safety node communication state

3.1.2.14
operational
safety node communication state where all functional safety functions are working properly

3.1.2.15
pre-operational
safety node communication state where all application outputs are in safety state

3.1.2.16
Unique Device Identification (UDID)
worldwide unique identification of a device

3.2 Symbols and abbreviated terms

3.2.1 Common symbols and abbreviated terms

CP	Communication Profile	[IEC 61784-1]
CPF	Communication Profile Family	[IEC 61784-1]
CRC	Cyclic Redundancy Check	
DLL	Data Link Layer	[ISO/IEC 7498-1]
DLPDU	Data Link Protocol Data Unit	
EMC	Electromagnetic Compatibility	
EUC	Equipment Under Control	[IEC 61508-4:2010]
E/E/PE	Electrical/Electronic/Programmable Electronic	[IEC 61508-4:2010]
FAL	Fieldbus Application Layer	[IEC 61158-5]
FS	Functional Safety	
FSCP	Functional Safety Communication Profile	
MTBF	Mean Time Between Failures	
MTTF	Mean Time To Failure	
PDU	Protocol Data Unit	[ISO/IEC 7498-1]
PFH	Average frequency of dangerous failure [h-1] per hour	[IEC 61508-6:2010[9]]
PhL	Physical Layer	[ISO/IEC 7498-1]
PL	Performance Level	[ISO 13849-1]

9 To be published.

PLC	Programmable Logic Controller	
SCL	Safety Communication Layer	
SIL	Safety Integrity Level	[IEC 61508-4:2010]

3.2.2 CPF 13: Additional symbols and abbreviated terms

ACM	Automatic Configuration Mode	
ADR	Address information (source or destination)	
APDU	Application Protocol Data Unit	[ISO/IEC 7498-1]
CN	Controlled Node	[IEC 61158-3-13]
CT	Consecutive Time field	
DBx	Data Octet	
DSADR	Destination SADR	
DVI	Device Vendor Information	
ID	Frame Identification	
LE	Length field	
LSB	Least Significant Octet	
MAC	Medium Access Layer	
MCM	Manual Configuration Mode	
MN	Managing Node	[IEC 61158-3-13]
MSB	Most Significant Octet	
PDO	Process Data Object	[IEC 61158-3-13]
RxSPDO	Receive Process data objects	
SADR	Safety Address	
SCM	Safety Configuration Manager	
SCT	Safety Control Time	
SD	Safety Domain	
SDG	Safety Domain Gateway	
SDN	Safety Domain Number	
SDO	Service Data Object	[IEC 61158-3-13]
SN	Safety Node	
SNMT	Safety Network Management	
SOD	Safety Object Dictionary	
SPDO	Safety Process Data Object	
SPST	Safety Parameterization Software Tool	
SSDO	Safety Service Data Object	
SSADR	Source SADR	
TADR	Additional SADR for timing information	
TR	Time Request Distinctive Number	
TReq	Time Synchronization Request	
TRes	Time Synchronization Response	
TÜV	Technischer Überwachungsverein	
TxSPDO	Safety Transmit Process data objects	
UDID	Unique Device Identification	

3.3 Conventions

3.3.1 Hexadecimal values

Some values within this part are specified in hexadecimal notation. This shall be depicted by appending a "h" to the value itself. The characters 1, 2, 3, 4, 5, 6, 8, 9, 0, a, b, c, d, e, f, A, B, C, D, E and F shall be used. All values in hexadecimal notation shall depict whole octets; therefore a hexadecimal value representation always consists of an even number of characters.

EXAMPLE 1 12ABh, 05h, 1234ABCDh

To depict data where a sequence of octets shall explicitly be defined – to be independent from the exact memory representation (Big-Endian or Little-Endian) – a hexadecimal sequence notation shall be used. Every octet is shown by a two digit hexadecimal value (without the h suffix), the single octet representations shall be separated by a dash (-).

EXAMPLE 2 00-00-00-00-00-00, AC-3E

3.3.2 Binary values

Some values within this part are specified in binary notation. This shall be depicted by appending a "b" to the value itself. The characters 1 and 0 shall be used.

EXAMPLE 11001010b

3.3.3 Wildcard digits

If parts of constant values are “don't care”, or have additional meaning outside the scope of the descriptive text, this shall be depicted by an x at the respective digit position.

EXAMPLE 12xBh, 1x0xx010b, 123x45

3.3.4 Diagrams

Graphical notations according to ISO/IEC 19501 are used throughout this part where appropriate.

4 Overview of FSCP 13/1 (Ethernet POWERLINK safety)

4.1 Functional Safety Communication Profile 13/1

Communication Profile Family 13 (commonly known as Ethernet POWERLINK[10]) defines communication profiles based on IEC 61158-3-13, IEC 61158-4-13, IEC 61158-5-13, and IEC 61158-6-13.

The basic profile CP 13/1 is defined in IEC 61784-2. The CPF 13 functional safety communication profile FSCP 13/1 (Ethernet POWERLINK safety) is based on the CPF 13 basic profiles in IEC 61784-2 and the safety communication layer specifications defined in this part.

10 Ethernet POWERLINK and Ethernet POWERLINK safety are trade names of the non-profit organization Ethernet POWERLINK Standardization Group (EPSG). This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the trade name holder or any of its products. Compliance to this standard does not require use of the trade names Ethernet POWERLINK or Ethernet POWERLINK safety. Use of the trade names Ethernet POWERLINK or Ethernet POWERLINK safety requires permission of Ethernet POWERLINK Standardization Group (EPSG).

4.2 Technical overview

FSCP 13/1 defines an open safety protocol for Ethernet-based communication in the microsecond range.

The FSCP 13/1 services and protocol specifies safety related data communication between Safety Nodes (SNs). The FSCP 13/1 protocol technology is designed to provide SIL 3 safety function according to IEC 61508. The network configuration is done by using the Safety Configuration Manager (SCM). Safety Network Management (SNMT) services are used for booting and runtime diagnosis of FSCP 13/1 networks. The Safety Object Dictionary (SOD) is database provided by every SN containing configuration and user data. Safety Service Data Objects (SSDO) are used for the exchange of spontaneous data. Safety Process Data Objects (SPDO) provide services for exchange of synchronous data.

Synchronous data communication between SNs is modelled according to the publisher subscriber model (see Figure 3), whereas spontaneous data communication use the client server model (see Figure 4).

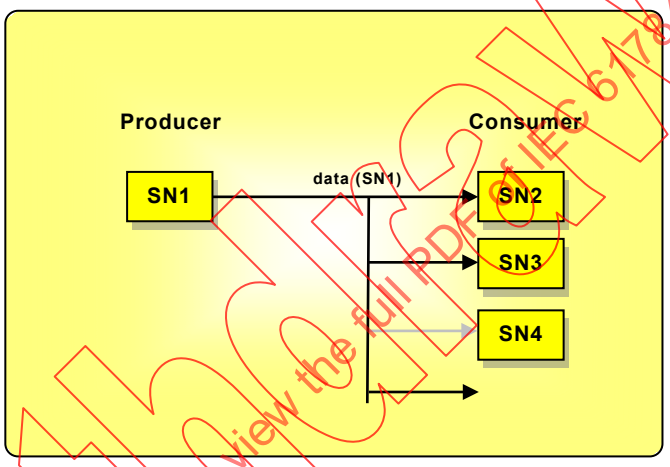


Figure 3 – Producer consumer example

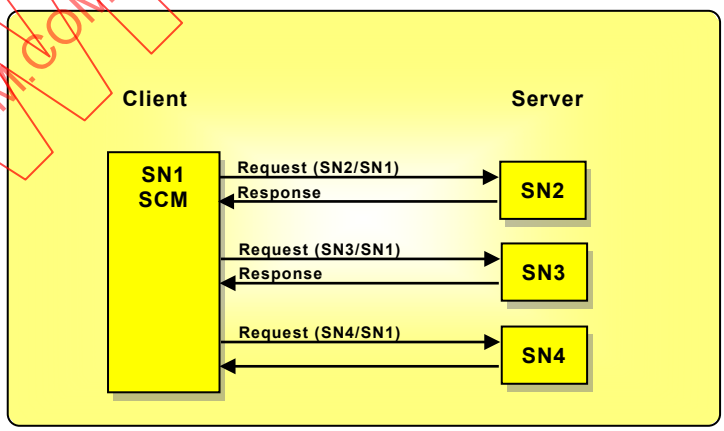


Figure 4 – Client server example

5 General

5.1 External documents providing specifications for the profile

The documents [51] and [52] may help understanding FSCP 13/1 concepts.

5.2 Safety functional requirements

The following requirements were used in the development of FSCP 13/1 and shall apply to the development of devices that implement the FSCP 13/1 protocol:

- FSCP 13/1 shall support Safety Integrity Level 3 (SIL 3) according to IEC 61508,
- The basic requirements for the development of FSCP 13/1 are specified in IEC 61784-3,
- FSCP 13/1 utilizes services defined in IEC 61158-5-13,
- FSCP 13/1 protocol is based on a black channel approach; there is no safety related dependency on the CPF 13 communication profile,
- Implementations of FSCP 13/1 shall comply with IEC 61508,
- CPF 13 requirements shall be unchanged for safety unless specified otherwise in this part,
- Safety communication and standard communication shall be independent. However, standard devices and FSCP 13/1 devices shall be able to use the same communication channel,
- Non safety related network infrastructure devices shall be usable to build a FSCP 13/1 network,
- Up to 1 023 safety related devices shall be supported within one safety domain,
- Up to 1 023 safety domains shall be supported,
- Bandwidth used by safety messages shall be kept as low as possible, safety PDUs shall be optimized for devices with less data (e.g. digital IO), and
- Unless specified otherwise in this part, the safe state of any value shall be zero (0).

5.3 Safety measures

FSCP 13/1 shall use the safety measures listed in Table 1 to detect communication errors for cyclic data transferred in SPDO (see 7.2). The safety measures shall be processed and monitored within each FSCP 13/1 device.

Table 1 – Communication errors and detection measures (cyclic)

Communication errors	Safety measures							
	Sequence number	Time stamp	Time expectation	Connection authentication	Feedback message[e]	Data integrity assurance	Redundancy with cross checking	Different data integrity assurance systems
Corruption						X	X	
Unintended repetition		X						
Incorrect sequence		X						
Loss			X[a]					
Unacceptable delay		X	X					
Insertion			X[b]					
Masquerade			X			X	X	X
Addressing				X				

NOTE The errors “loss” and “insertion” will not be detected by the use of a time stamp. A loss of data is detected within the maximum reaction time by the use of a watchdog. If a message was sent more than once (insertion), the timestamp is unchanged and the receiving device ignores this message.

a Missing PDUs shall be detected within maximum reaction time.

b Only one message shall be accepted during a defined time frame.

FSCP 13/1 shall use the safety measures listed in Table 2 to detect communication errors for acyclic data transferred in SSDO (see 7.3) or SNMT (see 7.4). The safety measures shall be processed and monitored within each FSCP 13/1 device.

Table 2 – Communication errors and detection measures (acyclic)

Communication errors	Safety measures							
	Sequence number	Time stamp	Time expectation	Connection authentication	Feedback message[e]	Data integrity assurance	Redundancy with cross checking	Different data integrity assurance systems
Corruption					X	X	X	
Unintended repetition	X							
Incorrect sequence	X							
Loss	X				X			
Unacceptable delay								
Insertion	X				X			
Masquerade						X	X	X
Addressing				X				
NOTE The errors “loss” and “insertion” will not be detected by the use of a time stamp. A loss of data is detected within the maximum reaction time by the use of a watchdog. If a message was sent more than once (insertion), the timestamp is unchanged and the receiving device ignores this message.								

In order to avoid systematic and stochastic errors, the following guidelines shall be considered during development and implementation of FSCP 13/1 (see 7.1):

- The producer of safety data shall:
  - generate a time stamp,

    NOTE Any new SPDU contains a new time stamp.
  - generate CRCs, and
  - duplicate the safety message;
- The consumer of a safety message data shall:
  - check PDU part one with CRC,
  - check PDU part two with CRC,
  - compare PDU part one and two data area bit-for-bit,
  - check the received address,
  - reset the internal watchdog (counter) if a new valid PDU has been received,
  - if a PDU has the same time stamp as the last one or an older one, the PDU shall be ignored,
  - if no valid PDU is received within a defined time (Safety Control Time, see 7.5.4.14), the device shall enter the safe state, and
  - if the watchdog (counter elapsed) has an overrun (not been reset within the maximum time), the device shall enter a safe state.

5.4 Safety communication layer structure

The FSCP 13/1 protocol is layered on top of a standard communication channel, which shall be considered a black channel. Figure 5 shows how the safety communication layer relates to the standard communication channel.

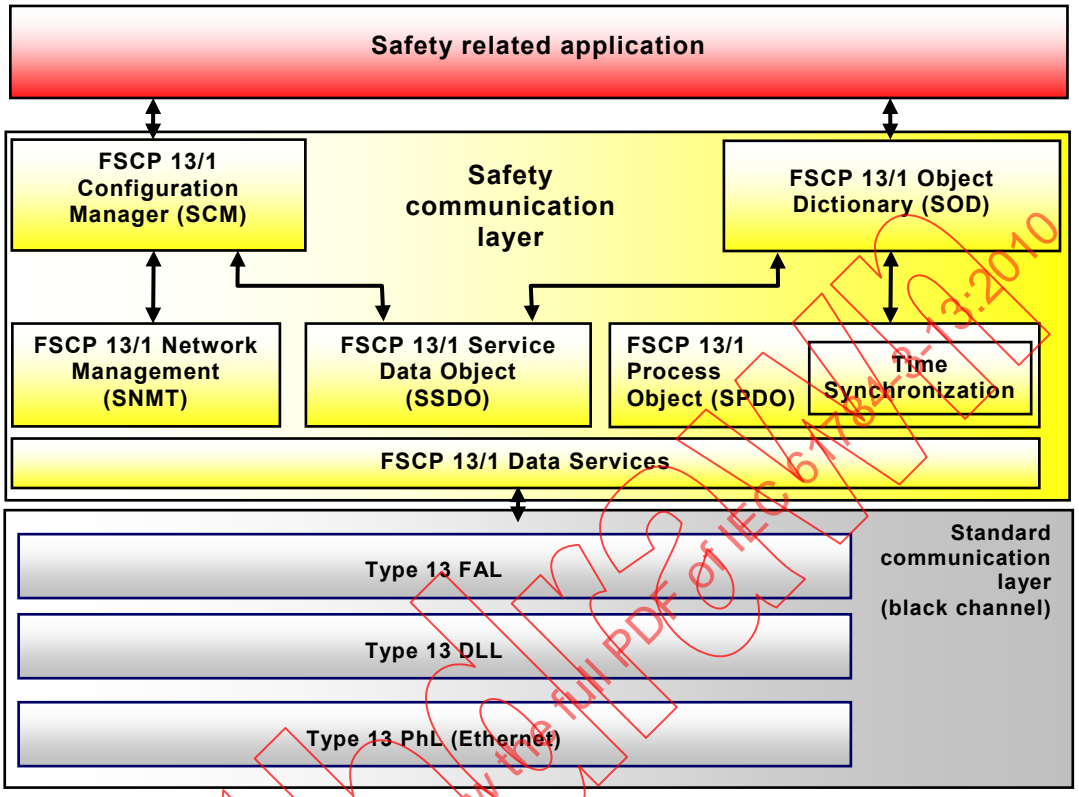


Figure 5 – Communication layer structure

Safety data is produced by the safety-related application entity. This data is encapsulated in a safety message by the safety communication layers and sent via the standard communication channel. The safety communication layer in the receiving device decodes the safety message and verifies integrity and timeliness of the data. The data is presented to the safety-related application entity. If no data is received within a defined time frame, the application entity is informed of this event, which may be used to set all outputs to the safe state.

The safety communication layer provides for:

- safe data transport (integrity and time monitoring),
- network bootup,
- run time diagnosis,
- commissioning,
- program and firmware download,
- parameterization, and
- device replacement.

All these functionalities can be accessed and controlled via the safety communication channel (see Figure 6).

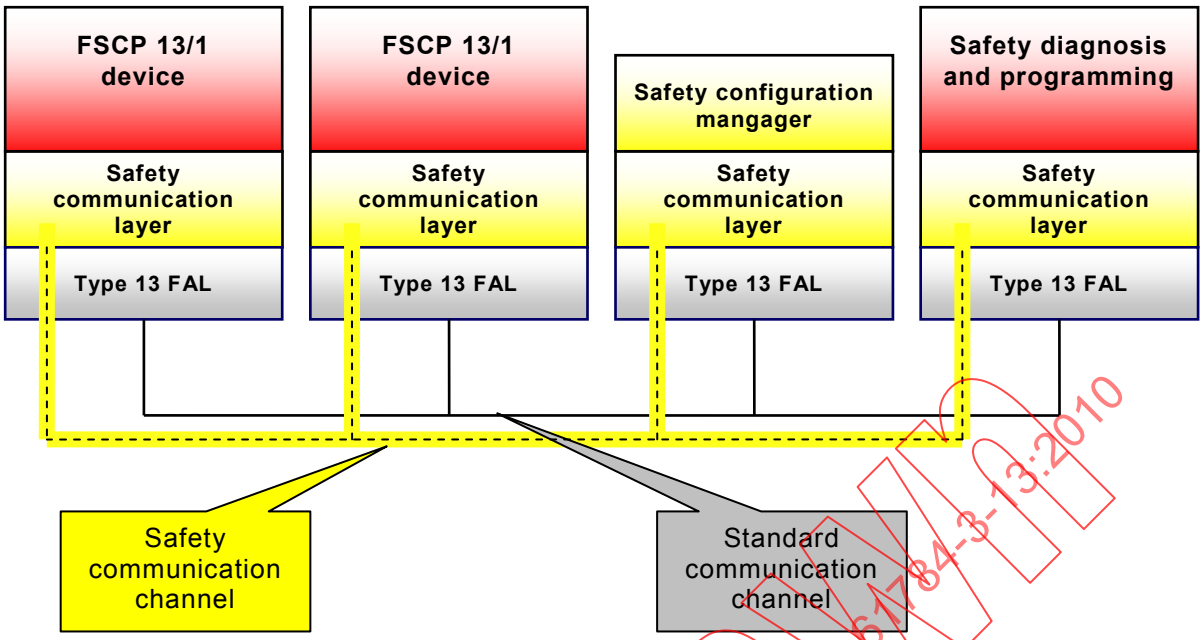


Figure 6 – Safety communication channel

5.5 Relationships with FAL (and DLL, PhL)

5.5.1 General

FSCP 13/1 shall use the services defined in IEC 61158-5-13.

5.5.2 Data types

Profiles defined in this part support all the CPF 13 data types as defined in IEC 61158-5-13.

6 Safety communication layer services

6.1 Modelling

6.1.1 Reference model

6.1.1.1 General

A FSCP 13/1 compliant implementation shall provide the following application objects:

- Safety Network Management (SNMT),
- Safety Configuration Manager (SCM),
- Safety Service Data Objects (SSDO),
- Safety Object Dictionary (SOD), and
- Safety Process Data Objects (SPDO).

How these application objects relate to each other is shown in Figure 5.

The usage of an underlying communication layer shall be mandatory.

6.1.1.2 Safety network management (SNMT)

The SNMT shall provide the required services for the Safety Configuration Manager (SCM) to configure and verify the safety related nodes within the network.

6.1.1.3 Safety service data objects (SSDO)

The SSDO hall provide the required services to access the Safety Object Dictionary (SOD) and to support the SCM.

6.1.1.4 Safety process data objects (SPDO)

The SPDO shall provide the required services for safety related process data exchange between certain entries within the SOD of communicating nodes. The time synchronization services, which are part of SPDO, shall verify the network performance between the communicating nodes.

6.1.1.5 Safety object dictionary (SOD)

The SOD shall describe the format and access services of the device specific data.

6.1.1.6 Safety configuration manager (SCM)

The SCM shall provide the required services for start-up, parameterization and all further measures to ensure safe operation over the entire life cycle of the application.

6.1.2 Communication model

FSCP 13/1 shall use FAL services defined in IEC 61158-5-13 for data exchange between Safety Nodes (SNs), FSCP 13/1 defines a producer / consumer communication model, built using these services. FSCP 13/1 nodes shall be logically separated into producer and consumer nodes.

Producer nodes shall send data identified by a specific Safety Address (SADR). Any SN within the Safety Domain may receive this data. This is done using the mechanism as described in 7.5. As a result of this communication model, the SADR uniquely identifies an SN as well as a safety PDU sent from a specific node.

Figure 7 shows characteristic producer / consumer communication. The producer node SN1 sends data. The data is uniquely identified by the producers SADR. Each node within the Safety Domain is able to receive this data.

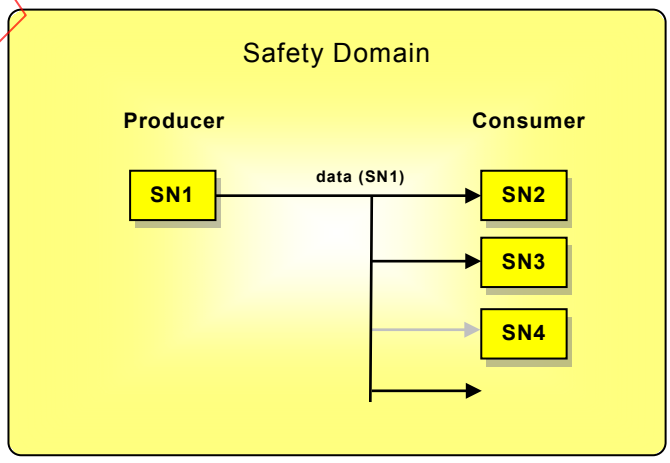


Figure 7 – Characteristic producer / consumer communication

Figure 8 demonstrates an extended version of the producer / consumer communication model which allows the producer to produce separated data for individual consumers. The producer shall use SADRs for this. The Safety Configuration Manager (SCM) shall ensure that the SADRs used are still unique within the Safety Domain.

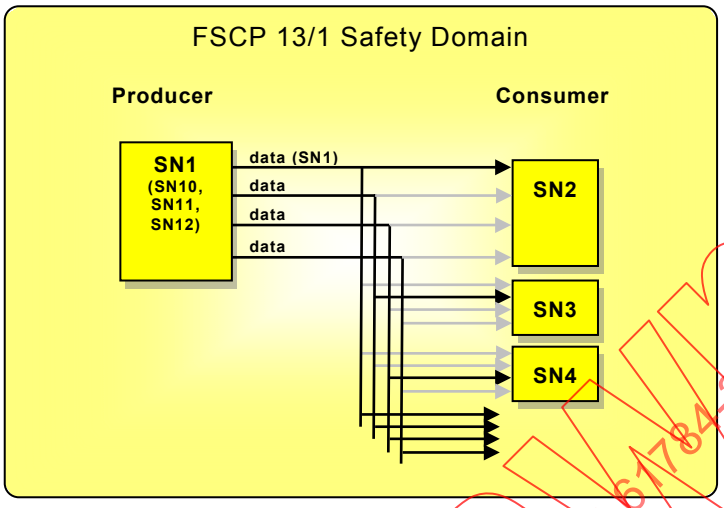


Figure 8 – Extended producer / consumer communication

Figure 9 shows the client server communication which shall be used for configuration services and network management. The SCM (or a dedicated configuration tool, see Clause 8) shall act as the client. SNs shall be limited to the server role.

Using the client server communication model, the client (SCM or Tool) sends a request to the server (SN) which is addressed within the ADR Field (see 7.1.2) using the server address and also contains the client address within the TADR Field (see 7.1.8). The server replies with a response which is addressed within the ADR Field (see 7.1.2) using the server address and also contains the client address within the TADR Field (see 7.1.8).

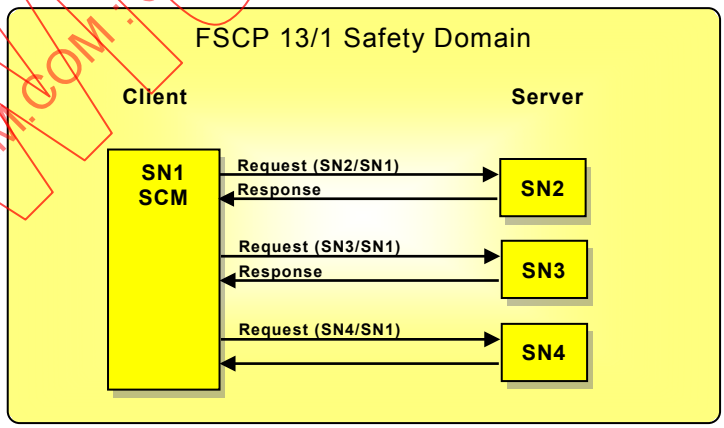


Figure 9 – Client Server communication

6.1.3 Device roles and topology

6.1.3.1 General

FSCP 13/1 device roles are defined in Table 3.

Table 3 – Device roles

Device role	Description
Safety Node (SN)	A FSCP 13/1 compliant device
Safety Configuration Manager(SCM)	Central service for managing a Safety Domain (SD) [a]
Safety Domain Gateway (SDG)	FSCP 13/1 device to exchange data between different SDs
[a] A Safety Domain defines an isolated address space of up to 1 023 SNs.	

An FSCP 13/1 device shall implement the device role of an SN and may implement SCM and/or an SDG.

Figure 10 gives an overview of possible interrelationships of device roles and Safety Domains. SD1 gives an example of a simple Safety Domain. Safety related communication of the FSCP 13/1 devices within SD1 is limited to the SNs within SD1.

SD2 in combination with SD3 shows the possibility to set up a FSCP 13/1 communication between different Safety Domains. This is done using a special SDG which is able to communicate with different Safety Domains. Inter-domain communication is handled by the device internally. SDG capability is independent from the SCM capability of a device. The SDG shall have one SADR in SD2 and one SADR in SD3.

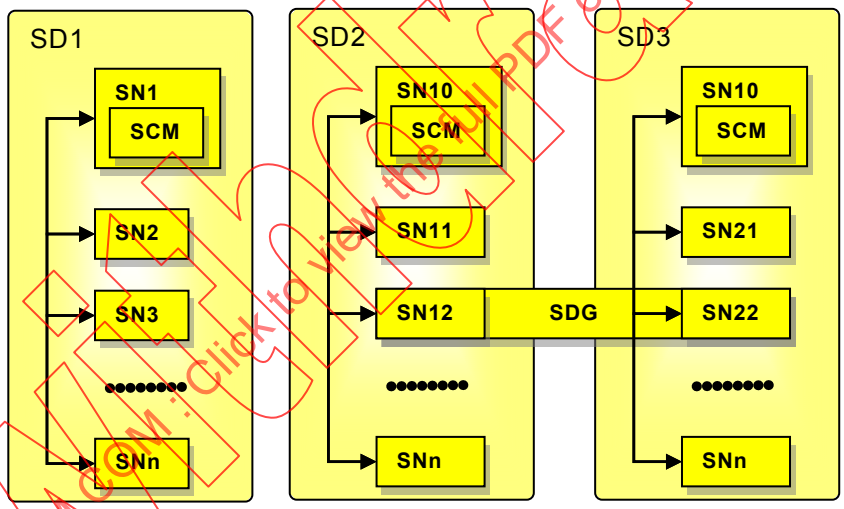


Figure 10 – Topology overview

6.1.3.2 Safety Node (SN)

An SN is a node within any network which conforms to this part. An SN shall be identified by:

- A logical address which is unique within an SD, called Safety Address (SADR). The number range for the SADR shall be 1 -- 1 023 (0 is reserved and shall not be used). The SADR shall be defined at the application planning stage. Duplicate SADRs within an SD shall be detected by the SCM during network verification at start-up of the SCM or at start-up of any node within the Safety Domain.
- A physical address which is globally unique within all FSCP 13/1 compliant devices, called Unique Device Identification (UDID). Duplicate UDIDs within an SD shall be detected by the SCM.

For operation, an SN may need further data like safety related parameters or application data. It depends on the capability of the device, where this data is stored:

- At least the device specific firmware, the UDID and the Device Vendor Information (DVI) (see 7.5.4.7) shall be stored permanently on the device. There shall be no support within FSCP 13/1 to change this data via the network.
- Simple SNs without permanent memory shall receive SDN, SADR and all further parameters from the SCM after start-up.

  NOTE 1 This requires permanent storage of this SN specific data on the SCM.

- Complex SNs with permanent memory or physical switches to adjust the SDN and SADR shall hold their data permanently on the device.

  NOTE 2 The SCM will store only the data which is not permanently stored on the SN; for all other data, the SCM will only verify it.

6.1.3.3 Safety Domain (SD)

6.1.3.3.1 General

A Safety Domain shall constitute an address space of up to 1 023 SADRs (1 – 1 023). Only SNs sharing the same SD shall be able to communicate directly with each other. SNs on different SDs shall be restricted to communication via a Safety Domain Gateway.

An SD is identified by the Safety Domain Number (SDN) having a number range of 1 -- 1 023 0 is reserved and shall not be used. The SDN is defined manually. When planning the application layout, unique SDNs shall be ensured.

6.1.3.3.2 Safety Domain protection

The danger of an external attack of FSCP 13/1 within the safety PDU process data (SPDO) is negligible. But there is a potential risk for the configuration services. When using the Internet in connection with FSCP 13/1, a reliable method of encryption (e.g. Virtual Private Network – VPN, Secure Socket Layer – SSL, IPsec) shall be used. Choosing an adequate encryption system is the responsibility of deployment planning and is outside the scope of this part. Within Local Area Networks (LANs), organizational actions like firewalls shall be implemented.

Figure 11 depicts an example of FSCP 13/1 Safety Domain protection.

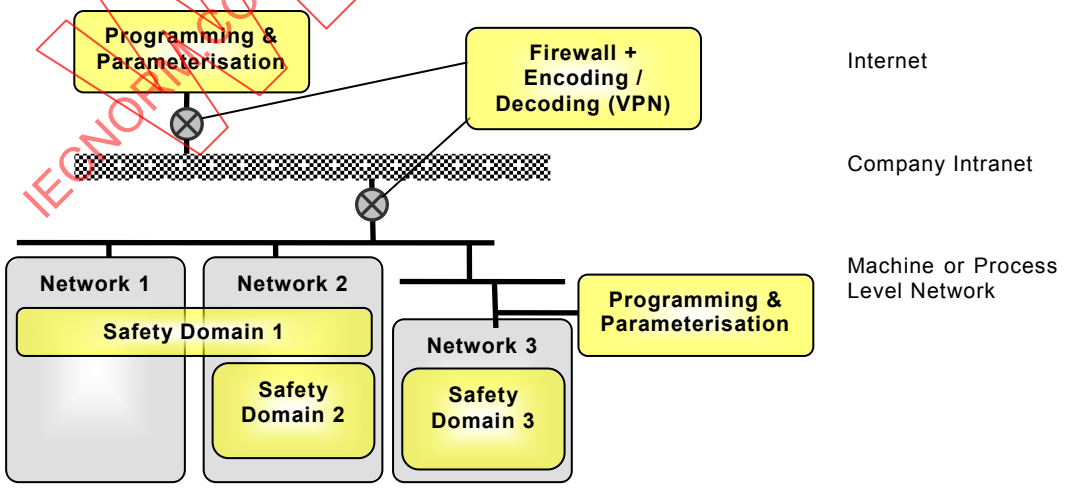


Figure 11 – Safety Domain protection (example)

6.1.3.3.3 Safety Domain separation

Interaction between Safety Domains as well as the danger of confusion during configuration and parameterization is excluded by using unique Safety Domain numbers.

NOTE If the size of the company intranet and the organization of the network management do not allow the administration of unique Safety Domain numbers, it's recommended to split it to several manageable network scopes and separate them by approved methods like firewalls.

To avoid safety critical situations, the SPDO and SSDO PDO data shall additionally be coded to be unique within a certain Safety Domain (see 7.1.10). Therefore Safety PDOs which erroneously are transmitted in the wrong domain can be identified and rejected.

Figure 12 depicts an example of Safety Domain separation.

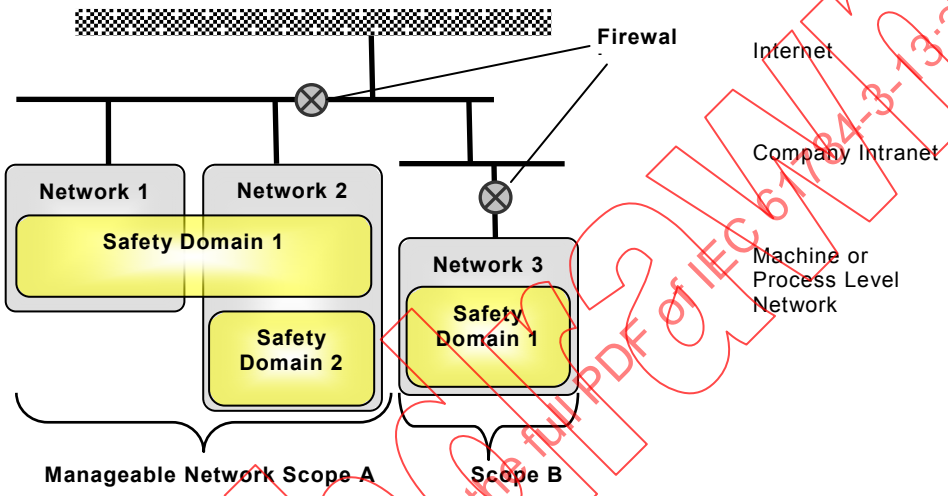


Figure 12 – Safety Domain separation (example)

6.1.3.4 Safety Domain Gateway (SDG)

This shall be a special FSCP 13/1 device role which is able to communicate with different Safety Domains (SD). Inter-domain communication is handled by the SDG internally. Within an SD, the SDG acts like a standard SN.

An SDG is responsible for:

— forward SSDO PDUs between Safety Domains, if an SN wants to send an SSDO to an SN in another Safety Domain, this SSDO is sent to the SDG which forwards the SSDO to the destination node, several SDG may have to be passed to reach the destination SN,

— forward SPDO data between Safety Domains, the SDG may provide means to read data from SPDOs in one Safety Domain and send this data into another Safety Domain inside the SPDO of the SDG.

The configuration of the SDG is vendor specific.

The scope of this part is to specify the safety related communication mechanisms which are needed for data exchange within a single Safety Domain. The communication between different Safety Domains (and the Safety Domain Gateway is a mean for this) is outside the scope of this part.

6.1.3.5 Configuration Manager (SCM)

The Safety Configuration Manager (SCM) shall be centrally responsible for managing FSCP 13/1. Within a Safety Domain, only one SN running the SCM service shall be permitted.

The SCM shall be responsible for:

— permanent storage of SN specific parameters which are not stored on the SN (see 6.1.3.2),

— unique assignment and verification of the SADR of the SNs within the SD,

— verification of the uniqueness of the UDID within the SD,

— verification of the expected parameters and DVI of the SNs, and

— sending a periodic life guard signal to all SNs within the SD.

NOTE This periodic life guard signal from the SCM is required within the SN to detect certain failure situations where no SCM is responsible for the SN.

6.2 Life cycle model

6.2.1 General

FSCP 13/1 supports the safety-related application during several phases of its life cycle, as defined in IEC 61508. The safety related communication is only one part of the complete safety concept of the machinery. Of course, the other safety related parts of the machinery shall be considered as well. This subclause focuses only on safety-related communication with FSCP 13/1.

6.2.2 Concept, planning and implementation

6.2.2.1 Application layout

When designing the application layout, the characteristics of FSCP 13/1 shall be taken into account. The following possibilities and restrictions shall be observed:

— Safety Domain, assigning SDNs,

— separation and protection of Safety Domains,

— Safety Addresses,

— Safety Configuration Manager, and

— Inter-domain communication.

6.2.2.2 Programming and parameterization

6.2.2.2.1 General

Programming and parameterization shall be carried out using safety related tools. Application files and parameters shall be stored in the SCM device only or directly in the corresponding device depending on the device capabilities.

Programming and parameterization of the safety related application shall be comprised of the following tasks:

— accessing SNs within an SD is referenced by the SADR only,

NOTE This allows installing several identical applications with different SDNs without changing the application.

— communication to other SDs is only possible via an SDG,

— locate SCM service,

— list DVI of the devices,

— implement safety related application, and

— specify safety related parameters (see 7.5).

The SCM is the central node to verify the safety related devices. FSCP 13/1 shall support two different strategies for configuration. This is necessary to satisfy all requirements coming from the wide range of applications FSCP 13/1 is ready for. These two strategies shall be:

— Automatic Configuration Mode (ACM), and

— Manual Configuration Mode (MCM).

6.2.2.2.2 Automatic Configuration Mode (ACM)

Using the Automatic Configuration Mode (ACM), the system automatically gathers the UDID from the connected SNs. The ACM shall be used only if all of the following preconditions are fulfilled:

— all safety related devices can be connected at configuration time,

— one device (single UDID) for each SN with a specific SADR,

— if a safety related device is replaced, the SCM verifies that the new device is of the same type as the replaced one; If this is fulfilled, the new UDID will be gathered and the operator shall confirm the replacement.

6.2.2.2.3 Manual Configuration Mode (MCM)

Manual Configuration Mode (MCM) is applicable for all kinds of applications; however this mode requires manual configuration of the safety related devices. So, this mode is mainly intended for applications with the following features:

— all connectable safety related devices can not be connected at one time,

EXAMPLE 1 Robotic arm with several safety related tools.

— multiple devices (multiple UDID) for an SN with a specific SADR,

EXAMPLE 2 Consider a modular machine application with several machine lines within a plant. Machine module A (equals to the SADR) may be used for all machine lines. For bottleneck reasons, there is more than one piece of equipment (i.e. more than one UDID) for module A (=SADR), and these pieces of equipment (UDID) are all applicable on each machine line.

— module replacement shall be handled automatically without manual intervention.

6.2.3 Commissioning

6.2.3.1 General

Commissioning can be separated into three main steps:

a) installing the application (see 6.2.3.2),

b) setting up the configuration depending on the configuration mode (see 6.2.3.3),

c) verification of the safety functionality (see 6.2.3.4).

6.2.3.2 Installation

Install applications and parameters on the FSCP 13/1 devices. This shall be done using one of the following methods:

— an FSCP 13/1 compliant programming tool which uses the specified FSCP 13/1 services (see 7.3), or

— proprietary safety related transfer protocol to exchange data from the programming system to a safety related device via network or direct communication connection (e.g. serial line).

During installation phase of the life cycle, the capability of the device shall determine what data and where the data shall be downloaded:

— if the device itself holds the data, the data shall be downloaded to the device,

— if the device is not able to hold the data, the data shall be stored on the SCM. The SCM shall ensure the proper update of all devices during power up using the FSCP 13/1 configuration services.

Following the installation of applications and parameters, the SDN shall be updated if needed (see 6.1.3.3).

6.2.3.3 Configuration setup

6.2.3.3.1 Configuration setup using ACM

A configuration setup using ACM shall be comprised of the following four steps:

a) power up all safety related devices,

b) ensure that communication layer works for all SNs,

c) SCM starts gathering UDID (without operator interference),

d) the commissioner shall acknowledge each device.

6.2.3.3.2 Configuration setup using MCM

A configuration setup using MCM shall be comprised of the following three steps:

a) the commissioner shall enter the list of possible SADRs with the corresponding UDIDs manually, for this a safety related configuration tool shall be used,

b) the SCM shall accept all configured SADR / UDID combinations,

c) the SCM shall not accept mismatches and shall not repair mismatches as done in the ACM. This means, all mismatches shall be corrected by the commissioner by entering the corrected value using the safety related configuration tool.

6.2.3.4 Verification

Verification shall be comprised of the following five steps:

a) the commissioner shall verify the safety functionality against the safety specification for the application. In the case of device replacements or changes in the routing within the communication layer, the commissioner shall continue with step 6.2.3.3;

b) If verification is carried out during commissioning, all combinations of exchangeable modules shall be verified;

NOTE This is especially important in the case of complex modular machines (in combination with MCM).

c) If verification is carried out due to maintenance, only the safety related functionality affected by the maintenance task shall be verified;

d) When verification is finished, an installation specific backup of the current configuration shall be made;

e) In case of an SCM replacement, a download of application files, parameters, and the configuration shall be done; otherwise the commissioning steps (installation, configuration and verification) shall be repeated.

6.2.4 Operation terms

6.2.4.1 Transfer of safety related data

During operation, the safety related data transfer is handled by FSCP 13/1. The underlying network is responsible for the data transport between the nodes with a defined timing quality. Loss of timing or transport quality within the underlying communication layer is detected by

the FSCP 13/1 layer (see 7.7.2) and will result in loss of availability, but not in loss of safety (black channel approach).

EXAMPLE See Figure 13.

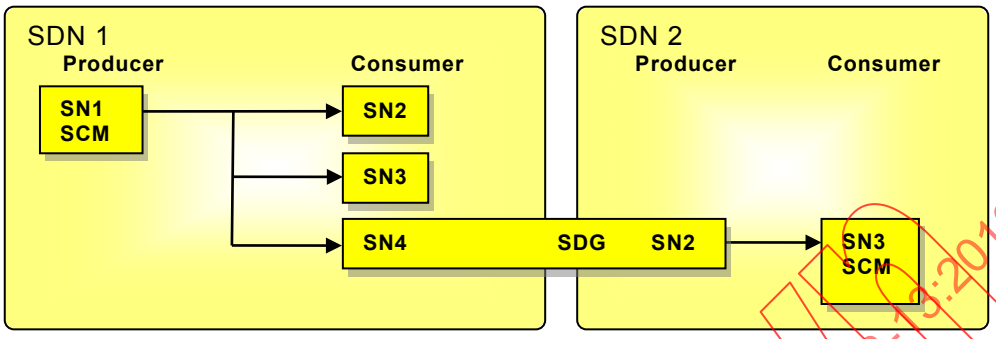


Figure 13 – Data flow example

6.2.4.2 Time synchronization and validation

In order to avoid any missed or sudden delays of data, network performance verification shall be done. From time to time each consumer shall query all connected producers for their relative time. By comparison of the relative time, the elapsed time and the time information from any transfer, the consumer shall check the timeliness of the received data.

The network performance verification is part of the data transfer. At first, a consumer sends a "Time Request" PDU to the connected producer. After receiving this PDU, the producer sends its data with an additional code, which enables the consumer to identify this message as a "Time Response". By measuring the time delay between the "Time Request" and the "Time Response", the consumer is able to check whether the network is fast enough to meet the requirements of the application depending on the worst case reaction time.

For a detailed description refer to 7.7.2.

6.2.4.3 Life guarding

With the service Life Guarding (see 7.5.4.5), the FSCP 13/1 shall ensure that only configured SNs are Operational in an FSCP 13/1 network. An SN expects frequent Life Guarding signals to stay in the Operational state. Otherwise it shall change to the state Pre-Operational where it is of no risk to the safety related network.

The interval of the Life Guarding signal shall be application specific, and is typically determined by the amount of time needed to verify the application safety functions.

Using an organizational measure like switching off all SNs in case of reconfiguring the SCM, may eliminate the risk in this situation. As a result, the interval for Life Guarding may be set to a very large value (several days) for these cases.

6.2.4.4 Startup after power up or reset

After power up, the SCM (see 7.7.9) shall do the network verification. If there was no change within the network, no special tasks are required.

6.2.4.5 Recover from network failure

Following a network breakdown, all affected SNs will enter the Pre-Operational state. Since the corresponding consumer SNs will get no further data, these nodes shall enter the safe state.

After recovery of the network, all affected SNs shall send an SNMT_SN_in_PRE_OP information to the SCM. The SCM, then, shall start with the initialization of the SNs.

6.2.5 Maintenance terms

See 9.6.

6.3 Non safety communication layer

6.3.1 General

This subclause lists the requirements on the non safety communication layer for transportation of safety PDUs. Communication protocols according to CFP 13 (see IEC 61784-2, IEC 61158-3-13, IEC 61158-4-13, IEC 61158-5-13 and IEC 61158-6-13) fulfil all these requirements.

6.3.2 Requirements for data transport

6.3.2.1 General

The non safety communication layer is responsible for transporting the data between the SNs. The coexistence of not safety related nodes and safety related nodes is allowed.

6.3.2.2 Masquerading

To avoid any possibilities of masquerading data as safety PDUs, non safety related nodes shall not

— contain code according to Clause 7,

— contain code which is able to generate safety PDUs according to Clause 7, or

— apply any mechanisms as defined in Clause 7.

See Clause 7 for specific information about handling this issue by protocol mechanisms.

The non safety communication layer is permitted read only access of data within a safety PDU. This shall be done by accessing the data within the safety PDUs without using the safety PDUs specific data integrity measures, such as CRC and data repetition of PDU part one in PDU part two.

6.3.2.3 Communication model

Seen from the view of CPF 13, Figure 14 depicts the common safety communication model of CNs with 0 to n SNs.

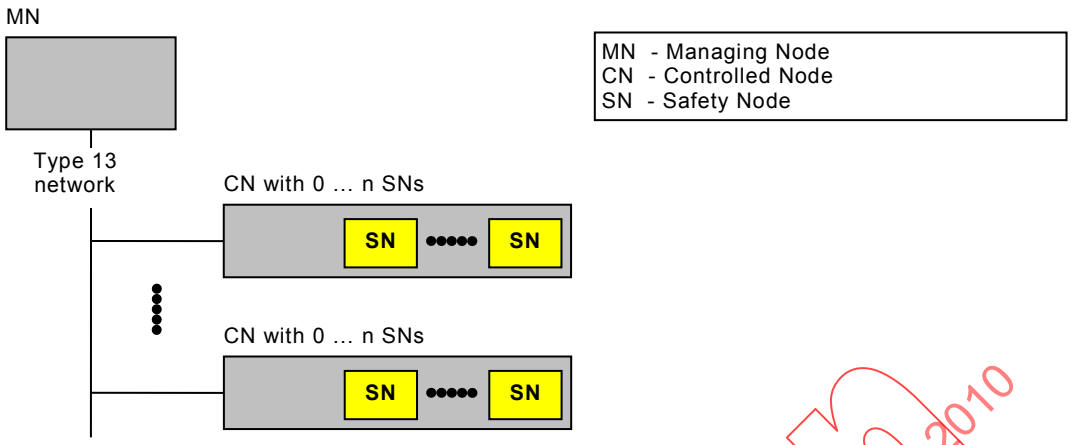


Figure 14 – Communication model

The MN and CN functionality is as defined in IEC 61158-3-13; the MN and CN are not involved in any safety function.

The safety function is implemented by an entity inside the CN which is called the Safety Node (SN). Different cases are possible:

- CN without an SN,
- CN with only one SN,
- CN with several safety related devices represented as a single SN. The safety of the module bus and the add-on modules itself shall be ensured by the vendor,
- CN with multiple SNs, or
- CN as a modular system and multiple SNs. The safety of the bus connecting the modules is provided by FSCP 13/1.

6.3.2.4 Transport of SPDO

The non safety communication layer shall transport the SPDO data between the SNs. The services as defined in IEC 61158-5-13 shall be used.

NOTE 1 SPDO payload is used for cyclic process data, like the state of an input or output. SPDOs are cyclically retransmitted within a certain time. Sending an SPDO is solely time triggered and not bound to a change in the data itself.

SPDO data shall be transmitted as safety PDUs inside Type 13 application layer PDUs (APDUs). Isoc2 APDUs according to IEC 61158-6-13 shall be used. An SPDO is generated cyclically by an SN, the producer, this SPDO can have one or many target SNs, the consumers. To transport an SPDO from its producer to its consumers, Type 13 mechanisms shall be used.

The basic principle of SPDO transport is shown in Figure 15.

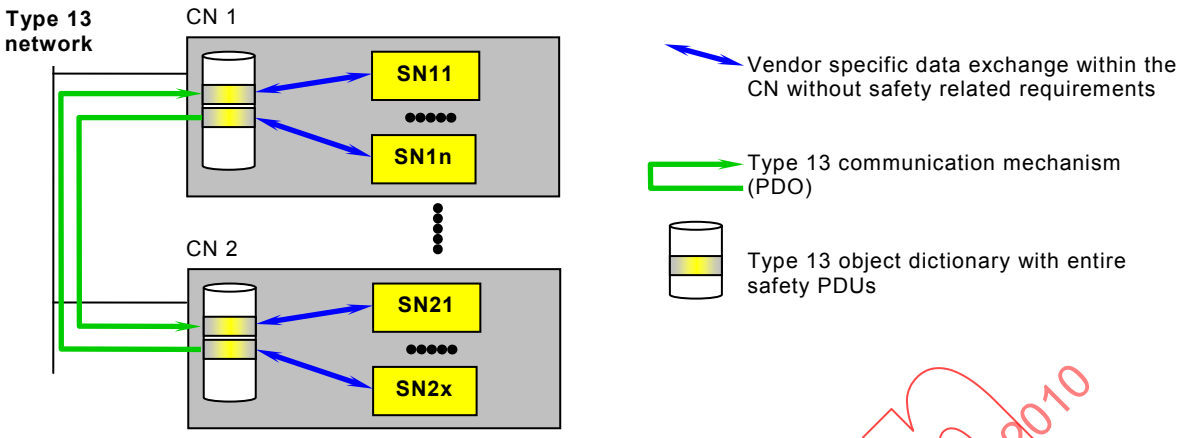


Figure 15 – SPDO transport

The transmission path between the producer SN and the consumer SN is defined as:

- the producer SN builds an SPDU,
- in the CN to which the producer SN is connected (producer CN), an object dictionary entry of data type Domain shall be defined to embody the SPDU (within the communication path, the SPDU shall not be fragmented),
- it is the responsibility of the producer CN implementation to move the SPDU data to the object dictionary entry,
- this object dictionary entry, then, shall be transmitted in a Type 13 application layer PDU (APDU), which is sent cyclically using the process data object ASE (PDO-ASE) as specified in IEC 61158-5-13,
- the object dictionary of the receiving CN (consumer CN) shall define an object of data type Domain, matching the definition of the producer's object dictionary entry; The consumer CN shall receive the APDU and copy the contained safety PDU to the Domain object mentioned before, and
- in the last step, the consumer CN implementation is responsible to move the SPDU data from the object dictionary entry to the consumer SN.

The mechanisms involved in transmitting the SPDO between the CN's object dictionary and the SNs (depicted by blue arrows in Figure 15) are outside the scope of this part and may be vendor specific.

NOTE 2 From the view of the Type 13 communication services and protocol it is not necessary to distinguish communication of SPDO from other data. There is no special treatment of safety data by Type 13 mechanisms. This fits with the black channel approach applied.

NOTE 3 The SNs involved in the communication do not need any knowledge about the Type 13 communication path.

6.3.2.5 Transport of SSDO

The non safety communication layer shall transport the SSDO data between the SNs. The services as defined in IEC 61158-5-13 shall be used.

The basic principle of SSDO transport is shown in Figure 16.

NOTE 1 SPDO payload is used for spontaneous data communication, like configuration or parameterization.

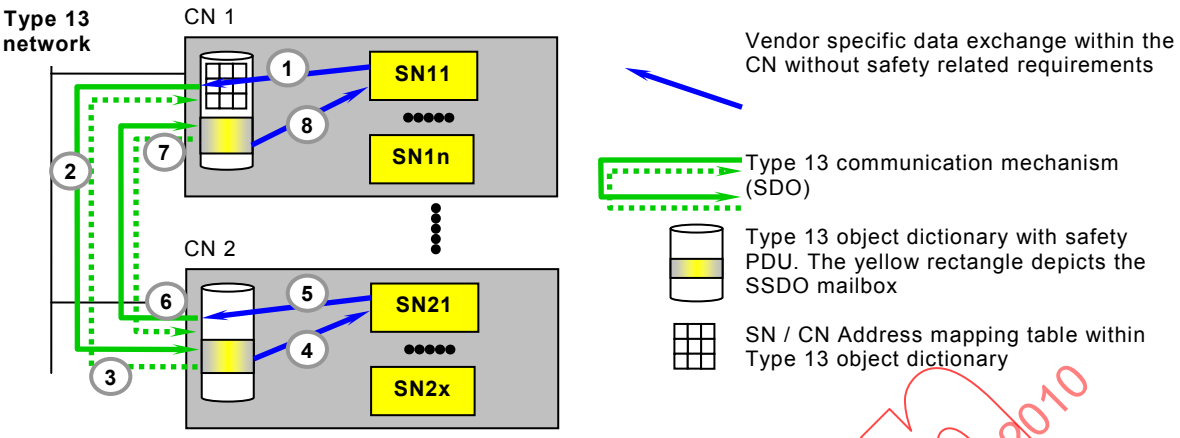


Figure 16 – SSDO transport

SSDO data shall be transmitted as safety PDUs inside Type 13 application layer PDUs (APDUs). Asyn2 APDUs with pduBody type SDO according to IEC 61158-6-13 shall be used.

An SSDO is generated spontaneously by an SN (the source SN) which is transmitted to exactly one target SN. To transport an SSDO from its source to its destination, Type 13 mechanisms shall be used.

Exchanging SSDOs between SNs is done using Type 13 services and protocols in combination with a mailbox mechanism (figures in parentheses references Figure 16):

(1) The source SN generates an SSDO request to a specific destination SN and builds a corresponding SPDU,

(2) the CN serving the source SN creates an SDO write access to the mailbox of the CN serving the destination SN, the SPDU of step (1) is written into the object dictionary of the target CN; This requires an SN to CN address mapping table within the CN serving the source SN, to get the Type 13 node ID of the CN serving the destination SN (target CN); It is the responsibility of the Type 13 system configuration to provide this address mapping table,

the mailbox shall be a Type 13 object dictionary entry of data type Domain which should be defined to embody an SPDU,

(3) the SDO write access is confirmed by the SDO protocol, the connection is kept open,

(4) the destination CN redirects the data (SPDU) within the mailbox to the destination SN, for this the destination CN shall read the address information inside the SPDU,

(5) the destination SN generates an SSDO response and builds a corresponding SPDU,

(6) the destination CN creates an SDO write access to the mailbox of the source CN and writes the response SPDU into the object dictionary of the source CN,

(7) the SDO write access is confirmed, the connection is closed, and

(8) the source CN redirects the data (SPDU) within the mailbox to the destination SN, for this the source CN shall read the address information inside the SPDU.

The mechanisms involved in transmitting the SPDO between the CN's object dictionary and the SNs (depicted by blue arrows in Figure 16) are outside the scope of this part and may be vendor specific.

NOTE 2 From the view of the Type 13 communication services and protocol it is not necessary to distinguish communication of SSDO from other data. There is no special treatment of safety data by Type 13 mechanisms. This fits with the black channel approach applied.

NOTE 3 The SNs involved in the communication do not need any knowledge about the Type 13 communication path.

6.3.2.6 Representation of diagnostic data

The non safety communication layer shall support services to provide FSCP 13/1 specific diagnostic data according to Clause 7.

The basic principle of diagnostic data representation is shown in Figure 17.

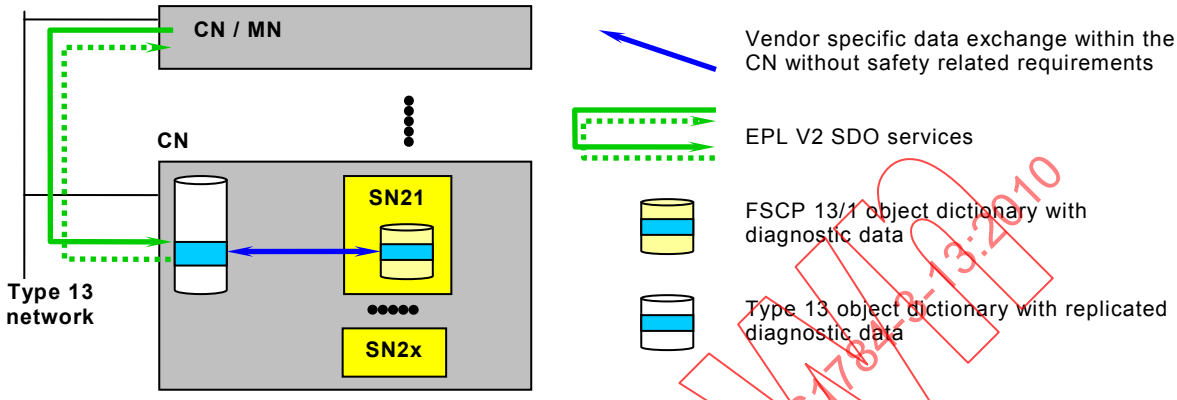


Figure 17 – Diagnostic data representation

Diagnostic data within an SN is stored in the SOD (see Clause 7).

NOTE Usually this diagnostic data is requested from distinguished nodes (e.g. central controller, human machine interface, service equipment) to process the diagnostic data and to report them to the end user.

The CN shall replicate the diagnostic data from the SOD in the Type 13 object dictionary within the CN, where they can access using Type 13 SDO services.

6.3.3 Domain protection and separation

In accordance with 6.1.3.3.2 and 6.1.3.3.3, the communication layer shall support proper security features to protect the domain against attacks from outside and to avoid interferences between separated domains.

7 Safety communication layer protocol

7.1 Safety PDU format

7.1.1 General

A safety PDU is transferred within the data part of a Type 13 APDU. One Type 13 APDU may contain zero to many safety PDUs, each reflecting a producer / consumer relationship between SNs, see Figure 18. Safety and non safety data can be mixed within a Type 13 APDU.

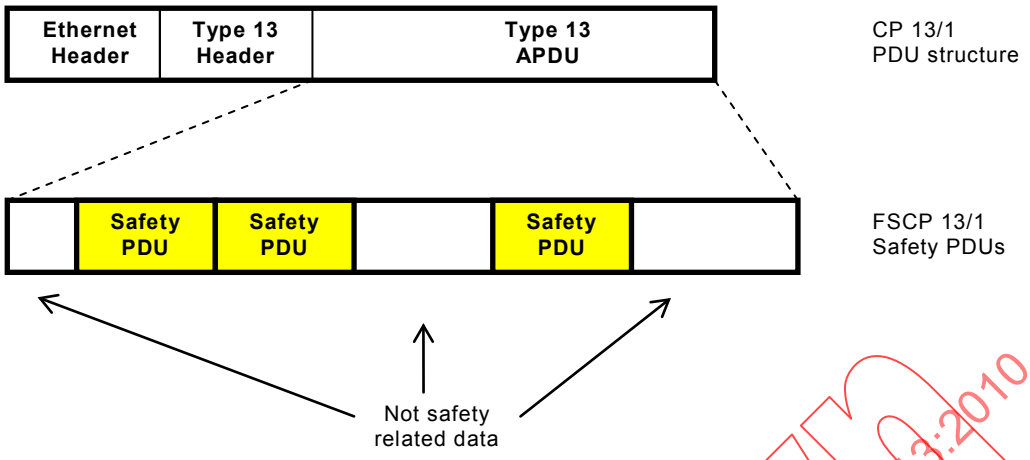


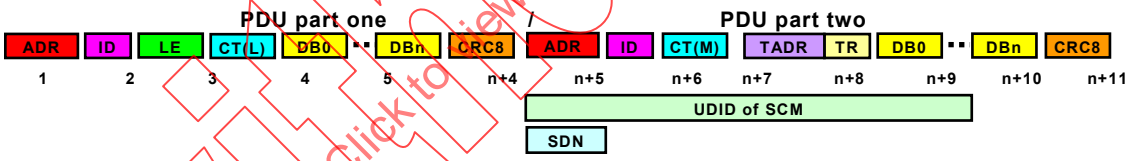
Figure 18 – Safety PDUs inside a CP 13/1 PDU

The structure of the safety PDU shall be as shown in Figure 19 and Figure 20. The safety PDU shall consist of two sub parts and is able to transport up to 254 octets of payload data. For discussion of error probability and hamming distance with respect to the safety PDU length see A.3 and A.4.

Depending on payload length two types of safety PDUs shall be used:

— 1 -- 8 octets of payload: PDU format as shown in Figure 19 shall be used (with an 8 bit CRC);

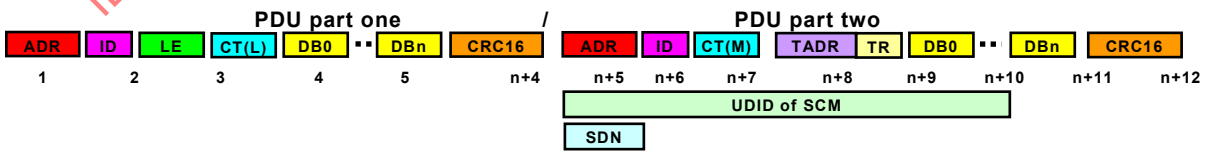
— 9 -- 254 octets of payload: PDU format as shown in Figure 20 shall be used (with a 16 bit CRC).



Key

n number of payload octets

Figure 19 – Safety PDU for n = 0 -- 8 octet payload data



Key

n number of payload octets

Figure 20 – Safety PDU for n = 9 -- 254 octet payload data

Application of this part according to the requirements defined in IEC 61784-3:2010, 5.6.1 (calculation of the residual error rate) results in a restriction of payload octets to the range starting from 9 to 25 octets.

NOTE 1 IEC 61784-3:2010, 5.6.1 states "unless a better error probability can be proven, a value of 10^{-2} shall be used". See A.3 for calculations of the residual error rate in case that the error probability is 10^{-2}.

NOTE 2 A.4 shows which preconditions have to be fulfilled to justify a bit error rate of 10^{-3} and the results of the residual error rate calculations in this case.

Table 4 specifies the safety PDU. The PDU shall be constructed in two parts, called PDU part one and PDU part two. The data of PDU part two of SPDO and SSDO telegrams (not SNMT telegrams) are additionally coded with the UDID of the SCM using a logical XOR operation (see 7.1.10).

Table 4 – PDU format

Octet offset [a]	Bit offset							
	7	6	5	4	3	2	1	0
0	ADR (Bit 0-7), see 7.1.2							
1	ID, see 7.1.3						ADR (8,9)	
2	LE, see 7.1.4							
3	CT (Bit 0-7) , see 7.1.5							
4 -- n+3	DB 0 to DB n, see 7.1.6							
n+4 -- n+4+o	CRC-8 / CRC-16, see 7.1.7							
n+5+o	ADR (Bit 0-7) XOR SDN (Bit 0-7)							
n+6+o	ID						ADR (8,9) XOR SDN (8,9)	
n+7+o	CT (Bit 8-15)							
n+8+o	TADR (Bit 0-7), see 7.1.8							
n+9+o	TR, see 7.1.9						TADR (8,9)	
n+10+o -- 2n+9+o	DB 0 -- DB n							
2n+10+o -- 2n+10+2o	CRC-8 / CRC-16							
NOTE The grey colored lines indicates the data of PDU part two.								
[a] Relative to start of the safety PDU. n shall be the number of payload octets in the field DB 0 to DB n, where 0 ≤ n ≤ 254. o shall be the CRC correction offset, if 0 ≤ n ≤ 8 then o = 0 (CRC 8), if 9 ≤ n ≤ 254 then o = 1 (CRC-16).								

The fields of the safety PDU are further defined in the following subclauses.

7.1.2 Address field (ADR)

The ADR field defines the Safety Address (SADR) of the Safety Node (SN). The field shall be a 10 bit value. ADR shall support a value range from 1 to 1 023 (0 is reserved and shall not be used).

The ADR field within PDU part two shall also be used to carry the Safety Domain Number (SDN) information by using a logical XOR operation (see 7.1.10).

The ADR field shall contain the SADR of the SN from which the PDU is sent (source address) or for which the PDU is designated (destination address). The source address shall be the SADR of the SN which sends the PDU; the destination address shall be the SADR of the SN which the telegram is sent to.

According to the content of the ADR fields of both PDU parts, the non safe communication layer is able to transport the data to the designated destination node without additional analyzing the ID field or other PDU attributes.

7.1.3 PDU identification field (ID)

The ID field shall identify the PDU type. Table 5 specifies the coding of the ID field. Reserved codes may be used in future editions of this part. Currently, they shall not be used but shall be ignored if received. For identification of the PDU type, bits 5, 6 and 7 shall be used. Each PDU type has different telegram types which shall be specified bit 2, 3 and 4 of the ID field.

Table 5 – PDU identification field (ID)

Bit Offset								PDU type
7	6	5	4	3	2	1	0	
0	0	0	0	0	x [a]	bit 9 of the address field	0 bit 8 of the address field	Not allowed (illegal PDU)
0	x	x	x	x	x			reserved
1	0	0	x	x	x			reserved
1	0	1	x	x	x			SNMT
1	1	0	x	x	x			SPDO
1	1	1	x	x	x			SSDO

[a] An x within this table specifies the telegram type for a specific PDU type.

Table 6 specifies all valid ID field combinations.

Table 6 – Used ID field combinations

Bit Offset								Telegram type
7	6	5	4	3	2	1	0	
1	0	1	0	0	0	bit 9 of the address field	bit 8 of the address field	SNMT_Request_UDID
1	0	1	0	0	1			SNMT_Response_UDID
1	0	1	0	1	0			SNMT_Assign_SADR
1	0	1	0	1	1			SNMT_SADR_assigned
1	0	1	1	0	0			SNMT_Service_Request
1	0	1	1	0	1			SNMT_Service_Response
1	0	1	1	1	1			SNMT_SN_reset_guarding_SCM
1	1	0	0	0	0			SPDO_Data_Only
1	1	0	0	1	0			SPDO_Data_with_Time_Request
1	1	0	1	0	0			SPDO_Data_with_Time_Response
1	1	1	0	0	0			SSDO_Service_Request
1	1	1	0	0	1			SSDO_Service_Response

SSDO and SNMT PDUs shall be divided into request and response telegrams as shown in Table 7. Bit 2 of the ID field shall be used to identify whether the telegram is a request or a response. Bits 0 and 1 of the octet where the ID field is located are used by the address field (see Table 4).

Table 7 – Request / response identification

Bit Offset								Description
7	6	5	4	3	2	1	0	
x	x	x	x	x	0			SSDO / SNMT Request
x	x	x	x	x	1			SSDO / SNMT Response

7.1.4 Length field (LE)

The LE field specifies the number of payload data octets within the PDU. The valid value range shall be 0 -- 254. The value of LE shall determine the type of CRC used for the PDU as specified in Table 8.

Table 8 – Type of CRC depending on LE

LE value	Header 2
0 -- 8	CRC-8
9 -- 254	CRC-16
255	reserved

7.1.5 Consecutive Time field (CT)

The Consecutive Time (CT) field shall be divided in an LSB part which is sent in PDU part one and an MSB part which is sent in PDU part two. Both octets build the CT data set with a number range of 0 -- 65 535. The CT field shall be used for the internal timer value of the SN when sending the PDU in case of an SPDO (see 7.2), in case of an SSDO, the CT field describes the SOD access request number (see 7.3). The time base of the CT field shall be defined in SOD object 1200h sub index 03h (see 7.5.4.11).

7.1.6 Payload data field (DB0 to DBn)

This field shall contain the payload data. A payload data length up to 254 octets shall be supported.

7.1.7 Cyclic Redundancy Check field (CRC-8 / CRC-16)

The CRC field shall contain the cyclic redundancy check value for the PDU part it belongs to. The type (CRC-8 or CRC-16) depends on the value of LE.

NOTE 1 The residual error probability as well as the expected hamming distance is discussed in A.3 and A.4.

Polynomial that shall be used for CRC-8: $X^8+X^5+X^3+X^2+X+^1$

Polynomial that shall be used for CRC-16: $X^{16}+X^{14}+X^{12}+X^{11}+X^8+X^5+X^4+X^2+1$

The following rules for CRC generation shall apply:

1) Seed value is 0 for CRC-8 and for CRC-16;
2) The most significant bit of the first octet is shifted in first;
3) The CRC shall be calculated over all fields of the PDU part excluding the CRC;
   i) PDU part one: ADR, ID, LE, CT(L), DB0–DBn;
   ii) PDU part two: ADR, ID, CT(H), TADR, TR, DB0–DBn.

The CRC-8 shall comply with the encoding rules of an UNSIGNED8 data type.

The CRC-16 shall comply with the encoding rules of an UNSIGNED16 data type.

NOTE 2 Usage of CRC8 is justified only if special preconditions regarding the bit error rate are satisfied, see A.4.

7.1.8 Time Request Address field (TADR)

The TADR field specifies:

— the Safety Address (SADR) of an SN which responds to the Time Synchronization Request in case of SPDO (see 7.2), or

— the SADR (source or destination) in case of SSDO (see 7.3) and SNMT (see 7.4).

7.1.9 Time Request Distinctive Number field (TR)

The TR field shall be a number in the range 0 -- 63, which shall be mirrored by the device in order to distinguish this message from other time request telegrams.

7.1.10 UDID of SCM coding (UDID of SCM)

The 6 octets UDID of the SCM shall be coded by a bitwise exclusive or operation with the first 6 octets of the payload data of PDU part two of SPDO and SSDO telegrams. SNMT telegrams shall not be coded in this way.

The UDID of the SCM is known for every device (see 7.4.3.12), the SN shall test every received SPDO and SSDO telegrams whether the correct UDID was used for coding the telegrams.

NOTE By doing this, mismatches in the domain separation (see 6.1.3.3.3) can be detected.

7.2 Safety Process Data Objects (SPDO)

7.2.1 General

Safety Process Data Objects (SPDOs) shall be used for transmitting safety-critical process data as well as time synchronization request/response data from an SPDO producer to the SPDO consumers. Reception of an SPDO by an SPDO consumer shall be determined by the address of the SPDO producer.

7.2.2 SPDO telegram types

The ID field (bit 2-4) shall identify the SDPO telegram type. Three telegram types shall be defined:

— data only telegram,

— data with Time Request telegram, and

— data with Time Response telegram.

Table 9 specifies the coding of the ID field.

Table 9 – SPDO telegram types (ID field, bits 2, 3 and 4)

Bit number of ID field								PDU Type
7	6	5	4	3	2	1	0	
1	1	0	0	0	0	bit 8,9 of the Address field		SPDO_Data_Only
1	1	0	x	x	1			Reserved
1	1	0	0	1	0			SPDO_Data_with_Time_Request
1	1	0	1	0	0			SPDO_Data_with_Time_Response
1	1	0	1	1	0			Reserved

7.2.3 Data Only telegram

The Data Only telegram (see Figure 21) shall be used to transmit process data from an SN (SPDO producer) without a time synchronization request or response. The data can be consumed by any other SN (SPDO consumer).

Together with the data (Data), the SPDO producer shall provide its SN address (ADR) as well as the number of payload data octets transmitted (LE) and the current value of its internal timer (CT).

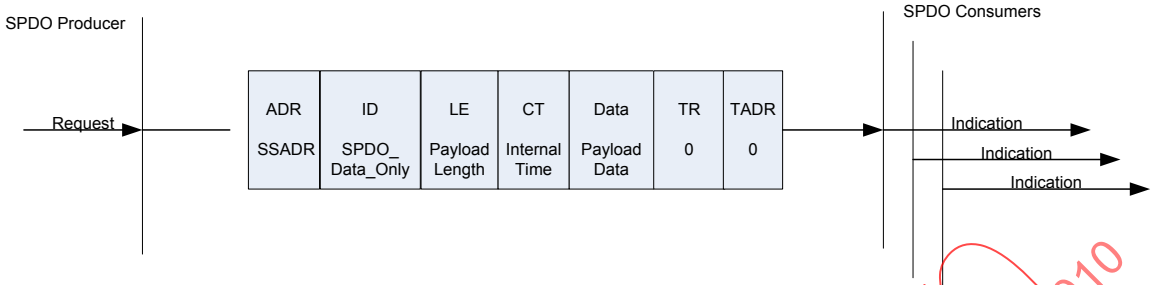


Figure 21 – SPDO_Data_Only telegram

Table 10 specifies the fields of an SPDO_Data_Only telegram.

Table 10 – Fields of SPDO_Data_Only telegram

Field	Information	Content / Value
ADR	Address of SPDO producer	SSADR
ID	SPDO_Data_only	$110000xx_b$
LE	Number of payload data octets	0 -- 254
CT	Internal time value of SPDO producer	Time
Data	Payload data	Data
TR	Not used	0
TADR	Not used	0

7.2.4 Data with Time Request telegram

The Data with Time Request telegram (see Figure 22) shall be used to transmit process data from an SN (SPDO producer) with a Time Synchronization Request to another SN addressed by TADR. The data can be consumed by any other SN (SPDO consumer).

Together with the data (Data), the SPDO producer provides its SN Address (ADR), the telegram type (ID) as well as the number of payload data octets (LE) transmitted, the current value of its internal timer (CT), a counter value for indicating the number of time requests (TR) and the address of the SN which is requested to provide its current internal time (TADR).

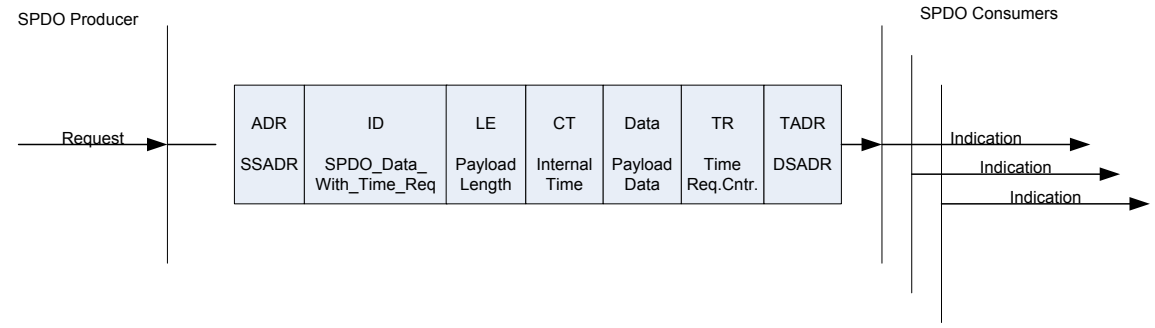


Figure 22 – SPDO_Data_with_Time_Request telegram

Table 11 specifies the fields of an SPDO_Data_with_Time_Request telegram.

Table 11 – Fields of SPDO_Data_with_Time_Request telegram

Field	Information	Content / Value
ADR	Address of SPDO producer	SSADR
ID	SPDO_Data_with_Time_Request	110010xxb
LE	Number of payload data octets	0 -- 254
CT	Internal time value of SN	Time
Data	Payload data	Data
TR	Internal Time Request Counter	0 -- 63
TADR	Address of SN from which time information is requested	DSADR

7.2.5 Data with Time Response telegram

The Data with Time Response telegram (see Figure 23) shall be used to transmit process data from an SN (SPDO producer) together with the internal timer value of the requested node. The data can be consumed by any other SN (SPDO consumer).

Together with the data (Data), the SPDO producer provides its SN address (ADR), the telegram type (ID) as well as the number of payload data octets (LE) transmitted, the current value of its internal timer (CT), the counter value for indicating the number of the time requests (TR) and the address of the SN which has issued the request (TADR).

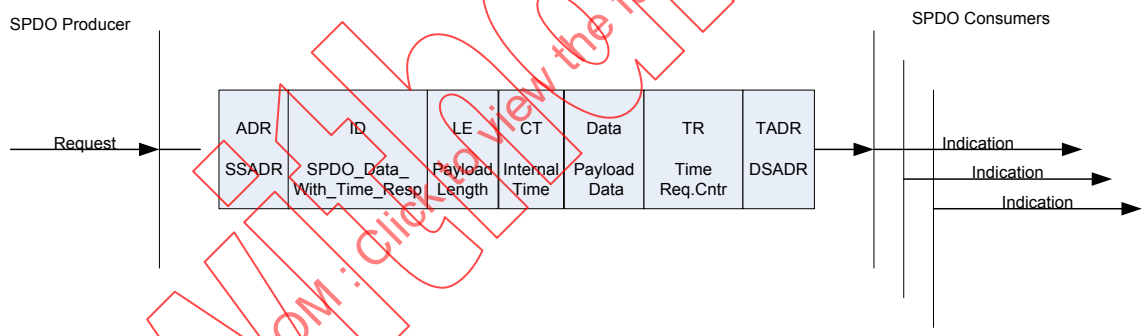


Figure 23 – SPDO_Data_with_Time_Response telegram

Table 12 specifies the fields of an SPDO_Data_with_Time_Response telegram.

Table 12 – Fields of SPDO_Data_with_Time_Response telegram

Field	Information	Content / Value
ADR	Address of SPDO producer	SSADR
ID	SPDO_Data_with_Time_Response	110100xxb
LE	Number of payload data octets	0 -- 254
CT	Internal time value of SPDO producer	Time
Data	Payload data	Data
TR	Time Request Counter value of corresponding request	0 -- 63
TADR	Address of the SN which issued the time information request	DSADR

7.3 Safety Service Data Object (SSDO)

7.3.1 General

Safety Service Data Objects (SSDOs) shall be used for accessing the Object Directory (SOD) of an SSDO-Server by an SSDO-Client. SSDO telegrams shall be identified by bits 7, 6 and 5 (111b) of the ID. Bit 3 and 4 of the ID shall identify an SOD-access (00b). Bit 2 shall identify request (0b) and response (1b) telegrams.

7.3.2 SSDO telegram types

The ID field identifies the telegram. The following telegram types are specified:

— SSDO Service Request

— SSDO Service Response

Table 13 specifies the coding of the ID field.

Table 13 – SSDO telegram types (ID field, bits 2, 3 and 4)

Bit number of ID field								Telegram type
7	6	5	4	3	2	1	0	
1	1	1	0	0	0			SSDO_Service_Request
1	1	1	0	0	1			SSDO_Service_Response

The SSDO Service Request telegram (SSDO_Service_Request) shall be used by an SSDO client to access the SOD of an SSDO server. With an SSDO_Service_Request, an SSDO client shall provide the server SN address (ADR), the length of the payload data (LE), the Access Request Number SANo (CT field). It shall also specify in the first data octet in the "Command Octet" (SACmd, see Table 14) access type (Read/Write), transfer type (expedited/segmented), toggle bit and initialize or last segment of the transfer. In data octets 1 and 2, the index of the accessed object dictionary entry, in data octet 3 the sub index of the accessed entry shall be specified. The SN address of the SSDO client shall be provided in TADR.

The SSDO Service Response telegram (SSDO_Service_Response) shall be used by an SSDO server to respond to a request from an SSDO client. The SOD Access Request Number (SANo) shall be unique for each connection and incremented by one with every new SOD Access Request telegram. The response telegram shall contain the SANo value of the corresponding service request.

The maximum number of payload data shall be 250 octets.

The SOD Access Command (SACmd) bit encoding is specified in Table 14.

Table 14 – SOD Access Command (SACmd) – bit coding

Bit No.	Name	Value	Meaning
0	Access Type	0	SOD Read Access
		1	SOD Write Access
1	Reserved 1	0	Reserved
2	Abort transfer	0	Successful transfer
		1	Abort transfer
3	Segmentation	0	Expedited SOD access

Bit No.	Name	Value	Meaning
		1	Segmented SOD access
4	Toggle	0	This bit shall alternate for each subsequent segment and not change its value in case of a repeated transmission. The first segment shall have the toggle bit set to 0. The toggle bit shall be equal for the request and the response telegram.
		1	
5	Initiate transfer	0	No initiate transfer
		1	Initiate transfer
6	End segment transfer	0	More segments to transfer
		1	No more segments to transfer
7	Reserved 2	0	Reserved

7.3.3 SSDO services and protocols

The following SSDO services shall be defined:

- SSDO Download, which is subdivided into:
  - SSDO Download Initiate,
  - SSDO Download Segment;
- SSDO Upload, which is subdivided into:
  - SSDO Upload Initiate,
  - SSDO Upload Segment;
- SSDO Abort.

Figure 24 defines the SSDO segmented and expedited download protocols.

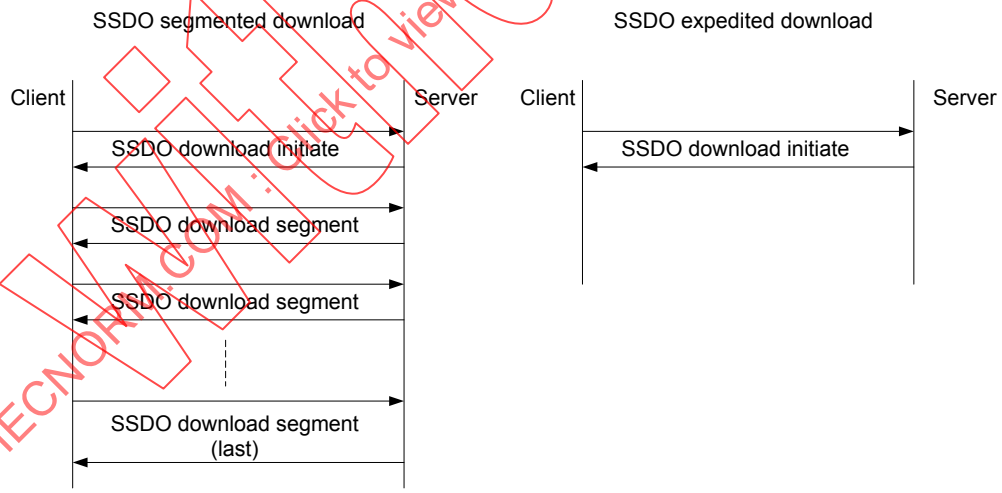


Figure 24 – SSDO download protocols

Figure 25 defines SSDO segmented and expedited upload protocols.

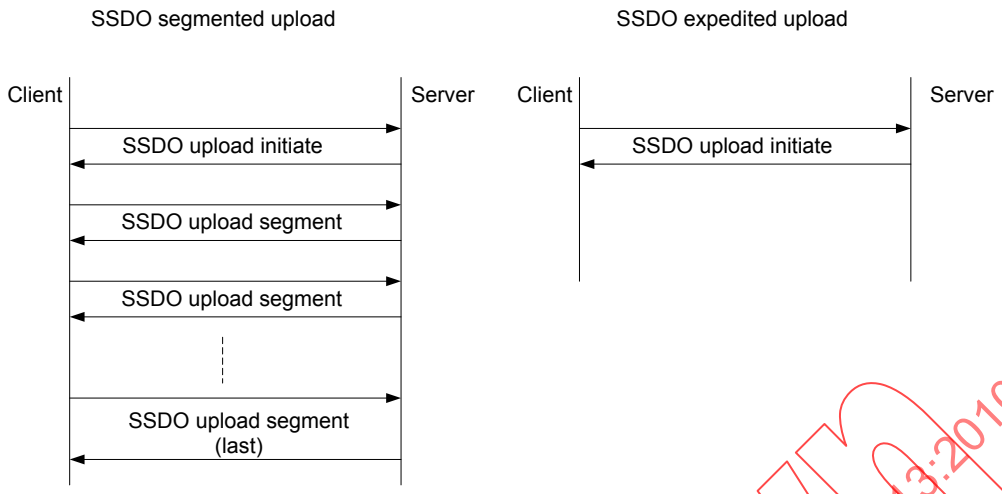


Figure 25 – SSDO upload protocols

7.3.4 SSDO Initiate Download

Figure 26 specifies the initiate download protocol.

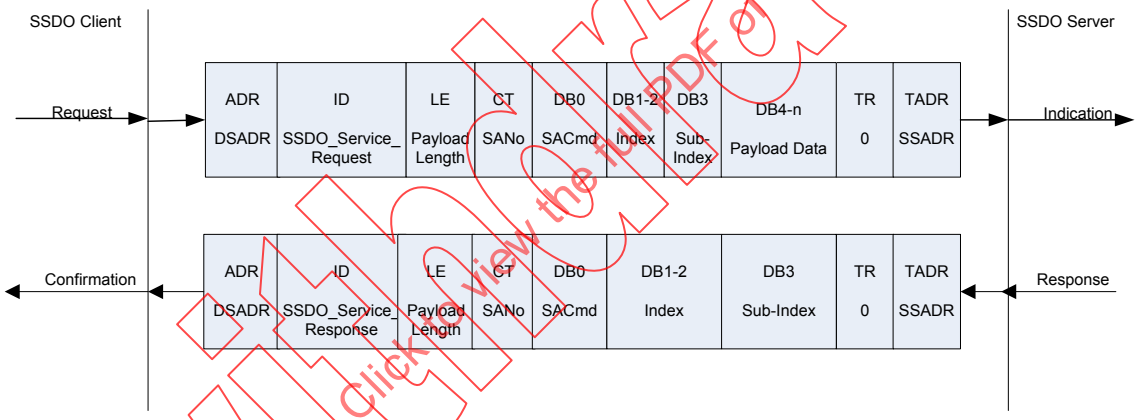


Figure 26 – SSDO Initiate Download protocol

Table 15 specifies the fields of an Initiate Download SSDO_Service_Request telegram.

Table 15 – Fields of Initiate Download SSDO_Service_Request telegram

Field	Information	Content / Value
ADR	Address of SSDO Server to be accessed	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Length of payload data	4 -- 254
CT	SOD Access Request number (SANo)	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) expedited or segmented	00100001b or 00101001b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
DB4 – DBn	Payload data (In case of a segmented download, DB4-DB7 contains the data size of the payload data of the block. If the data size is 0, the size is not indicated.)	Payload data
TR	Not used	0
TADR	Address of SSDO Client	SSADR

Table 16 specifies the fields of an Initiate Download SSDO_Service_Response telegram.

Table 16 – Fields of Initiate Download SSDO_Service_Response telegram

Field	Information of	Content / Value
ADR	Address of SSDO Client to be responded	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Length of payload data	4
CT	Echo of SOD Access Request number	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) expedited or Segmented	00100001b or 00101001b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
TR	Not used	0
TADR	Address of SSDO Server	SSADR

7.3.5 SSDO Segmented Download

Figure 27 specifies the Segmented Download protocol.

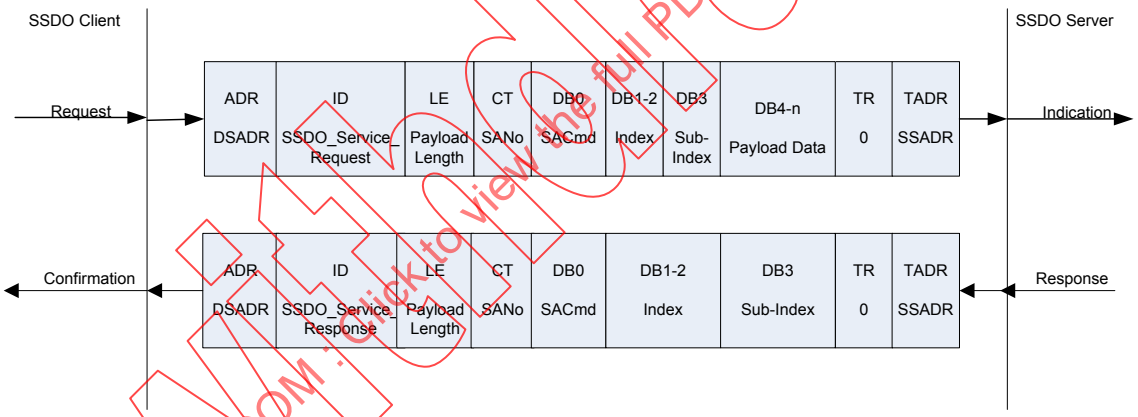


Figure 27 – SSDO Segmented Download protocol

Table 17 specifies the fields of a Segmented Download SSDO_Service_Request telegram.

Table 17 – Fields of Segmented Download SSDO_Service_Request telegram

Field	Information	Content / Value
ADR	Address of SSDO Server to be accessed	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Length of payload data	4 -- 254
CT	SOD Access Request number (SANo)	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) middle segment or end segment	000x1001b or 010x1001b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
DB4 – DBn	Payload data	Payload data

Field	Information	Content / Value
TR	Not used	0
TADR	Address of SSDO Client	SSADR

Table 18 specifies the fields of a Segmented Download SSDO_Service_Response telegram.

Table 18 – Fields of Segmented Download SSDO_Service_Response telegram

Field	Information	Content / Value
ADR	Address of SSDO Client to be responded	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Length of payload data	4
CT	Echo of SOD Access Request number	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) middle segment or end segment	000x1001b or 010x1001b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
TR	Not used	0
TADR	Address of SSDO Server	SSADR

7.3.6 SSDO Initiate Upload

Figure 28 specifies the Initiate Upload protocol. The Initiate Upload telegram shall always be expedited. Based on the response the SSDO client shall continue with expedited or segmented transfer.

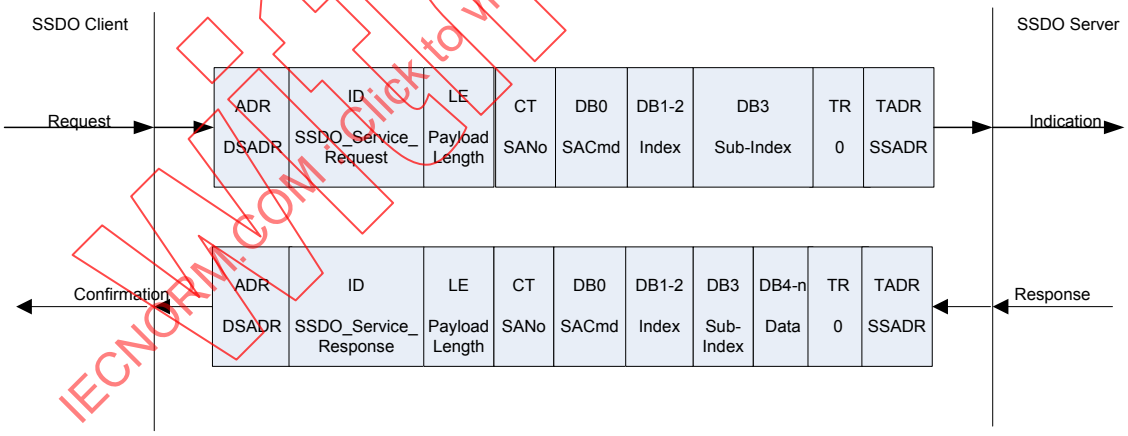


Figure 28 – SSDO Initiate Upload protocol

Table 19 specifies the fields of an Initiate Upload SSDO_Service_Request telegram.

Table 19 – Fields of Initiate Upload SSDO_Service_Request telegram

Field	Information	Content / Value
ADR	Address of SSDO Server to be accessed	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Length of payload data	4
CT	SOD Access Request number (SANo)	1 -- 65 535, 0 not allowed

Field	Information	Content / Value
DB0	SOD Access Command (SACmd) expedited	00100000b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
TR	Not used	0
TADR	Address of SSDO Client	SSADR

Table 20 specifies the fields of an Initiate Upload SSDO_Service_Response telegram.

Table 20 – Fields of Initiate Upload SSDO_Service_Response telegram

Field	Information	Content / Value
ADR	Address of SSDO Client to be responded	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Length of payload data	4 -- 254
CT	Echo of SOD Access Request number	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) expedited or Segmented	00100000b or 00101000b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
DB4 – DBn	Payload data (In case of a segmented upload, DB4-DB7 contains the data size of the payload data of the block. If the data size is 0, the size is not indicated.)	Payload data
TR	Not used	0
TADR	Address of SSDO Server	SSADR

7.3.7 SSDO Segmented Upload

Figure 29 specifies the Segmented Upload protocol. The SSDO client always requests a middle segment. Based on the response, the SSDO client shall decide whether a middle segment or the end segment was sent.

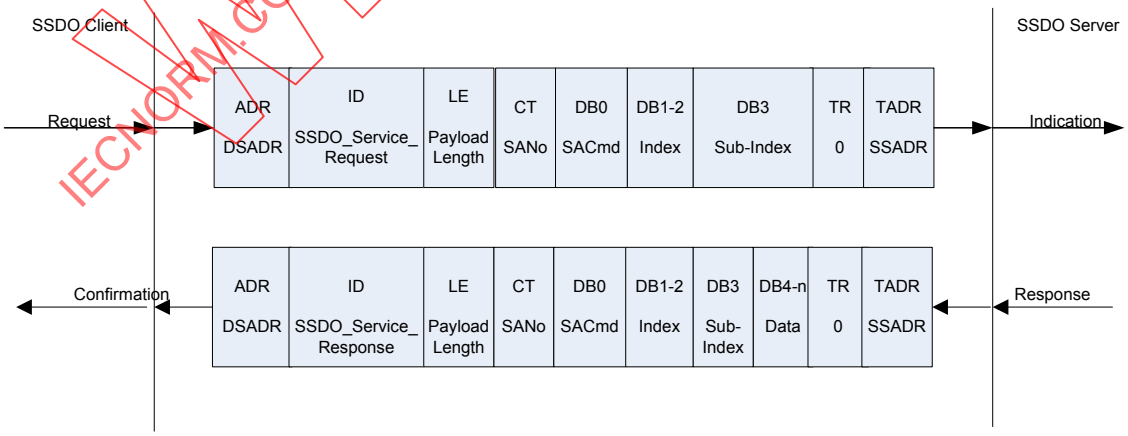


Figure 29 – SSDO Segmented Upload protocol

Table 21 specifies the fields of a Segmented Upload SSDO_Service_Request telegram.

Table 21 – Fields of Segmented Upload SSDO_Service_Request telegram

Field	Information	Content / Value
ADR	Address of SSDO Server to be accessed	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Length of payload data	4
CT	SOD Access Request number (SANo)	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) middle segment	000x1000b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
TR	Not used	0
TADR	Address of SSDO Client	SSADR

Table 22 specifies the fields of a Segmented Upload SSDO_Service_Response telegram.

Table 22 – Fields of Segmented Upload SSDO_Service_Response telegram

Field	Information	Content / Value
ADR	Address of SSDO Client to be responded	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Length of payload data	4 -- 254
CT	Echo of SOD Access Request number	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd) middle segment or end segment	000x1000b or 010x1000b
DB1, DB2	Index of SOD entry to be accessed	0 -- 65 535
DB3	Sub index of SOD entry to be accessed	0 -- 255
DB4 -- DBn	Payload data	Payload data
TR	Not used	0
TADR	Address of SSDO Server	SSADR

7.3.8 SSDO Abort

Figure 30 specifies the SSDO Abort protocol.

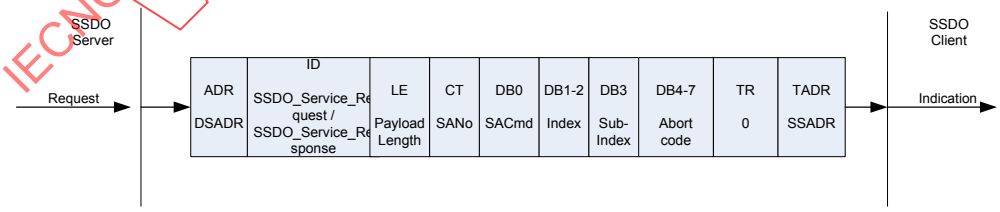


Figure 30 – SSDO Abort protocol

Table 23 specifies the fields of an Abort SSDO_Service_Request telegram.

Table 23 – Fields of Segmented Upload SSDO_Service_Request telegram

Field	Information	Content / Value
ADR	Address of the SSDO Client to be informed	DSADR
ID	SSDO_Service_Request	111000xxb

Field	Information	Content / Value
LE	Length of payload data	8
CT	SOD Access Request number (SANo)	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd)	00000100b
DB1, DB2	Index of SOD entry	0 -- 65 535
DB3	Sub index of SOD entry	0 -- 255
DB4 -- DB7	Abort code, see Table 25	Abort code
TR	Not used	0
TADR	Address of the SSDO Server initiating the abort	SSADR

Table 24 specifies the fields of an Abort SSDO_Service_Response telegram.

Table 24 – Fields of Segmented Upload SSDO_Service_Response telegram

Field	Information	Content / Value
ADR	Address of the SSDO Client to be informed	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Length of payload data	8
CT	SOD Access Request number (SANo)	1 -- 65 535, 0 not allowed
DB0	SOD Access Command (SACmd)	00000100b
DB1, DB2	Index of SOD entry	0 -- 65 535
DB3	Sub index of SOD entry	0 -- 255
DB4 -- DB7	Abort code, see Table 25	Abort code
TR	Not used	0
TADR	Address of the SSDO Server initiating the abort	SSADR

The payload data of the SSDO Abort telegram shall contain an error code as specified in Table 25. The abort code shall be encoded as UNSIGNED32 value. Abort codes not listed in Table 25 shall be reserved.

Table 25 – SSDO Abort codes

Abort code [a]	Description
05030000h	Reserved
05040000h	SSDO protocol timed out
05040001h	Client/server Command ID not valid or unknown
05040002h	Invalid block size
05040003h	Invalid sequence number
05040004h	Reserved
05040005h	Out of memory
06010000h	Unsupported access to an object
06010001h	Attempt to read a write-only object
06010002h	Attempt to write a read-only object
06020000h	Object does not exist in the object dictionary
06040041h	Object cannot be mapped to the SPDO
06040042h	The number and length of the objects to be mapped would exceed SPDO length
06040043h	General parameter incompatibility
06040047h	General internal incompatibility in the device
06060000h	Access failed due to a hardware error
06070010h	Data type does not match, length of service parameter does not match
06070012h	Data type does not match, length of service parameter too high
06070013h	Data type does not match, length of service parameter too low
06090011h	Sub index does not exist

Abort code [a]	Description
06090030h	Value range of parameter exceeded (only for write access)
06090031h	Value of parameter written too high
06090032h	Value of parameter written too high
06090036h	Value of parameter written too high
08000000h	Value of parameter written too high
08000020h	Data cannot be transferred or stored to the application
08000021h	Data cannot be transferred or stored to the application because of local control
08000022h	Data cannot be transferred or stored to the application because of the present device state
08000023h	Data cannot be transferred or stored to the application because application is busy
[a] The Abort code is shown in hexadecimal notation, as designated by the "h" suffix.	

7.4 Safety Network Management (SNMT)

7.4.1 General

The Safety Network Management (SNMT) is node oriented and follows a logical Master/Slave structure. The SNMT Master shall be implemented by an SCM or a configuration tool. The following SNMT services shall be provided:

- UDID request,
- node address assignment,
- extended Services, which shall be designated by the DB0 field and comprise the following services:
  - communication state control,
  - node guarding,
  - additional node address assignment.

7.4.2 SNMT telegram types

The ID field (bits 2 -- 4) shall identify the SNMT telegram type. Table 26 specifies the SNMT telegram types:

Table 26 – SNMT telegram types (ID field, bits 2, 3 and 4)

Bit number of ID field								Telegram type	Service
7	6	5	4	3	2	1	0		
1	0	1	0	0	0			SNMT_Request_UDID	UDID request
1	0	1	0	0	1			SNMT_Response_UDID	
1	0	1	0	1	0			SNMT_Assign_SADR	Node address assignment
1	0	1	0	1	1			SNMT_SADR_assigned	
1	0	1	1	0	0			SNMT_Service_Request	Extended Services
1	0	1	1	0	1			SNMT_Service_Response	
1	0	1	1	1	1			SNMT_SN_reset_guarding_SCM	Reset guarding timeout

7.4.3 SNMT services and protocols

7.4.3.1 UDID Request / Response

The UDID Request / Response shall enable an SNMT Master to retrieve the UDID of a specific SNMT Slave. For this, the SNMT Master shall send the UDID Request to one specific SNMT Slave. The SNMT Slave shall reply, even if the value within the ADR field does not match the SNs internal SADR.

NOTE 1 This may happen if there was no prior SADR assignment to the SN.

The TADR field of the response shall contain the SADR of the UDID Request.

NOTE 2 If the SNMT slave would respond with its own (possibly invalid SADR), the SNMT master may not be able to match the response to a request.

An error while executing this protocol shall be indicated by UDID 00-00-00-00-00-00 in the response.

Figure 31 specifies the UDID Request protocol.

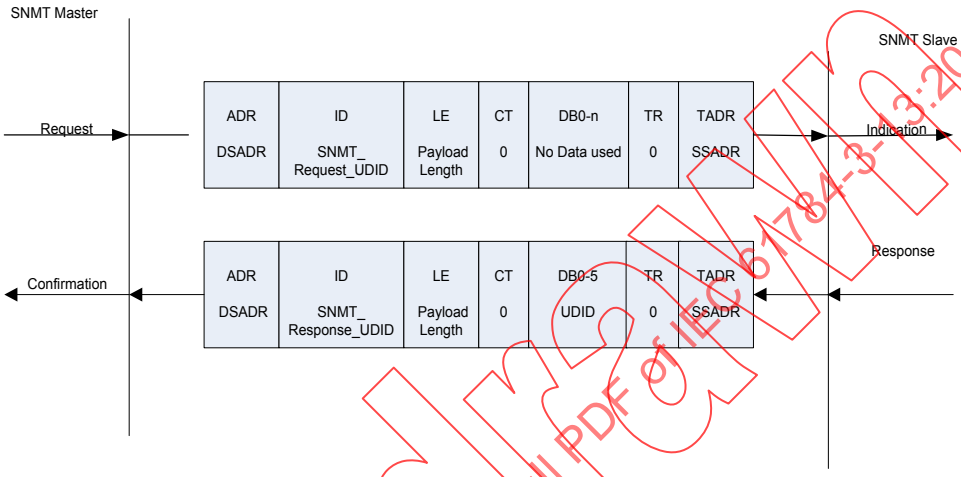


Figure 31 – UDID Request / Response protocol

Table 27 specifies the fields of an SNMT_Request_UDID telegram.

Table 27 – Fields of SNMT_Request_UDID telegram

Field	Information	Content / Value
ADR	Address of the SNMT Slave to be addressed	DSADR
ID	SNMT_Request_UDID	101000xxb
LE	Length of payload data	0
CT	Not used	0
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 28 specifies the fields of an SNMT_Response_UDID telegram.

Table 28 – Fields of SNMT_Response_UDID telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT_Response_UDID	101001xxb
LE	Length of payload data	6
CT	Not used	0
DB0 -- DB5	UDID of the SN	UDID
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

7.4.3.2 SADR assignment

The SADR Assignment shall be used by an SNMT Master to set the SADR of a specific SNMT Slave. The protocol further shall assign the safety domain number (SDN) to the SNMT Slave. The SDN shall be set inside the address field of the PDU part two using a logical XOR operation (see 7.1). A SN shall send a response only if the received UDID matches the own UDID.

If the SDN/SADR requested by the SNMT Master can not be assigned, the SNMT Slave shall reply its current SDN/SADR. The SNMT Master shall then compare the replied SDN/SADR and UDID with the set SDN/SADR/UDID to check whether the assignment was successful.

NOTE Unsuccessful assignment can occur if the nodes SDN/SADR are hard coded using a switch and do not match the SCM value.

An error while executing the service shall be indicated whit UDID 00-00-00-00-00-00 in the response.

Figure 32 specifies the SADR Assignment protocol.

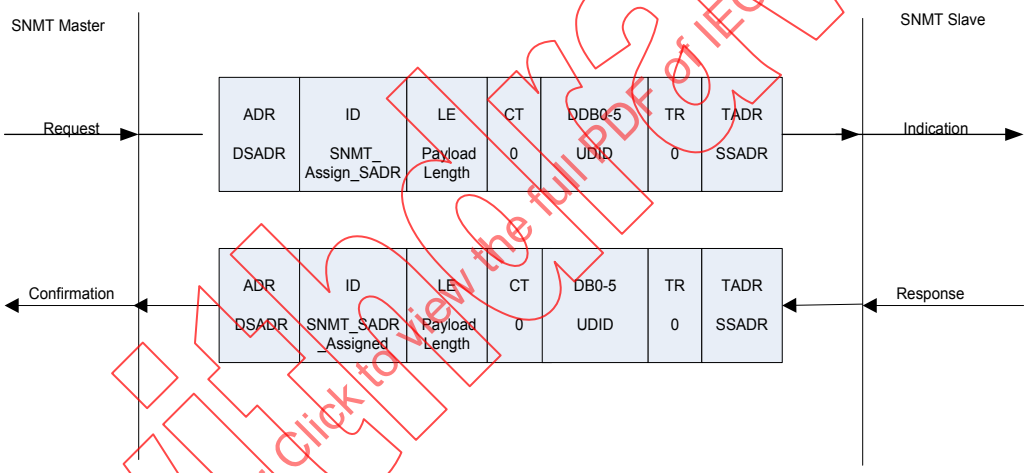


Figure 32 – SADR Assignment protocol

Table 29 specifies the fields of an SNMT_Assign_SADR telegram.

Table 29 – Fields of SNMT_Assign_SADR telegram

Field	Information	Content / Value
ADR	Address of the SNMT Slave to be assigned	DSADR
ID	SNMT_Assign_SADR	101010xxb
LE	Length of payload data	6
CT	Not used	0
DB0 -- DB5	UDID of the SN	UDID
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 30 specifies the fields of an SNMT_SADR_ Assigned telegram.

Table 30 – Fields of SNMT_SADR_ Assigned telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT_SADR_ Assigned	101011xxb
LE	Length of payload data	6
CT	Not used	0
DB0 -- DB5	UDID of the SN	UDID
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

7.4.3.3 Reset Node Guarding Time

The service Reset Node Guarding Time shall be used by the SN to trigger a node guarding on the SCM. The SCM shall guard all SNs whether the SADR or SDN in the telegram are match the SADR and SDN of the SCM or not.

The telegram shall always be sent with SADR and SDN being 1.

NOTE This is due to the fact that the telegram is sent before an SADR or SDN is assigned to the node.

Figure 33 specifies the Reset Node Guarding Time protocol.

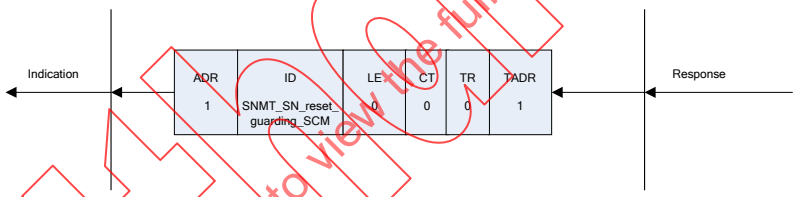


Figure 33 – Reset Node Guarding Time protocol

Table 31 specifies the fields of an SNMT_SN_reset_guarding_SCM telegram.

Table 31 – Fields of SNMT_SN_reset_guarding_SCM telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	1
ID	SNMT_SN_reset_guarding_SCM	$101111xx_b$
LE	Length of payload data	0
CT	Not used	0
TR	Not used	0
TADR	Address of SNMT Slave	1

7.4.3.4 SNMT Extended Services

SNMT Extended Services is specified by the ID field and the DB0 field of the FSPC 13/1 PDU (see Table 32 and Table 33).

Table 32 – SNMT request telegram types

Bit number of DB0								SNMT telegram type
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	SNMT_SN_set_to_PRE_OP
0	0	0	0	0	0	1	0	SNMT_SN_set_to_OP
0	0	0	0	0	1	0	0	SNMT_SCM_set_to_STOP
0	0	0	0	0	1	1	0	SNMT_SCM_set_to_OP
0	0	0	0	1	0	0	0	SNMT_SCM_guard_SN
0	0	0	0	1	0	1	0	SNMT_assign_additional_SADR
0	0	0	0	1	1	0	0	SNMT_SN_ACK
0	0	0	0	1	1	1	0	SNMT_assign_UDID_SCM

Table 33 – SNMT response telegram types

Bit number of DB0								SNMT telegram type
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	1	SNMT_SN_status_PRE_OP
0	0	0	0	0	0	1	1	SNMT_SN_status_OP
0	0	0	0	0	1	0	1	SNMT_assigned_additional_SADR
0	0	0	0	0	1	1	1	SNMT_SN_FAIL
0	0	0	0	1	0	0	1	SNMT_SN_busy
0	0	0	0	1	1	1	1	SNMT_assigned_UDID_SCM

7.4.3.5 SN set to Pre-Operational

The SN set to Pre-Operational service shall be used to switch an SNMT Slave from the Operational to the Pre-Operational communication state.

Figure 34 specifies the SN set to Pre-Operational protocol.

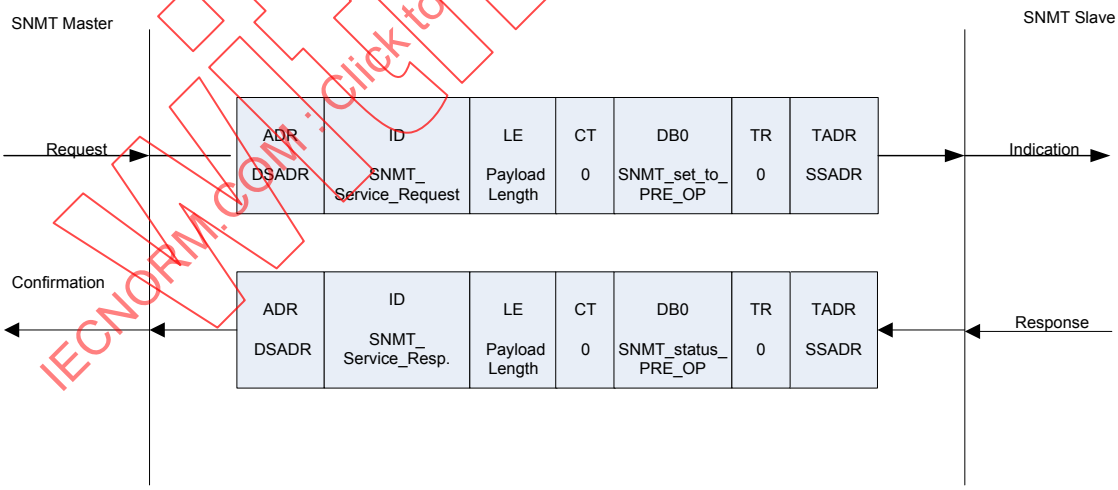


Figure 34 – SN set to Pre-Operational protocol

Table 34 specifies the fields of an SNMT_SN_set_to_PRE_OP telegram.

Table 34 – Fields of SNMT_SN_set_to_PRE_OP telegram

Field	Information	Content / Value
ADR	Address of the SNMT Slave to be accessed	DSADR
ID	SNMT Service Request	101100xxb

Field	Information	Content / Value
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SN_set_to_PRE_OP	00000000b
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 35 specifies the fields of an SNMT_SN_status_PRE_OP telegram.

Table 35 – Fields of SNMT_SN_status_PRE_OP telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SN_status_PRE_OP	00000001b
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

7.4.3.6 SN set to Operational

The service SN set to Operational shall be used to switch an SNMT Slave from Pre-Operational to Operational communication state. The response shall either be the information that the SN has entered the Operational state, or error information which the SCM reports to its application.

Figure 35 specifies the SN set to Operational protocol.

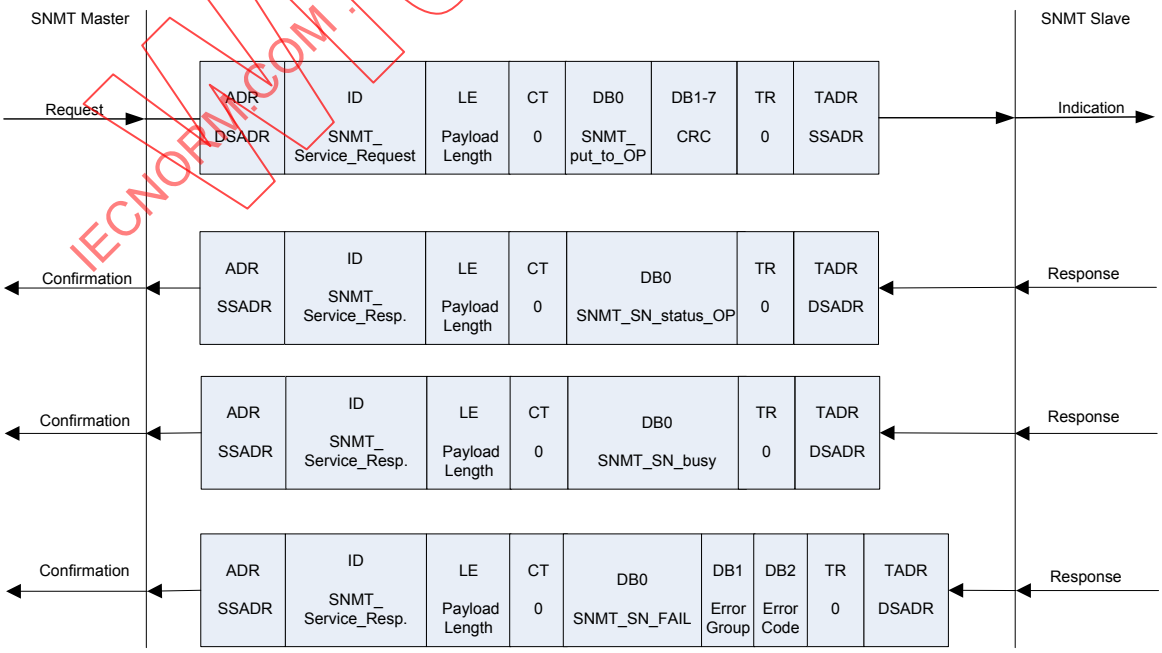


Figure 35 – SN set to Operational protocol

Table 36 specifies the fields of an SNMT_SN_set_to_OP telegram.

Table 36 – Fields of SNMT_SN_set_to_OP telegram

Field	Information	Content / Value
ADR	Address of SNMT Slave to be accessed	DSADR
ID	SNMT Service Request	101100xxb
LE	Length of payload data	7
CT	Not used	0
DB0	SNMT_SN_set_to_OP	00000010b
DB1 -- DB2	Parameter checksum	See 7.5.4.7
DB3 -- DB6	Parameter timestamp	See 7.5.4.7
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 37 specifies the fields of an SNMT_SN_status_OP telegram.

Table 37 – Fields of SNMT_SN_status_OP telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SN_status_OP	00000011b
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

Table 38 specifies the fields of an SNMT_SN_busy telegram.

Table 38 – Fields of SNMT_SN_busy telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SN_busy[a]	00001001b
TR	Not used	0
TADR	Address of SNMT Slave	SSADR
[a] SNMT_SN_busy shall be replied by the application (e.g. during calculation of the CRC or saving parameters).		

Table 39 specifies the fields of an SNMT_SN_FAIL telegram.

Table 39 – Fields of SNMT_SN_FAIL telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	3
CT	Not used	0
DB0	SNMT_SN_FAIL	00000111b
DB1	Error Group (see Table 40)	0 -- 255
DB2	Error Code (see Table 41)	0 -- 255
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

Table 40 specifies SNMT_SN_FAIL Error Group values.

Table 40 – SNMT_SN_FAIL Error Group values

Value	Error Group
0	Device
1	Application
2	Parameter
3	Vendor specific
4	FSCP 13/1 Stack
5 -- 255	Reserved for future use

Table 41 specifies SNMT_SN_FAIL Error Code values.

Table 41 – SNMT_SN_FAIL Error Code values

Value	Error Group
0	Default
1 -- 255	Vendor specific

7.4.3.7 SNMT SN Acknowledge

The service SN Acknowledge shall be used by the SCM to acknowledge a reported error (see 7.4.3.6) at the SN.

Figure 36 specifies the SN Acknowledge protocol.

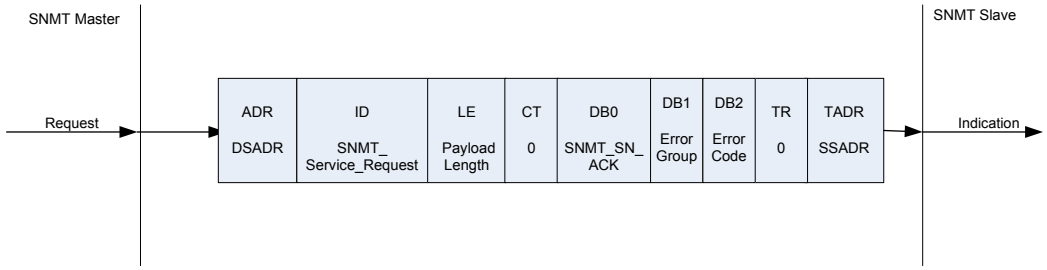


Figure 36 – SN Acknowledge protocol

Table 42 specifies the fields of an SNMT_SN_ACK telegram.

Table 42 – Fields of SNMT_SN_ACK telegram

Field	Information	Content / Value
ADR	Address of SNMT Slave to be accessed	DSADR
ID	SNMT Service Request	101100xxb
LE	Length of payload data	3
CT	Not used	0
DB0	SNMT_SN_ACK	00001100b
DB1	Error Group (see Table 40)	0 -- 255
DB2	Error Code (see Table 41)	0 -- 255
TR	Not used	0
TADR	Address of SNMT Master	SSADR

7.4.3.8 SCM set to stop

The SCM set to stop service shall be used to switch an SCM off (from Operational to Stopped communication state) for system configuration by means of an external tool.

Figure 37 specifies the SCM set to stop protocol.

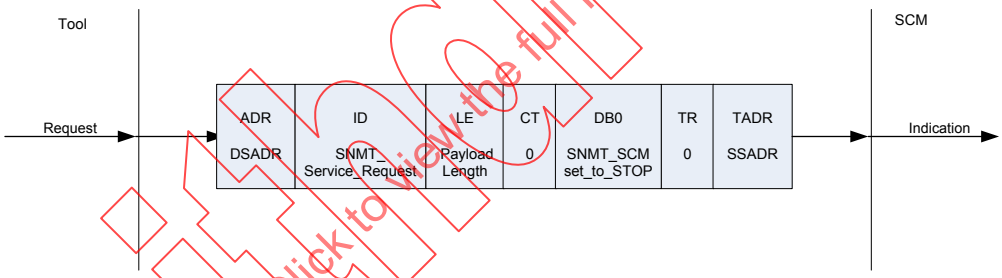


Figure 37 – SN set to stop protocol

Table 43 specifies the fields of an SNMT_SCM_set_to_STOP telegram.

Table 43 – Fields of SNMT_SCM_set_to_STOP telegram

Field	Information	Content / Value
ADR	Address of the SCM to be accessed	DSADR
ID	SNMT Service Request	101100xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SCM_set_to_STOP	00000100b
TR	Not used	0
TADR	Address of the TOOL	SSADR

7.4.3.9 SCM set to Operational

The SCM set to Operational service shall be used to switch the SCM from Stopped to Operational communication state. This service may be applied after completion of the configuration by means of an external tool.

Figure 38 specifies the SCM set to Operational protocol.

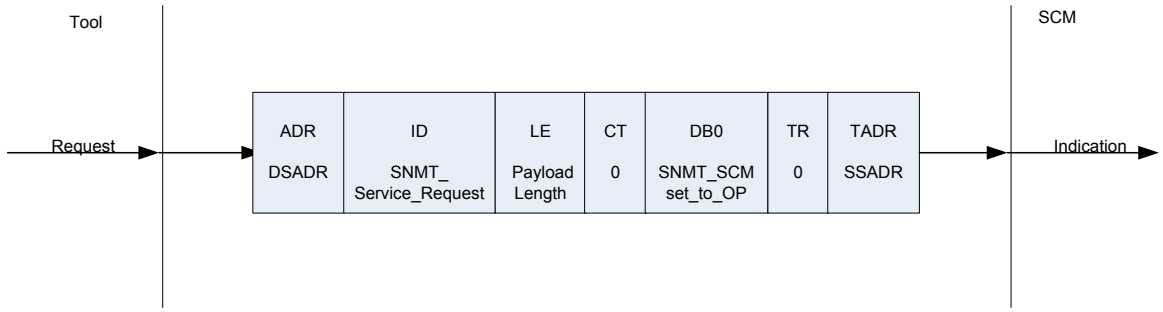


Figure 38 – SCM set to Operational protocol

Table 44 specifies the fields of an SNMT_SCM_set_to_OP telegram.

Table 44 – Fields of SNMT_SCM_set_to_OP telegram

Field	Information	Content / Value
ADR	Address of the SCM to be accessed	DSADR
ID	SNMT_Service_Request	101100xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SCM_set_to_OP	00000110b
TR	Not used	0
TADR	Address of the TOOL	SSADR

7.4.3.10 Node Guarding

The Node Guarding service shall be used to guard an SN in Operational communication state. The SN shall respond with its status. If an SN does not receive an SNMT_SCM_guard_SN telegram within its life time (see 7.5.4.5), it shall switch back to Pre-Operational. Signaling a lost request or response to the application shall be vendor specific.

Figure 39 specifies the Node Guarding protocol.

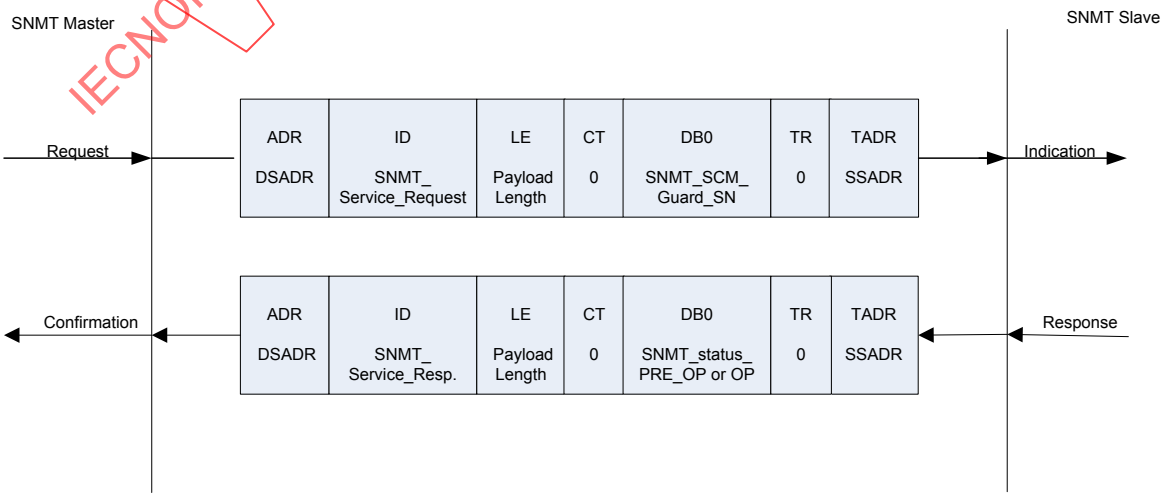


Figure 39 – Node Guarding protocol

Table 45 specifies the fields of an SNMT_SCM_guard_SN telegram.

Table 45 – Fields of SNMT_SCM_guard_SN telegram

Field	Information	Content / Value
ADR	Address of SNMT Slave to be accessed	DSADR
ID	SNMT Service Request	101100xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SCM_guard_SN	00001000b
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 46 specifies the fields of SNMT_SN_status_PRE_OP / SNMT_SN_status_OP telegrams.

Table 46 – Fields of SNMT_SN_status_OP/SNMT_SN_status_OP telegrams

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	1
CT	Not used	0
DB0	SNMT_SN_status_PRE_OP or SNMT_SN_status_OP	00000001b or 00000011b
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

7.4.3.11 Additional SADR Assignment

The Additional SADR Assignment service shall be used to assign an additional SADR to one SNMT Slave. This service shall only be used if an SN does support more than one TxSPDO (see 7.5.4.16).

If the SNMT Slave does not support more than one TxSPDO, the SNMT Slave shall reply with 0 as the SADR value.

Figure 40 specifies the Additional SADR Assignment protocol.

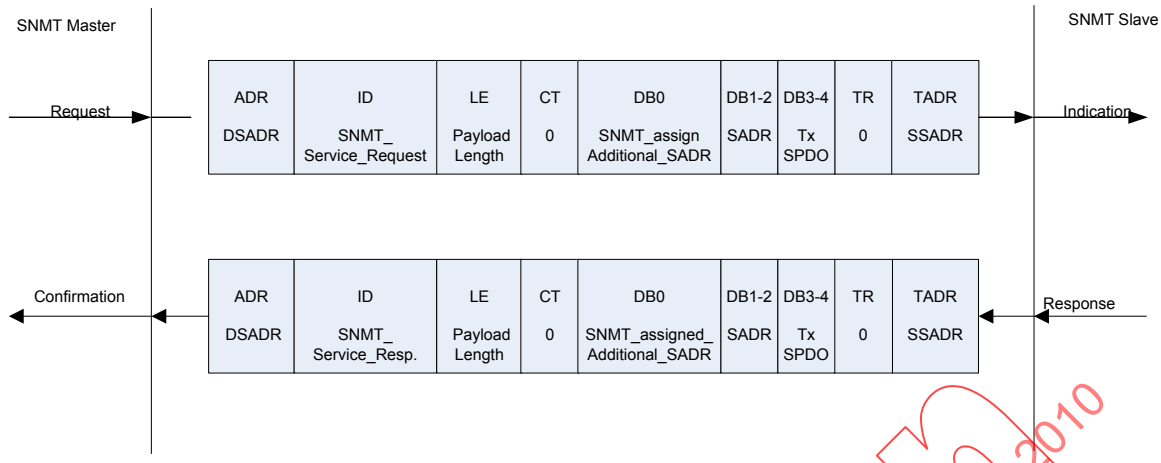


Figure 40 – Additional SADR Assignment protocol

Table 47 specifies the fields of an SNMT_assign_additional_SADR telegram.

Table 47 – Fields of SNMT_assign_additional_SADR telegram

Field	Information	Content / Value
ADR	Address of SNMT Slave to be accessed	DSADR
ID	SNMT Service Request	101100xxb
LE	Length of payload data	5
CT	Not used	0
DB0	SNMT_assign_additional_SADR	00001010b
DB1,DB2	Additional SADR to be assigned	SADR
DB3,DB4	TxSPDO to which additional SADR is assigned	2 -- 1 023
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 48 specifies the fields of an SNMT_assigned_additional_SADR telegram.

Table 48 – Fields of SNMT_assigned_additional_SADR telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	5
CT	Not used	0
DB0	SNMT_assigned_additional_SADR	00000101b
DB1, DB2	SADR that was assigned	SADR
DB3, DB4	TxSPDO to which SADR was assigned	2 -- 1 023
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

7.4.3.12 UDID of SCM Assignment

The UDID of SCM Assignment service shall be used to assign the UDID of the SCM to the SNMT Slave.

An error while executing the service shall be indicated with UDID value 00-00-00-00-00-00 in the response.

Figure 41 specifies the UDID of SCM Assignment protocol.

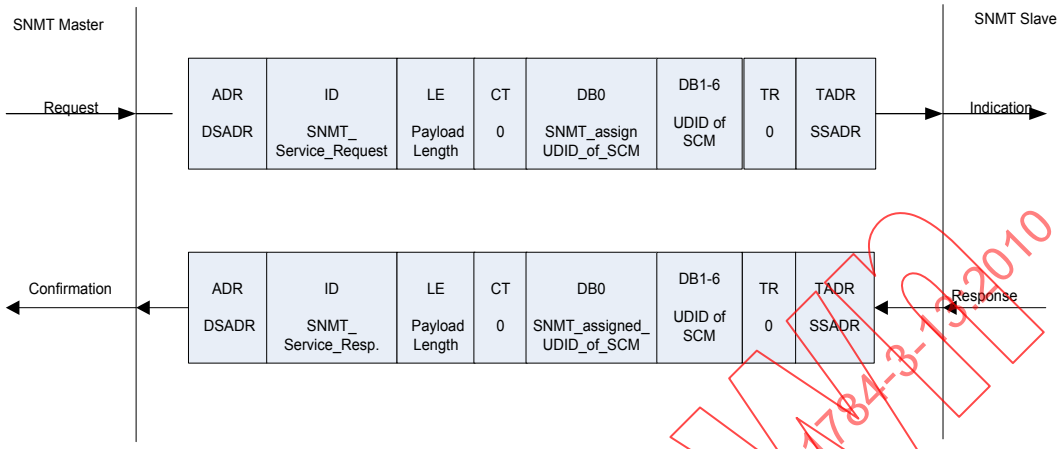


Figure 41 – UDID of SCM Assignment protocol

Table 49 specifies the fields of an SNMT_assign_UDID_of_SCM telegram.

Table 49 – Fields of SNMT_assign_UDID_of_SCM telegram

Field	Information	Content / Value
ADR	Address of SNMT Slave to be accessed	DSADR
ID	SNMT Service Request	101100xxb
LE	Length of payload data	7
CT	Not used	0
DB0	SNMT_assign_UDID_of_SCM	00001110b
DB1 -- DB6	UDID of the SCM	UDID
TR	Not used	0
TADR	Address of SNMT Master	SSADR

Table 50 specifies the fields of an SNMT_assigned_UDID_of_SCM telegram.

Table 50 – Fields of SNMT_assigned_UDID_of_SCM telegram

Field	Information	Content / Value
ADR	Address of SNMT Master	DSADR
ID	SNMT Service Response	101101xxb
LE	Length of payload data	7
CT	Not used	0
DB0	SNMT_assigned_UDID_of_SCM	00001111b
DB1 -- DB6	UDID of the SCM	UDID
TR	Not used	0
TADR	Address of SNMT Slave	SSADR

7.5 Safety Object dictionary (SOD)

7.5.1 General

The following subclasses specify the Safety Object Dictionary structure and entries which shall be common for all devices.

7.5.2 Object dictionary entry definition

7.5.2.1 General

An Object Dictionary entry shall be defined by the following items:

- index,
- object type,
- name,
- data type,
- category,
- access,
- value range,
- default value,
- SPDO mapping.

7.5.2.2 Index

Index shall denote the objects position within the Object Dictionary. This acts as a kind of address to reference the desired data field. An index shall be declared as hexadecimal value.

An Object may be subdivided to sub indices. The sub index shall be used to reference data fields within a compound object such as an array or record. The sub index is not specified here.

7.5.2.3 Object type

Object type shall denote what kind of object is at that particular index within the Object Dictionary. Object types shall be as specified in Table 51.

Table 51 – Object type definition

Object type	Comments	Code
NULL	A dictionary entry with no data fields	0
DOMAIN	Large variable amount of data e.g. executable program code	2
DEFTYPE	Denotes a basic data type definition (e.g. BOOLEAN, UNSIGNED16, REAL32)	5
DEFSTRUCT	Defines a record type	6
VAR	A single value (e.g. UNSIGNED8, BOOLENA, REAL32, INTEGER16, VISIBLE STRING)	7
ARRAY	A multiple data field object where each data field is a simple variable of the SAME basic data type e.g. array of UNSIGNED16 etc. Sub index 0 is of UNSIGNED8 and therefore not part of the ARRAY data	8
RECORD	A multiple data field object where the data fields may be any combination of simple variables. Sub index 0 is an UNSIGNED8 and therefore not part of the RECORD data	9

7.5.2.4 Name

Name shall provide a simple textual description of the function of that particular object. Name shall comply with IEC 61131-3. A name shall consist of:

— domain prefix, indicating the association of the object to a functional domain, up to 4 uppercase characters followed by underline,

— name (verbally) composed of words, each word shall be leaded by an uppercase character followed by lowercase characters or digits, no underlines or spaces,

— data type postfix, indicating the data type of the object (underline followed by up to 5 uppercase characters or digits).

Total length of name shall not exceed 32 characters.

7.5.2.5 Data type

Data type shall provide information about the data type of the object. It includes the following pre-defined basic data types: BOOLEAN, floating point number, UNSIGNED Integer, Signed Integer, visible/octet string and DOMAIN. It also includes the pre-defined compound data types and may also include types which are either manufacturer or device specific.

The following shall not be defined:

— records of records,

— arrays of records, or

— records with arrays as fields of that record.

In the case where an object is an array or a record, the sub index shall be used to reference one basic data type data field within the object.

Data type shall be a basic data type for ARRAY type objects and a compound data type for RECORD type objects.

7.5.2.6 Category

Category defines whether the object is Mandatory (M) or Optional (O). A mandatory object shall be implemented on a device. An optional object may be implemented on a device. The support of certain objects or features however may require the implementation of related objects. In this case, the relations are described in the detailed object specification.

Category may be Conditional (Cond) if the M/O category of an object depends on the implementation of another object.

Category shall refer to a compound data type object as a whole, but not to the particular sub index. A mandatory object may be composed of mandatory and optional sub indices. This means that the object shall be supported but some of its sub indices are optional. Defining optional objects with mandatory sub indices is also permitted. This means that these sub indices shall be supported if the object is implemented.

7.5.2.7 Access

Access defines the access rights for a particular object. The view point is from the outside into the device.

Access rights are specified in Table 52.

Table 52 – Access attributes for data objects

Access right	Description
rw	read and write access
wo	write only access
ro	read only access
Const	read only access, value is constant

Optional objects shall be listed in the Object Dictionary with the attribute rw may be implemented as read only. Exceptions are defined in the detailed object specification.

7.5.2.8 Value range

This entry shall indicate the value range of the respective object. It may consist of one or more distinct values or ranges of values. If the item shown is a data type identifier, the complete value range of the mentioned data type shall be allowed.

7.5.2.9 Default value

This entry shall indicate the initialization value that shall be assigned by the profile implementation. For mapable objects, this value shall also represent the safe state value.

7.5.2.10 SPDO mapping

This entry indicates whether an entry may be mapped to an SPDO message.

SPDO mapping values shall be as specified in Table 53.

Table 53 – SPDO mapping attributes for data objects

Value	Description
Opt	Object may be mapped into an SPDO
No	Object shall not be mapped into an SPDO

7.5.2.11 Example basic data type object definition

A complete basic data type object definition (Object Type = VAR or DOMAIN) example is shown in Table 54.

Table 54 – Basic data type object definition example

Item	Value
Index	1234h
Name	SSDO_VarDummyParameter_S16
Object type	VAR
Data type	INTEGER16
Value range	-15 000 – 10 000
Default value	0
Category	M
Access	rw
SPDO mapping	Opt

7.5.2.12 Example compound data type object definition

Compound data type object definitions (Object Type = ARRAY or RECORD) are of reduced form, because Access, Value Range, Default Value, and SPDO mapping shall be defined individually for the sub indices.

Table 55 gives a compound data type object definition example.

Table 55 – Compound data type object definition example

Item	Value
Index	2345h
Name	SSDO_ArrayDummyParameter_AU16
Object type	ARRAY
Data type	UNSIGNED16
Category	M

7.5.2.13 Sub index definition

7.5.2.13.1 General

Compound object types (ARRAY, RECORD objects) shall be composed of up to 256 data items. Each data item shall be addressed by an UNSIGNED8 type sub index.

Sub indices shall be interpreted as specified in Table 56.

Table 56 – Sub index interpretation

Sub index	Name
00h	NumberOfEntries
01h -- FDh	Object Specific Data
FEh	ChecksumOfObject
FFh	StructureOfObject

7.5.2.13.2 Sub index 00h – NumberOfEntries

NumberOfEntries shall describe the highest available sub index, not considering FEh and FFh. This entry shall be encoded as UNSIGNED8, regardless of the type of the object. If the object exists, NumberOfEntries shall be mandatory. Data Type and Category shall not be denoted in NumberOfEntries descriptions.

In this part, NumberOfEntries is specified in the form as shown in Table 57. Elements depicted by <...> shall be set according the specific context.

Table 57 – NumberOfEntries sub index specification

Item	Value
Sub index	00h
Name	NumberOfEntries
Value range	<valid values, for example 1 -- 15>
Default value	<default value, for example 15>
Access	<Access, for example rw>
SPDO mapping	No

For ARRAYs, NumberOfEntries shall express the number of occupied items in a list (Access shall be rw). For RECORDs, NumberOfEntries shall be constant (Access shall be ro or Const).

7.5.2.13.3 Sub index 01h - FDh – Object specific data

Sub indices between 01h and FDh shall hold the object's payload data. The highest accessible index shall be given by NumberOfEntries.

In this part, Object Specific Data of RECORD type objects are specified in the form as shown in Table 58. Elements depicted by <…> shall be set according the specific context.

Table 58 – RECORD type object sub index specification

Item	Value
Sub index	<sub index in hexadecimal notation, for example 01h>
Name	<name>
Object type	
Data type	<data type, for example UNSIGNED8>
Value range	<valid values, for example 1 -- 255>
Default value	<default value, for example 1>
Category	<mandatory or optional, for example M>
Access	<Access, for example rw>
SPDO mapping	<mapping possible, for example Opt>

Name shall be composed of a descriptive text followed by a data type postfix. Subclause 7.5.2.4 shall apply, with the single exception that the domain prefix is omitted.

If a name entry is defined in a data type definition, the sub index name shall be equal to it.

Category shall refer to the respective sub index only. Mandatory (M) shall denote that the sub index shall be implemented if the object is implemented. A mandatory sub index shall not force the parent object to be mandatory.

In this part, Object Specific Data of ARRAY type objects are specified in the form as shown in Table 59. Elements depicted by <…> shall be set according the specific context.

Table 59 – ARRAY type object sub index specification

Item	Value
Sub index	<sub index in hexadecimal notation, for example 01h>
Name	<name>
Value range	<data type, for example UNSIGNED16>
Default value	<default value, for example 0>
Category	<mandatory or optional, for example M>
Access	<Access, for example rw>
SPDO mapping	<mapping possible, for example Opt>

Name shall be a descriptive text. Subclause 7.5.2.4 shall apply, with the exceptions that, the domain prefix and data type postfix are omitted. If a name entry is defined in a data type definition, the sub index name shall be equal to it.

Category shall refer to the respective sub index only. Mandatory (M) shall denote that the sub index shall be implemented if the object is implemented. A mandatory sub index shall not force the parent object to be mandatory.

NOTE Despite the functional equivalence of all sub indices in an array, more than one sub index entry can be defined to a particular object to show differences of the Category, Access and / or SPDO Mapping options of the sub indices.

7.5.2.13.4 Sub index FEh – ChecksumOfObject

Sub index FEh shall hold the checksum of the data within the parent object. Regardless of the type of parent object, it shall be a CRC-8 and encoded as UNSIGNED8. For building the CRC-8, all writeable data from sub index 00h-FDh shall be used (read only data shall not be taken into consideration because it cannot get changed by the SCM). The polynomial for calculating the CRC-8 shall be the same as for calculating the CRC-8 of the safety PDU part one and two (see 7.1.7).

Objects which are not relevant for the SOD checksum calculation shall set sub index FEh to zero.

EXAMPLE The index 1001h is not relevant for the SOD checksum because it does not contain any parameters which are to be set by the SCM.

7.5.2.13.5 Sub index FFh – StructureOfObject

Sub index FFh may denote the structure of the object by providing the data type and the object type. Regardless of the type of the object, it is encoded as UNSIGNED32.

Encoding of StructureOfObject is specified by Table 60.

Table 60 – StructureOfObject encoding

	MSB [a]			LSB [b]
Bit-No.	31 -- 24	23 -- 16	15 -- 8	7 -- 0
Value	00h	Data Type (refer Table 61)		Object Type (refer Table 51)
		(High octet)	(Low octet)	
[a] Most significant octet.				
[b] Least significant octet.				

NOTE It is optional to support sub index FFh. If it is supported throughout the Object Dictionary and the structure of the compound data types is provided as well, it is possible to upload the entire structure of the Object Dictionary.

If StructureOfEntry is not supported, sub index FFh shall be reserved.

StructureOfObject is not shown in object definitions in this specification.

7.5.3 Data type entry specification

7.5.3.1 General

The basic data types shall be placed in the Object Dictionary for definition purposes only. However, indices in the range 0001h -- 0007h may be mapped in order to define the appropriate space of the RxSPDO (Dummy Mapping) as not being used by the device (do not care). The indices 0009h to 000Bh and 000Fh shall not be mapped to an SPDO.

Index range 0001h to 04FFh shall be used for data type specifying object dictionary entries.

Table 61 specifies the data types used in the SOD.

Table 61 – Object dictionary data types

Index	Object	Name
0001h	DEFTYPE	BOOLEAN
0002h	DEFTYPE	INTEGER8
0003h	DEFTYPE	INTEGER16
0004h	DEFTYPE	INTEGER32
0005h	DEFTYPE	UNSIGNED8
0006h	DEFTYPE	UNSIGNED16
0007h	DEFTYPE	UNSIGNED32
0008h	DEFTYPE	REAL32
0009h	DEFTYPE	VISIBLE_STRING
000Ah	DEFTYPE	OCTET_STRING
000Bh	DEFTYPE	UNICODE_STRING
000Ch	reserved	
000Dh	reserved	
000Eh	reserved	
000Fh	DEFTYPE	DOMAIN
0010h	DEFTYPE	INTEGER24
0011h	DEFTYPE	REAL64
0012h	DEFTYPE	INTEGER40
0013h	DEFTYPE	INTEGER48
0014h	DEFTYPE	INTEGER56
0015h	DEFTYPE	INTEGER64
0016h	DEFTYPE	UNSIGNED24
0017h	reserved	
0018h	DEFTYPE	UNSIGNED40
0019h	DEFTYPE	UNSIGNED48
001Ah	DEFTYPE	UNSIGNED56
001Bh	DEFTYPE	UNSIGNED64
001Ch -- 001Fh	reserved	
0020h	DEFSTRUCT	SSDO_LifeGuardRecord_TYPE
0021h	DEFSTRUCT	SINF_DevVendorInfoRecord_TYPE
0022h	DEFSTRUCT	SCOM_CommonCommParamRecord_TYPE
0023h	DEFSTRUCT	SSDO_CommParamRecord_TYPE
0024h	DEFSTRUCT	SNMT_CommParamRecord_TYPE
0025h	DEFSTRUCT	SPDO_RxCommParamRecord_TYPE
0026h	DEFSTRUCT	SPDO_TxCommParamRecord_TYPE
0027h	DEFSTRUCT	SSCM_SADRDVIListRecord_TYPE
0028h	DEFSTRUCT	SSCM_UDIDSADRListRecord_TYPE
0029h	DEFSTRUCT	SSCM_SpecificParametersRecord_TYPE
002Ah -- 003Fh	reserved	

7.5.3.2 Basic data types

For basic data types (Object Type = DEFTYPE) see 5.5.2.

A device may optionally provide the length of the basic data types encoded as UNSIGNED32 for read access of the index that refers to the data type.

EXAMPLE Index 0007h (UNSIGNED32) contains the value 20h=32d because the data type UNSIGNED32 is encoded using a bit sequence of 32 bit. If the length is variable (e.g. 000Fh = Domain), the entry contains 0h.

7.5.3.3 Compound data types

For the supported compound data types, a device may optionally provide the structure of that data type for read access at the corresponding data type index. Sub index 0 then provides the number of entries at this index not counting sub indices 0, 254 and 255; the following sub indices contain the data type according to Table 61 encoded as UNSIGNED16.

The entry at Index 0021h describing the structure of the Identity object SINF_DevVendorInfoRecord_TYPE is specified in Table 62 and Table 63.

Table 62 – 0021h Compound data type description

Item	Value
Index	0021h
Name	SINF_DevVendorInfoRecord_TYPE
Object type	DEFSTRUCT

Table 63 – 0021h Compound sub index descriptions

Sub index	Component name	Value	Data Type
00h	NumberOfEntries	06h	UNSIGNED16
01h	VendorId_U32	07h	UNSIGNED32
02h	ProductCode_U32	07h	UNSIGNED32
03h	RevisionNo_U32	07h	UNSIGNED32
04h	SerialNo_U32	07h	UNSIGNED32
05h	FirmWareChecksum_U32	07h	UNSIGNED32
06h	ParameterChecksum_U16	06h	UNSIGNED16
07h	ParameterTimestamp_U32	07h	UNSIGNED32

Standard (simple) and compound manufacturer specific data types shall be distinguished by attempting to read sub index 1h: For a compound data type, the device shall return a value and sub index 0h contains the number of sub indices that follow; For a standard data type, the device shall abort the SOD Access because no sub index 1h is available.

7.5.4 Object description

7.5.4.1 General

This and the following subclasses describe the standard object dictionary entries for an SN and SCM. The implementation of these objects may differ from the description within the specification.

EXAMPLE The Main SADR might be a read only parameter in case it is to be set using a dip switch.

Table 64 to Table 72 specify the FSCP 13/1 standard objects.

Table 64 – Standard objects

Index	Object type	Name	Data type	Acc	M/O
1001h	VAR	Error register	UNSIGNED8	ro	M
1002h	VAR	Manufacturer status register	UNSIGNED32	ro	O
1003h	ARRAY	Pre-defined error field	UNSIGNED32	rw	O
100Ch	RECORD	Life guarding	SSDO_LifeGuardRecord_TYPE (20h)	-	M
100Dh	VAR	Refresh Time for Reset Guarding	UNSIGNED32	rw	M
1018h	RECORD	Device vendor information	SINF_DevVendorInfoRecord_TYPE (21h)	-	M
1019h	VAR	Unique device id	OCTET_STRING	ro	M
101Ah	DOMAIN	Parameter download	DOMAIN	rw	O
101Bh	RECORD	SCM specific parameters	SSCM_SpecificParametersRecord_TYPE (29h)	-	O

Table 65 – Common communication objects

Index	Object type	Name	Data type	Acc	M/O
1200h	RECORD	common communication parameters	SCOM_CommonCommParamRecord_TYPE (22h)	-	M
1201h	RECORD	SSDO communication parameters	SSDO_CommParamRecord_TYPE (23h)	-	C[a]
1202h	RECORD	SNMT communication parameters	SNMT_CommParamRecord_TYPE (24h)	-	C

[a] The objects shall only be available if the node can be an SSDO client.

Table 66 – Receive SPDO communication objects

Index	Object type	Name	Data type	Acc	M/O
1400h	RECORD	1st RxSPDO communication parameters	SPDO_RxCommParamRecord_TYPE (25h)	rw	M[a]
1401h	RECORD	2nd RxSPDO communication parameters	SPDO_RxCommParamRecord_TYPE (25h)	rw	M[a]
:::	:::	:::	:::	:::	:::
17FEh	RECORD	1 023rd RxSPDO communication parameters	SPDO_RxCommParamRecord_TYPE (25h)	rw	M[a]

[a] If a device supports SPDOs, the corresponding SPDO communication parameter and SPDO mapping entries in the Object dictionary are mandatory.

Table 67 – Receive SPDO mapping objects

Index	Object type	Name	Data type	Acc	M/O
1800h	ARRAY	1st RxSPDO mapping parameters	UNSIGNED32	rw	M[a]
1801h	ARRAY	2nd RxSPDO mapping parameters	UNSIGNED32	rw	M[a]
:::	:::	:::	:::	:::	:::
1BFEh	ARRAY	1 023rd RxSPDO mapping parameters	UNSIGNED32	rw	M[a]
[a] If a device supports SPDOs, the corresponding SPDO communication parameter and SPDO mapping entries in the Object dictionary are mandatory.					

Table 68 – Transmit SPDO communication objects

Index	Object type	Name	Data type	Acc	M/O
1C00h	VAR	1st TxSPDO communication parameters	SPDO_TxCommParamRecord_TYPE (26h)	rw	M[a]
1C01h	VAR	2nd TxSPDO communication parameters	SPDO_TxCommParamRecord_TYPE (26h)	rw	M[a]
:::	:::	:::	:::	:::	:::
1FFEh	VAR	1 023rd TxSPDO communication parameters	SPDO_TxCommParamRecord_TYPE (26h)	rw	M[a]
[a] If a device supports SPDOs, the corresponding SPDO communication parameter and SPDO mapping entries in the Object dictionary are mandatory.					

Table 69 – Transmit SPDO mapping objects

Index	Object type	Name	Data type	Acc	M/O
C000h	ARRAY	1st TxSPDO mapping parameters	UNSIGNED32	rw	M[a]
C001h	ARRAY	2nd TxSPDO mapping parameters	UNSIGNED32	rw	M[a]
:::	:::	:::	:::	:::	:::
C3FEh	ARRAY	1 023rd TxSPDO mapping parameters	UNSIGNED32	rw	M[a]
[a] If a device supports SPDOs, the corresponding SPDO communication parameter and SPDO mapping entries in the Object dictionary are mandatory.					

Table 70 – SADR DVI list

Index	Object type	Name	Data type	Acc	M/O
C400h	RECORD	1st SADR-DVI List	SSCM_SADRDVIListRecord_TYPE (27h)	rw	C[a]
C401h	RECORD	2nd SADR-DVI List	SSCM_SADRDVIListRecord_TYPE (27h)	rw	C[a]
:::	:::	:::	:::	:::	:::
C7FEh	RECORD	1 023rd SADR-DVI List	SSCM_SADRDVIListRecord_TYPE (27h)	rw	C[a]
[a] Only needs to be implemented if the device has SCM functionality.					

Table 71 – Additional SADR list

Index	Object type	Name	Data type	Acc	M/O
C801h	ARRAY	1st SADR	UNSIGNED16	rw	C[a]
C802h	ARRAY	2nd SADR	UNSIGNED16	rw	C[a]
:::	:::	:::	:::	:::	:::
CBFFh	ARRAY	1 023rd SADR	UNSIGNED16	rw	C[a]
[a] Only needs to be implemented if the device has SCM functionality.					

Table 72 – SADR UDID list

Index	Object type	Name	Data type	Acc	M/O
CC01h	RECORD	1st SADR-UDID List	SSCM_UDIDSADRListRecord_TYPE (28h)	rw	C[a]
CC02h	RECORD	2nd SADR-UDID List	SSCM_UDIDSADRListRecord_TYPE (28h)	rw	C[a]
:::	:::	:::	:::	:::	:::
CFFFh	RECORD	1 023rd SADR-UDID List	SSCM_UDIDSADRListRecord_TYPE (28h)	rw	C[a]
[a] Only needs to be implemented if the device has SCM functionality.					

7.5.4.2 Object 1001h: Error Register

This object shall be an error register for the device. The device may map internal errors in this octet. If the value of the object is zero, the device shall be ok, all other values shall signal that the device is in an error state.

Object 1001h Error Register is specified in Table 73.

Table 73 – Object 1001h Error Register

Item	Value
Index	1001h
Name	SERR_GeneralError_U8
Object type	VAR
Data type	UNSIGNED8
Value range	UNSIGNED8
Default value	0
Category	M
Access	ro
SPDO mapping	Yes
Relevant for SOD checksum	No

The value interpretation for Object 1001h Error Register is specified in Table 74.

Table 74 – Object 1001h Error Register value interpretation

Bit	M/O	Description
0	M	generic error
1	O	current
2	O	voltage
3	O	temperature
4	O	communication error (overrun, error state)
5	O	device profile specific
6	O	Reserved (always 0)
7	O	manufacturer specific

7.5.4.3 Object 1002h: Manufacturer status register

This object shall be a status register for manufacturer purposes. In this part, only the size and location of this object shall be defined.

Object 1002h Manufacturer status register is specified in Table 75.

Table 75 – Object 1002h Manufacturer status register

Item	Value
Index	1002h
Name	SERR_ManufacturerStatus_U32
Object type	VAR
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	O
Access	ro
SPDO mapping	Yes
Relevant for SOD checksum	No

7.5.4.4 Object 1003h: Pre defined error field

The object at index 1003h shall store the errors that have occurred on the device. Every new error shall be stored at sub index 01h, older errors shall be moved to the next higher sub index. In doing so, it provides an error history. Sub index 0 shall provide the number of errors in the list. Setting sub index 0 to zero shall be used for deleting the list.

Object 1003h Pre defined error field is specified in Table 76 to Table 79.

Table 76 – Object 1003h Pre defined error field

Item	Value
Index	1003h
Name	SERR_ErrorHistory_AU32
Object type	ARRAY
Data type	UNSIGNED32
Category	O
Relevant for SOD checksum	No

Table 77 – Object 1003h sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	1 -- 253
Default value	1
Access	rw
SPDO mapping	No
[a] Number of errors.	

Table 78 – Object 1003h sub index 01h

Item	Value
Sub index	01h
Name	Error_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Error within the history list.	

Table 79 – Object 1003h sub index 02h to FDh

Item	Value
Sub index	02h -- FDh
Name	Error_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	O
Access	ro
SPDO mapping	No
[a] Error within the history list.	

7.5.4.5 Object 100Ch: Life Guarding

This object defines the Life Guarding protocol. The guard time shall be the time interval in which the SN should receive a life time signal from the SCM. The life time factor multiplied by the guard time results in the life time of the node. The time interval for sending the life time signal to the SNs shall be calculated by dividing the GuardTime_U32 by the number of nodes. The guard time interval for an SN within the SCM is started after receiving the last guarding response from the SN.

Object 100Ch Life Guarding is specified in Table 80 to Table 83.

Table 80 – Object 100Ch Life Guarding

Item	Value
Index	100Ch
Name	SSDO_LifeGuard_REC
Object type	RECORD
Data type	SSDO_LifeGuardRecord_TYPE
Category	M
Relevant for SOD checksum	Yes

Table 81 – Object 100Ch sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	2
Default value	2
Access	ro
SPDO mapping	No
[a] Number of Entries within this index.	

Table 82 – Object 100Ch sub index 01h

Item	Value
Sub index	01h
Name	GuardTime_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	10 000 000
Category	M
Access	rw
SPDO mapping	No
[a] Guard time for the lifetime protocol. The time base of this parameter is defined by CommonCommParam_REC. ConsecutiveTimeBase_I8.The values' range may be limited to a smaller value than 232. This limitation is vendor specific and/or due to implementation reasons.	

Table 83 – Object 100Ch sub index 02h

Item	Value
Sub index	02h
Name	LifeTimeFactor_U8
Data type	UNSIGNED8
Value range	1 -- 255
Default value	2
Category	M
Access	rw
SPDO mapping	No
[a] Number of times the GuardTime_U32 may elapse before the node switches back from Operational to Pre-Operational.	

7.5.4.6 Object 100Dh: Refresh Interval of Reset Guarding

This object contains the refresh interval of the Reset Guarding (see 7.7.6.3).

Object 100Dh Refresh Interval of Reset Guarding is specified in Table 84.

Table 84 – Object 100Dh Refresh Interval of Reset Guarding

Item	Value
Index	100Dh
Name	SNMT_ResetGuarding_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	100 000
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	No
[a] The sub index provides the refresh time for the SNMT_SN_reset_guarding_SCM service. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

7.5.4.7 Object 1018h: Device Vendor Information

This object shall contain the vendor specific device information. It shall also define compatible devices. Any device with an identical Vendor-ID, Product code, Major Revision Number and a Minor Revision Number greater or equal to another device shall be compatible. The manufacturer of a device shall verify and ensure the compatibility for minor revision changes on a safety related device.

Object 1018h Device Vendor Information is specified in Table 85 to Table 93.

Table 85 – Object 1018h Device Vendor Information

Item	Value
Index	1018h
Name	SINF_DevVendorInfo_REC
Object type	RECORD
Data type	SINF_DevVendorInfoRecord_TYPE
Category	M
Relevant for SOD checksum	No

Table 86 – Object 1018h sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	7
Default value	7
Access	ro
SPDO mapping	No
[a] Number of Entries within this index.	

Table 87 – Object 1018h sub index 01h

Item	Value
Sub index	01h
Name	VendorID_32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Vendor ID according EPSG Vendor List.	

Table 88 – Object 1018h sub index 02h

Item	Value
Sub index	02h
Name	ProductCode_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] The manufacturer-specific product code identifies a specific device version.	

Table 89 – Object 1018h sub index 03h

Item	Value
Sub index	03h
Name	RevisionNumber_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] See Table 94.	

Table 90 – Object 1018h sub index 04h

Item	Value
Sub index	04h
Name	SerialNumber_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Serial number of the device.	

Table 91 – Object 1018h sub index 05h

Item	Value
Sub index	05h
Name	FirmWareChecksum_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Checksum used to test the integrity of the firmware which is stored on the device. Calculation is vendor specific.	

Table 92 – Object 1018h sub index 06h

Item	Value
Sub index	06h
Name	ParameterChecksum_U16 [a]
Data type	UNSIGNED16
Value range	UNSIGNED16
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] This checksum is a CRC-16 (using the same polynomial as for the safety PDU parts one and two, see 7.1.7) which is built from the checksums of the available objects (sub index FEh). The checksum shall be verified in each device at start-up. Checksums from indices which are marked as not relevant for the SOD checksum are to be left out. If this parameter is read and the SN is in pre-Operational state, the checksum shall be recalculated prior to the response generation.	

Table 93 – Object 1018h sub index 07h

Item	Value
Sub index	07h
Name	ParameterTimestamp_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] The parameter timestamp is created by the configuration tool and is verified in each device at start-up by the SCM (see 7.7.12.2).	

The manufacturer-specific Revision Number shall consist of a major revision number and a minor revision number. The major revision number shall identify specific device behaviour. If the device functionality is expanded, the major revision shall be incremented. The minor revision number shall identify different versions with the same device behaviour.

Table 94 specifies the encoding of RevisionNumber_U32.

Table 94 – Structure of Revision Number

31	16	15	0
Major revision number		Minor revision number	
MSB			LSB

7.5.4.8 Object 1019h: Unique Device ID

The object at index 1019h shall contain the Unique Device Identification (UDID) for the device. The UDID shall be composed of the MAC Vendor ID and a vendor's unique device ID.

NOTE Since the uniqueness of the UDID of connected SNs is verified by the SCM, the allocation of the UDID during the vendor's production is not a safety critical production step.

The composition of the UDID shall be as following:

3 octets MAC Vendor Id + 3 octet vendor-specific unique device number (e.g. serial number).

EXAMPLE MAC Vendor ID: 06065h and Serial Number: 471112h gives UDID: 00-60-65-47-11-12.

UDID 00-00-00-00-00-00 shall be reserved.

Object 1019h Unique Device ID is specified in Table 95.

Table 95 – Object 1019h Unique Device ID

Item	Value
Index	1019h
Name	SINF_UniqueDeviceID_OSTR6 [a]
Object type	OCTET_STRING6
Data type	OCTET_STRING6
Value range	
Default value	0
Category	M
Access	Const
SPDO mapping	No
Relevant for SOD checksum	No
[a] The UDID shall be a globally unique identifier which consists of 6 octets.	

7.5.4.9 Object 101Ah: Parameter Download

This object shall be used for receiving an application specific parameter set from the SCM.

Object 101Ah Parameter Download is specified in Table 96.

Table 96 – Object 101Ah Parameter Download

Item	Value
Index	101Ah
Name	SSDO_ParameterDownload_DOM
Data type	DOMAIN
Value range	DOMAIN
Default value	0
Category	O
Access	rw
SPDO mapping	No
Relevant for SOD checksum	No

7.5.4.10 Object 101Bh: SCM Parameters

The object at index $101B_h$ shall contain the SCM specific parameters.

Object 101Bh SCM Parameters is specified in Table 97, Table 98 and Table 99.

Table 97 – Object 101Bh SCM Parameters

Item	Value
Index	101Bh
Name	SSCM_SpecificParameters_REC
Object type	RECORD
Data type	SSCM_SpecificParametersRecord_TYPE
Category	O
SPDO mapping	Yes

Table 98 – Object 101Bh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	1
Default value	1
Access	ro
SPDO mapping	No
[a] Number of Entries within this index.	

Table 99 – Object 101Bh sub index 01h

Item	Value
Sub index	01h
Name	ConfigurationMode_U8
Data type	UNSIGNED8
Value range	0 -- 1
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] SCM configuration mode according to 6.2.2, 0 = ACM, 1 = MCM.	

7.5.4.11 Object 1200h: Common Communication Parameter

This object shall contain the common communication parameters for the device.

Sub index 3 shall contain the consecutive time base for the module. It is not necessary for a device to support all possible time bases. The minimum requirement shall support a time base of 100 µs (type 2).

Object 1200h Common Communication Parameter is specified in Table 100 to Table 105.

Table 100 – Object 1200h Common Communication Parameter

Item	Value
Index	1200h
Name	SCOM_CommonCommParam_REC
Object type	RECORD
Data type	SCOM_CommonCommParamRecord_TYPE
Category	M
Relevant for SOD checksum	Yes

Table 101 – Object 1200h sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	3
Default value	3
Access	ro
SPDO mapping	No
[a] Number of Entries within this index.	

Table 102 – Object 1200h sub index 01h

Item	Value
Sub index	01h
Name	SDN_U16 [a]
Data type	UNSIGNED16
Value range	0 – 1 023
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Safety Domain Number of the device.	

Table 103 – Object 1200h sub index 02h

Item	Value
Sub index	02h
Name	SADR_U16 [a]
Data type	UNSIGNED16
Value range	0 -- 1 023
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Safety Address of the SCM within the SD.	

Table 104 – Object 1200h sub index 03h

Item	Value
Sub index	03h
Name	ConsecutiveTimeBase_I8 [a]
Data type	INTEGER8
Value range	0 -- 3
Default value	2
Category	M
Access	rw
SPDO mapping	No
[a] Time base identification for CT field, 0 = 1 µs, 1 = 10 µs, 2 = 100 µs, 3 = 1 000 µs.	

Time base of CT field shall be calculated by Equation (1).

$$TimeBase = 1\,\mu s \times 10^{ConsecutiveTimeBase\_I8} \quad (1)$$

where

TimeBase is the time base of CT field in µs;

ConsecutiveTimeBase_I8 is the time base identification

Table 105 – Object 1200h sub index 04h

Item	Value
Sub index	04h
Name	UDIDofSCM_ OSTR6 [a]
Data type	OCTET_STRING6
Value range	OCTET_STRING6
Default value	UDID of the own SN
Category	M
Access	rw
SPDO mapping	No
[a] UDID of the SCM.	

7.5.4.12 Object 1201h: SSDO Communication Parameter

This object defines the SSDO communication parameters of an SSDO client.

Object 1201h SSDO Communication Parameter is specified in Table 106 to Table 109.

Table 106 – Object 1201h SSDO Communication Parameter

Item	Value
Index	1201h
Name	SSDO_CommParam_REC
Object type	RECORD
Data type	SSDO_CommParamRecord_TYPE
Category	C
Relevant for SOD checksum	Yes

Table 107 – Object 1201h sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	2
Default value	2
Access	ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 108 – Object 1201h sub index 01h

Item	Value
Sub index	01h
Name	Timeout_U32 [a]
Data type	UNSIGNED32
Value range	1 -- 2^{32}
Default value	500 000
Category	M
Access	rw
SPDO mapping	No

[a] The sub index provides the timeout for an SSDO telegram. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.

Table 109 – Object 1201h sub index 02h

Item	Value
Sub index	02h
Name	Retries_U8 [a]
Data type	UNSIGNED8
Value range	0 -- 255
Default value	0
Category	M
Access	rw
SPDO mapping	No

[a] This sub index provides the number of retries for an SSDO telegram.

7.5.4.13 Object 1202h: SNMT Communication Parameter

This object defines the SNMT communication parameters of an SNMT master.

Object 1202h SNMT Communication Parameter is specified in Table 110 to Table 113.

Table 110 – Object 1202h SNMT Communication Parameter

Item	Value
Index	1202h
Name	SNMT_CommParam_REC
Object type	RECORD
Data type	SNMT_CommParamRecord_TYPE
Category	C
Relevant for SOD checksum	Yes

Table 111 – Object 1202h sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	2
Default value	2
Access	ro
SPDO mapping	No
[a] Number of Entries within this index.	

Table 112 – Object 1202h sub index 01h

Item	Value
Sub index	01h
Name	Timeout_U32 [a]
Data type	UNSIGNED32
Value range	1 -- 2^{32}
Default value	500 000
Category	M
Access	rw
SPDO mapping	No
[a] The sub index provides the timeout for an SNMT telegram. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 113 – Object 1202h sub index 02h

Item	Value
Sub index	02h
Name	Retries_U8 [a]
Data type	UNSIGNED8
Value range	0 -- 255
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] This sub index provides the number of retries for an SNMT telegram.	

7.5.4.14 Object 1400h -- 17FEh: RxSPDO Communication Parameter

These objects define from which SADR data shall be received.

One producer may have more SADRs (one for each TxSPDO). Therefore one consumer can have more than one RxSPDO for the same physical producer.

1400h -- 17FEh RxSPDO Communication Parameter is specified in Table 114 to Table 127.

Table 114 – Object 1400h -- 17FEh RxSPDO Communication Parameter

Item	Value
Index	1400h -- 17FEh
Name	SPDO_RxCommParam_XXXh_REC
Object type	RECORD
Data type	SPDO_RxCommParamRecord_TYPE
Category	O
Relevant for SOD checksum	Yes

Table 115 – Object 1400h -- 17FEh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	8
Default value	8
Access	ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 116 – Object 1400h -- 17FEh sub index 01h

Item	Value
Sub index	01h
Name	SADR_U16 [a]
Data type	UNSIGNED16
Value range	0 -- 1 023
Default value	0
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] Defines the SADR from which the RxSPDO data is to be received. Setting the SADR to zero deactivates the SPDO.	

Table 117 – Object 1400h -- 17FEh sub index 02h

Item	Value
Sub index	02h
Name	SCT_U32 [a]
Data type	UNSIGNED32
Value range	1 -- 2^{32}
Default value	1
Category	M
Access	rw
SPDO mapping	No
[a] Defines the SCT timer for the data from this RxSPDO (see 7.7.1.2). The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 118 – Object 1400h -- 17FEh sub index 03h

Item	Value
Sub index	03h
Name	NumberOfConsecutiveTReq_U8 [a]
Data type	UNSIGNED8
Value range	1 -- 63
Default value	1
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] The sub index provides the number of consecutive time requests.	

Table 119 – Object 1400h -- 17FEh sub index 04h

Item	Value
Sub index	04h
Name	TimeDelayTReq_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] This sub index provides the time delay between two sets of consecutive time requests. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 120 – Object 1400h -- 17FEh sub index 05h

Item	Value
Sub index	05h
Name	TimeDelaySync_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	1
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] This sub index provides the time delay between successful time synchronization and the next time request. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 121 – Object 1400h -- 17FEh sub index 06h

Item	Value
Sub index	06h
Name	MinTSyncPropagationDelay_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 65 535
Default value	1
Category	M
Access	rw
SPDO mapping	No
[a] The sub index provides the minimum allowed propagation delay between time request and time response. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 122 – Object 1400h -- 17FEh sub index 07h

Item	Value
Sub index	07h
Name	MaxTSyncPropagationDelay_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 65 535
Default value	1
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] The sub index provides the maximum allowed propagation delay between time request and time response. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 123 – Object 1400h -- 17FEh sub index 08h

Item	Value
Sub index	08h
Name	MinSPDOPropagationDelay_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 65 535
Default value	1
Category	M
Access	rw
SPDO mapping	No
[a] The sub index provides the minimum allowed propagation delay between sending and receiving an SPDO. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 124 – Object 1400h -- 17FEh sub index 09h

Item	Value
Sub index	09h
Name	MaxSPDOPropagationDelay_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 65 535
Default value	1
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] The sub index provides the minimum allowed propagation delay between sending and receiving an SPDO. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 125 – Object 1400h -- 17FEh sub index 0Ah

Item	Value
Sub index	0Ah
Name	BestCaseTResDelay_U16 [a]
Data type	UNSIGNED16
Value range	0 -- 65 535
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] The sub index provides the best case delay between sending the time request and setting up the time response. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 126 – Object 1400h -- 17FEh sub index 0Bh

Item	Value
Sub index	0Bh
Name	TimeRequestCycle_U32 [a]
Data type	UNSIGNED32
Value range	1 -- 2^{32}
Default value	1
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] The sub index provides the time request cycle time. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 127 – Object 1400h -- 17FEh sub index 0Ch

Item	Value
Sub index	0Ch
Name	TxSPDONo_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 1 023
Default value	1
Category	M
Access	rw
SPDO mapping	No
[a] Contains the TxSPDO number with which the time request is to be sent.	

7.5.4.15 Object 1800h -- 1BFEh: RxSPDO Mapping Parameter

This object defines the RxSPDO mapping parameters. The mapping parameters may be predefined by the vendor and read only. For a detailed description of SPDO mapping, refer to 7.6.

Object 1800h -- 1BFEh RxSPDO Mapping Parameter is specified in Table 128 to Table 131.

Table 128 – Object 1800h -- 1BFEh RxSPDO communication parameter

Item	Value
Index	1800h -- 1BFEh
Name	SPDO_RxMappParam_XXXh_AU32
Object type	ARRAY
Data type	UNSIGNED32
Category	O
Relevant for SOD checksum	Yes

Table 129 – Object 1800h -- 1BFEh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	0 -- 253
Default value	0
Access	rw / ro
SPDO mapping	No
[a] Number of Entries within this index.	

Table 130 – Object 1800h -- 1BFEh sub index 01h

Item	Value
Sub index	01h
Name	SPDOMapping_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw / ro
SPDO mapping	No
[a] The sub index provides the mapping parameter for this TxSPDO.	

Table 131 – Object 1800h -- 1BFEh sub index 02h -- FDh

Item	Value
Sub index	02h -- FDh
Name	SPDOMapping_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	O
Access	rw / ro
SPDO mapping	No
[a] The sub index provides the mapping parameter for this TxSPDO.	

7.5.4.16 Object 1C00h -- 1FFEh: TxSPDO Communication Parameter

Each SN shall have at least one TxSPDO, either for sending data (producer) or for the time request (consumer).

Object 1C00h -- 1FFEh TxSPDO Communication Parameter is specified in Table 132 to Table 136.

Table 132 – 1Object C00h -- 1FFEh TxSPDO communication parameter

Item	Value
Index	1C00h -- 1FFEh
Name	SPDO_TxCommParam_XXXh_REC
Object type	RECORD
Data type	SPDO_TxCommParamRecord_TYPE
Category	M/O
Relevant for SOD checksum	Yes

Table 133 – Object 1C00h -- 1FFEh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	3
Default value	3
Access	ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 134 – Object 1C00h -- 1FFEh sub index 01h

Item	Value
Sub index	01h
Name	SADR_XXXh_U16 [a]
Data type	UNSIGNED16
Value range	0 -- 1 023
Default value	0
Category	M
Access	rw
SPDO mapping	No
Relevant for SOD checksum	
[a] Defines the SADR that is used when broadcasting the data of the corresponding TxSPDO within the network. Setting the SADR to zero deactivates the SPDO.	

Table 135 – Object 1C00h -- 1FFEh sub index 02h

Item	Value
Sub index	02h
Name	RefreshPrescale_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 32 767
Default value	1
Category	M
Access	rw
SPDO mapping	No
[a] The sub index provides the refresh time of the producer. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 136 – Object 1C00h -- 1FFEh sub index 03h

Item	Value
Sub index	03h
Name	NumberOfTRes_U8 [a]
Data type	UNSIGNED8
Value range	UNSIGNED8
Default value	0
Category	M/C
Access	rw
SPDO mapping	No
[a] Defines the number of consecutive TRes for a received TReq. If the parameter is zero, the TxSPDO can only be used for TReq telegrams (for devices which are consumers only).	

7.5.4.17 Object C000h -- C3FEh: TxSPDO Mapping Parameter

This object defines the TxSPDO mapping parameters. The mapping parameters may be predefined by the vendor and read only. For a detailed description of the SPDO mapping, refer to 7.6.

Object C000h -- C3FEh TxSPDO Mapping Parameter is specified in Table 137 to Table 140.

Table 137 – Object C000h -- C3FEh TxSPDO mapping parameter

Item	Value
Index	C000h -- C3FEh
Name	SPDO_TxMappParam_XXXh_AU32
Object type	ARRAY
Data type	UNSIGNED32
Category	M/O
Relevant for SOD checksum	Yes

Table 138 – Object C000h -- C3FEh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	0 -- 253
Default value	0
Access	rw / ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 139 – Object C000h -- C3FEh sub index 01h

Item	Value
Sub index	01h
Name	SPDOMapping_U32 [a]
Object type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw/ro
SPDO mapping	No
Relevant for SOD checksum	
[a] The sub index provides the mapping parameter for this TxSPDO.	

Table 140 – Object C000h -- C3FEh sub index 02h -- FDh

Item	Value
Sub index	02h -- FDh
Name	SPDOMapping_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	O
Access	rw/ro
SPDO mapping	No
[a] The sub index provides the mapping parameter for this TxSPDO.	

7.5.4.18 Object C400h -- C7FEh: SADR-DVI List

This object shall be implemented if the SN has SCM functionality. It shall contain a list of all SNs within the SD. The list shall contain the parameters for each SN which are needed to verify the SN during start up. The maximum number of supported SNs shall be vendor specific. Therefore the first two indices (C400h, C401h) shall be mandatory (one SN for the SCM and at least one additional SN), all other shall be optional.

Object C400h -- C7FEh SADR-DVI List is specified in Table 141 to Table 153.

Table 141 – Object C400h -- C7FEh SADR-DVI list

Item	Value
Index	C400h -- C7FEh
Name	SSCM_SADRDVIList_XXXh_REC
Object type	RECORD
Data type	SSCM_SADRDVIListRecord_TYPE
Category	C
Relevant for SOD checksum	No

Table 142 – Object C000h -- C3FEh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	09h -- 0Bh
Default value	9
Access	ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 143 – Object C000h -- C3FEh sub index 01h

Item	Value
Sub index	01h
Name	SADR_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 1 023
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the expected SADR of the corresponding SN.	

Table 144 – Object C000h -- C3FEh sub index 02h

Item	Value
Sub index	02h
Name	VendorId_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the Vendor ID of the corresponding SN.	

Table 145 – Object C000h -- C3FEh sub index 03h

Item	Value
Sub index	03h
Name	ProductCode_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the product code of the corresponding SN.	

Table 146 – Object C000h -- C3FEh sub index 04h

Item	Value
Sub index	04h
Name	RevisionNumber_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the revision number of the corresponding SN.	

Table 147 – Object C000h -- C3FEh sub index 05h

Item	Value
Sub index	05h
Name	ModuleStatus_U8 [a]
Data type	UNSIGNED8
Value range	0 -- 6
Default value	0
Category	M
Access	ro
SPDO mapping	No
[a] Status of the corresponding SN, 0 = missing, 1 = invalid, 2 = wrong SADR, 3 = UDID mismatch, 4 = wrong parameters, 5 = valid, 6 = OK.	

Table 148 – Object C000h -- C3FEh sub index 06h

Item	Value
Sub index	06h
Name	ParameterSetChecksum_U16 [a]
Data type	UNSIGNED16
Value range	UNSIGNED16
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Checksum of parameter set of the corresponding SN.	

Table 149 – Object C000h -- C3FEh sub index 07h

Item	Value
Sub index	07h
Name	ParameterSetTimestamp_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Timestamp of parameter set of the corresponding SN.	

Table 150 – Object C000h -- C3FEh sub index 08h

Item	Value
Sub index	08h
Name	MaximumSsdoPayloadLen_U16 [a]
Data type	UNSIGNED16
Value range	8 -- 254
Default value	8
Category	M
Access	rw
SPDO mapping	No
[a] Maximum length of the payload of an SSDO telegram. This parameter is important because not all SNs might support long SSDO telegrams.	

Table 151 – Object C000h -- C3FEh sub index 09h

Item	Value
Sub index	09h
Name	SnmtCrcPollInterval_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	100
Category	M
Access	rw
SPDO mapping	No
[a] This sub index provides the poll interval for the SNMT_SN_set_to_OP command. The time base of this parameter is defined by CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Table 152 – Object C000h -- C3FEh sub index 0Ah

Item	Value
Sub index	0Ah
Name	ParameterSetLength_U32 [a]
Data type	UNSIGNED32
Value range	UNSIGNED32
Default value	0
Category	O
Access	rw
SPDO mapping	No
[a] Contains the length of the parameter set for the corresponding SN.	

Table 153 – Object C000h -- C3FEh sub index 0Bh

Item	Value
Sub index	0Bh
Name	ParameterSet_DOM [a]
Data type	DOMAIN
Value range	DOMAIN
Default value	0
Category	C
Access	rw
SPDO mapping	No
[a] Contains the parameter set for the corresponding SN. This sub index needs to be implemented if sub index 0Ah was implemented.	

7.5.4.19 Object C801h -- CBFFh: Additional SADR list

This Object shall describe the addresses (SADR) of an SN. Sub index 1 shall describe the "Main-SADR" to which the corresponding address belongs. Sub index 2 shall describe an additional TxSPDO address. The "Main-SADR" is the SADR of the 1st TxSPDO and is also used by the SSDO services.

Object C801h -- CBFFh Additional SADR List is specified in Table 154 to Table 157.

Table 154 – Object C801h -- CBFFh Additional SADR list

Item	Value
Index	C801h -- CBFFh
Name	SSCM_SADRMembership_XXXh_AU16
Object type	ARRAY
Data type	UNSIGNED16
Category	O
Relevant for SOD checksum	No

Table 155 – Object C801h -- CBFFh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	2
Default value	2
Access	ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 156 – Object C801h -- CBFFh sub index 01h

Item	Value
Sub index	01h
Name	SADR_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 1 023
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the “Main-SADR” which uses the SADR for a TxSPDO.	

Table 157 – Object C801h -- CBFFh sub index 02h

Item	Value
Sub index	02h
Name	TxSPDONo_U16 [a]
Data type	UNSIGNED16
Value range	1 -- 1 023
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the TxSPDO number, the SADR is used for.	

EXAMPLE

Given the configuration shown in Table 158 below, this is to be interpreted as follows:

1st line: SADR 1 (Index – C800h) belongs to the SN with “Main-SADR” 1 and is used in TxSPDO 1

2nd line: SADR 2 (Index – C800h) belongs to the SN with “Main-SADR” 2 and is used in TxSPDO 1

3rd line: SADR 3 (Index – C800h) belongs to the SN with “Main-SADR” 3 and is used in TxSPDO 1

4th line: SADR 4 (Index – C800h) belongs to the SN with “Main-SADR” 1 and is used in TxSPDO 2

5th line: SADR 5 (Index – C800h) belongs to the SN with “Main-SADR” 1 and is used in TxSPDO 3

6th line: SADR 6 (Index – C800h) belongs to the SN with “Main-SADR” 0, 0 is not a valid SADR -> therefore this SADR does not belong to any SN

Summary: the SN with “Main-SADR” 1 has 3 TxSPDOs using SADR 1, 4 and 5. SN 2 and 3 only have one SADR (therefore only 1 TxSPDO).

Table 158 – Object Additional SADR List Example

Index (hex)	Sub index 0	Sub index 1	Sub index 2
C801	2	1	1
C802	2	2	1
C803	2	3	1
C804	2	1	2
C805	2	1	3
C806	2	0	0

7.5.4.20 Object CC01h -- CFFFh: SADR-UDID List

These objects shall contain information specifying which UDIDs can belong to which SADR. One logical SN (described with the “Main-SADR”) can be switched between more than one physical device (modular machine part). All valid devices (described by their UDID) are listed within these objects.

Object CC01h -- CFFFh SADR-UDID List is specified in Table 159, Table 160 and Table 161.

Table 159 – Object CC01h -- CFFFh SADR-UDID list

Item	Value
Index	CC01h -- CFFFh
Name	SSCM_SADRUDIDList_XXXh_AOSTR6
Object type	ARRAY
Data type	OCTET_STRING6
Category	O
Relevant for SOD checksum	No

Table 160 – Object C801h -- CBFFh sub index 00h

Item	Value
Sub index	00h
Name	NumberOfEntries [a]
Value range	1 -- 253
Default value	1
Access	ro
SPDO mapping	No
[a] Number of entries within this index.	

Table 161 – Object C801h -- CBFFh sub index 01h -- FDh

Item	Value
Sub index	01h -- FDh
Name	UDID_OSTR6 [a]
Data type	OCTET_STRING6
Value range	OCTET_STRING6
Default value	0
Category	M
Access	rw
SPDO mapping	No
[a] Contains the UDIDs which are allowed for the corresponding SADR.	

EXAMPLE Given the configuration shown in Table 162, this is to be interpreted as follows. For SADR 0001 (CC01h - CC00h), two UDIDs are allowed. Either UDID 50-aa-c0-43-56-3f or 50-aa-c0-43-56-42 is accepted as UDID of the module with SADR 0001.

Table 162 – SADR-UDID List Example

Index (hex)	Sub index 0	Sub index 1	Sub index 2
CC01	2	50-aa-c0-43-56-3f	50-aa-c0-43-56-42

7.6 Safety related PDO mapping

7.6.1 General

The real-time data transfer shall be performed by means of Safety Process Data Objects (SPDO).

Data length and mapping of application objects into SPDOs shall be determined by the corresponding SPDO mapping structures within the Safety Object Dictionary (SOD). The mapping of application objects into an SPDO may be transmitted to a device during the device configuration process by applying the SSDO services to the corresponding entries of the Object Dictionary.

The length of SPDOs of a device is application-specific and shall be determined by the corresponding mapping object.

There are two uses for SPDOs. The first is data transmission and the second data reception. Transmit SPDOs (TxSPDOs) and Receive SPDOs (RxSPDOs) are distinguished. Devices supporting TxSPDOs are SPDO producers and devices which are able to receive SPDOs are called SPDO consumers.

SPDOs is described by the SPDO communication parameter and the SPDO mapping parameter. For each SPDO, the pair of communication and mapping parameters shall be mandatory.

SPDO communication parameter shall describe the communication capabilities of the SPDO.

SPDO mapping parameter shall describe mapping for each object contained in SPDO payload data to object dictionary entries and vice versa.

SPDO mapping shall be variable or static. Static mapping shall be pre-defined by the vendor and shall not be modified in any way. Variable mapping shall be configured during commissioning.

7.6.2 Transmit SPDOs

The TxSPDO communication parameters shall be specified by SOD indices 1C00h -- 1FFEh (SPDO_TxCommParam_XXXh_REC). An SN may support up to 1 023 TxSPDOs. The implementation of the first TxSPDO shall be mandatory. The amount of additional TxSPDOs is vendor specific. The TxSPDO mapping parameters is described by indices C000h -- C3FEh (SPDO_TxMappParam_XXXh_AU32). The data to be transmitted shall be assembled according to the corresponding mapping data.

Sending SPDO data shall be implicitly limited to the Operational state. Refer to 7.5 for the corresponding entries within the SOD.

7.6.3 Receive SPDOs

The RxSPDO communication parameters is described by indices 1400h -- 17FEh (RxSPDOCommParam_XXXh_REC). An SN may support up to 1 023 RxSPDOs. The implementation of RxSPDOs shall be optional (e.g. an input module, which is a simple producer, does not need to have an RxSPDO). The RxSPDO mapping parameters is described by indices 1800h -- 1BFEh (SPDO_RxMappParam_XXXh_AU32). The received data shall be transferred to the communication objects assigned to them by the RxSPDO mapping parameters. The application accesses the received SPDO data by reading these communication objects.

Receiving SPDO data shall be implicitly limited to the Operational state. Refer to 7.5 for the corresponding entries within the SOD.

7.6.4 SPDO mapping parameter

The indices 1800h -- 1BFEh (SPDO_TxMappParam_XXXh_AU32) and C000h -- C3FEh (SPDO_RxMappParam_XXXh_AU32) shall contain the mapping parameters for transmitting and receiving SPDOs. For both indices, sub index 0 shall contain the number of valid mapping entries which is also the number of communication variables (SOD entries) to be transmitted or received. The sub indices 01h to NumberOfEntries shall contain information about the mapped application variables. These entries shall describe the SPDO contents by their index, sub index and length. All three values shall be given in hexadecimal notation. The encoding of a mapping entry (sub index 01_h -- FD_h) is specified by Table 163. The length entry contains the length of the object in bits (01h -- FFh). This parameter can be used to check the overall mapping length.

Table 163 – Structure of SPDO mapping entry

Bit 31 -- bit 16	Bit 15 -- bit 8	Bit 7 -- bit 0
Index (16 bit)	sub index (8 bit)	data length (8 bit)

The commissioner shall be responsible for the correctness of the mapping data. If objects are mapped which shall not be mapped according to the object attribute, the SN shall not switch to state Operational. If data within an RxSPDO is not used by the consumer, it shall be mapped to the data types (index 0001h -- 0007h), these serve as “dummy entries”. The corresponding data within the RxSPDO shall not be evaluated by the device. This feature is useful if a TxSPDO is used for several consumers, each only utilizing parts of the transmitted data. It shall not be possible to map those “dummy entries” into a TxSPDO.

7.6.5 SPDO mapping example

Example devices (see Figure 42) are:

— the SN with “Main-SADR” 12 is an input module with 32 digital inputs,
— the SN with “Main-SADR” 13 is an output module with 8 digital outputs, and
— the SN with “Main-SADR” 14 is an output module with 16 digital outputs.

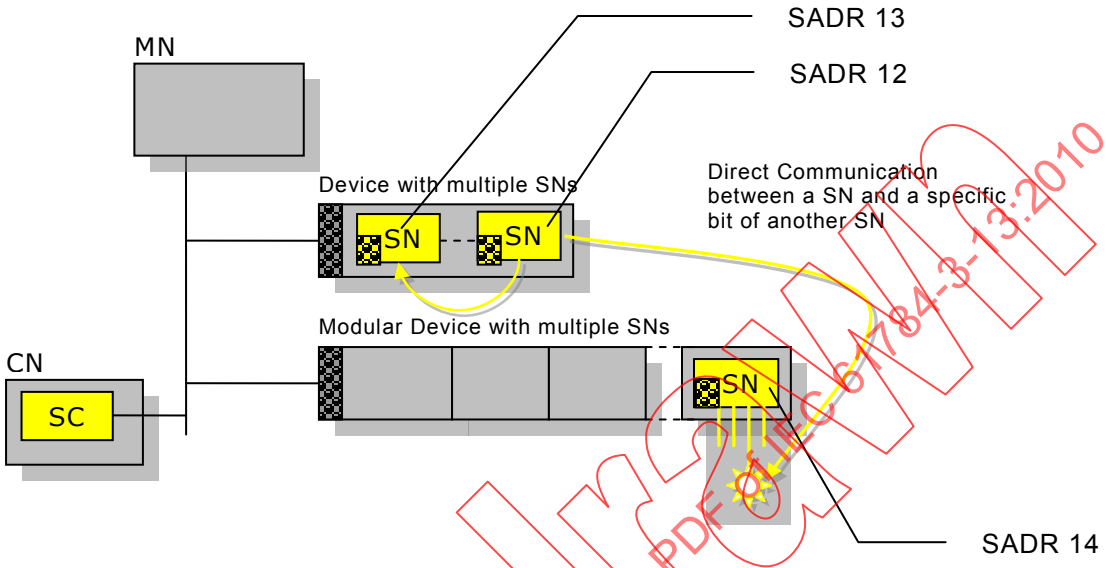


Figure 42 – SPDO mapping example

It is assumed that the inputs are available at SOD-index 6000h sub indices 01h – 04h and that the data type is UNSIGNED8. It is also assumed that the outputs need to be written to SOD index 6200h sub indices 01h respectively 01h and 02h. The data type again is UNSIGNED8. It is then assumed that SN 13 needs the information from the 3rd input octet and that SN 14 needs the information from the 1st and 4th input octet.

The communication parameters from SN 12 (producer) are shown in Table 164.

Table 164 – Mapping example table 1

Index	Sub index	Data	Description
1C00h	1	12	The TxSPDO is broadcast under the SADR 12

The mapping of SN 12 (producer) may be specified as in Table 165.

Table 165 – Mapping example table 2

Index	Sub index	Data	Description
C000h	0	4	four valid mapping entries
	1	60000108h	Mapping entry 1: 8bits of the object from index 6000h sub index 1 are mapped into the SPDO
	2	60000208h	Mapping entry 2: 8bits of the object from index 6000h sub index 2 are mapped into the SPDO
	3	60000308h	Mapping entry 3: 8bits of the object from index 6000h sub index 3 are mapped into the SPDO
	4	60000408h	Mapping entry 4: 8bits of the object from index 6000h sub index 4 are mapped into the SPDO

This TxSPDO mapping could be a predefined mapping. Input 2 is mapped into the TxSPDO although it is not needed by any of the SNs.

According to the mapping parameters, the payload data of the TxSPDO looks as in Table 166.

Table 166 – Mapping example table 3

Octet	Data
1	Input Octet 1
2	Input Octet 2
3	Input Octet 3
4	Input Octet 4

The communication parameter for SN 13 is specified as in Table 167.

Table 167 – Mapping example table 4

Index	Sub index	Data	Description
1400h	1	12	SN 13 will map the data of a received TxSPDO from SADR 12 according to the mapping entries

The mapping of the SN 13 looks as in Table 168.

Table 168 – Mapping example table 5

Index	Sub index	Data	Description
1800h	0	2	two valid mapping entries
	1	00060010h	Mapping entry 1: 16 bits of the RxSPDO are mapped into the object at index 0006h sub index 0
	2	62000108h	Mapping entry 2: 8bits of the RxSPDO are mapped into the object at index 6200h sub index 1
	3	00050008h	Mapping entry 2: 8bits of the RxSPDO are mapped into the object at index 0005h sub index 0

As described above, SN 13 is only interested in the 3rd input octet of SN 12. Therefore, the first two input octets (first two octets of the RxSPDO) are ignored by using “dummy mapping”. The 4th octet is also mapped to a “dummy entry”. This is required because only RxSPDOs which have the same length as the mapped data are accepted.

The communication parameter of SN 14 is shown in Table 169.

Table 169 – Mapping example table 6

Index	Sub index	Data	Description
1400h	1	12	SN 14 will map the data of a received TxSPDO from SADR 12 according to the mapping entries

The mapping of the SN 14 is shown in Table 170.

Table 170 – Mapping example table 7

Index	Sub index	Data	Description
1800h	0	3	three valid mapping entries
	1	62000108h	Mapping entry 1: 8bits of the RxSPDO are mapped into the object at index 6200h sub index 1
	2	00060010h	Mapping entry 2: 16 bits of the RxSPDO are mapped into the object at index 0006h sub index 0
	3	62000208h	Mapping entry 3: 8bits of the RxSPDO are mapped into the object at index 6200h sub index 2

As described above, SN 14 is only interested in input octets 1 and 4 of SN 12. Input octets 2 and 3 (respectively octet 2 and 3 of the RxSPDO) are ignored by using "dummy mapping".

7.6.6 SPDO error handling

7.6.6.1 Non-mapable application object

The SN shall ensure that only mapable application objects are accepted. If a non mapable object is recognized, the SN shall not store this mapping data into the SOD.

NOTE As a result, the checksum of the SOD will be different from the expected checksum stored on the SCM and therefore, the SN will not be switched to the Operational state (see 7.7.12.2).

7.6.6.2 Unexpected length of RxSPDO

If an SPDO is received with a length that is not equal to the length of the mapped data, the SPDO shall be ignored. Additional diagnosis data may be provided via the SOD.

7.7 State and sequence diagrams

7.7.1 Safety Process Data Object (SPDO)

7.7.1.1 Safety Process Data Object producer (TxSPDO)

The producer shall start sending its data cyclically or on state changes when switching to Operational.

NOTE The refresh time for sending the data can be adjusted within the SOD.

The Safety Process Data Object Producer state diagram is specified by Figure 43.

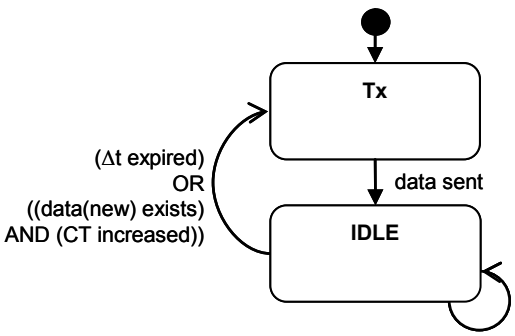
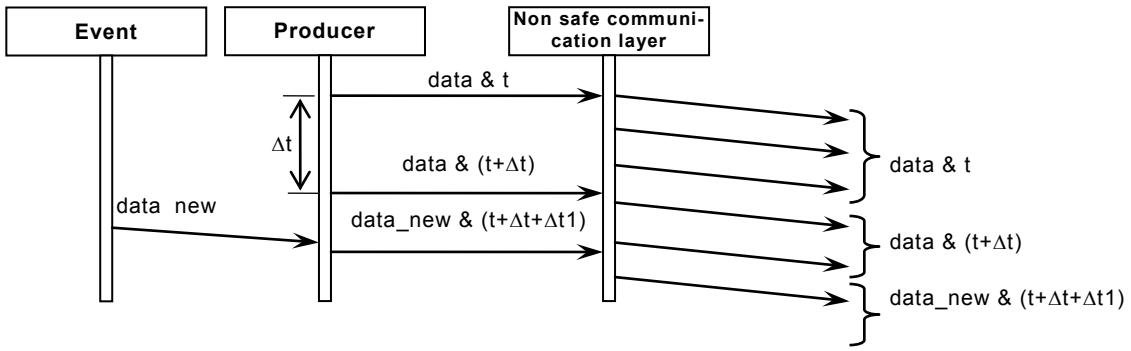


Figure 43 – State diagram TxSPDO

The Safety Process Data Object Producer sequence diagram is specified by Figure 44.



Key
See Table 171.

Figure 44 – SPDO communication producer

Table 171 specifies items used in Figure 44.

Table 171 – SPDO communication producer item description

Item	Description	Min value	Max value	SOD
data_new	New data from safety related application or new TRes received	-	-	-
t	Internal time [CT]	0	65 535	-
Δt	Refresh prescale [CT]	1	32 767	Object Index 1C00h -- 1FFEh sub index 02h (see 7.5.4.16)
Δt1	Time difference of new data [CT]	>0	Δt	-

Table 172 specifies the SPDO communication producer states.

Table 172 – SPDO communication producer state description

State	Description
Tx	Sending data
IDLE	Wait until refresh time elapsed or new data available

7.7.1.2 Safety Process Data Object consumer (RxSPDO)

7.7.1.2.1 General behaviour

For each received RxSPDO action as specified in the Safety Process Data Object Producer, state diagram (Figure 45) shall be taken by the consumer. The data shall be processed. Processing data shall include time synchronization and time validation, see 7.7.1.2.2. If a consumer receives valid data from a producer, the SCT timer shall be restarted. If the SCT timer elapses, all data related to the RxSPDO shall be set to a safe state.

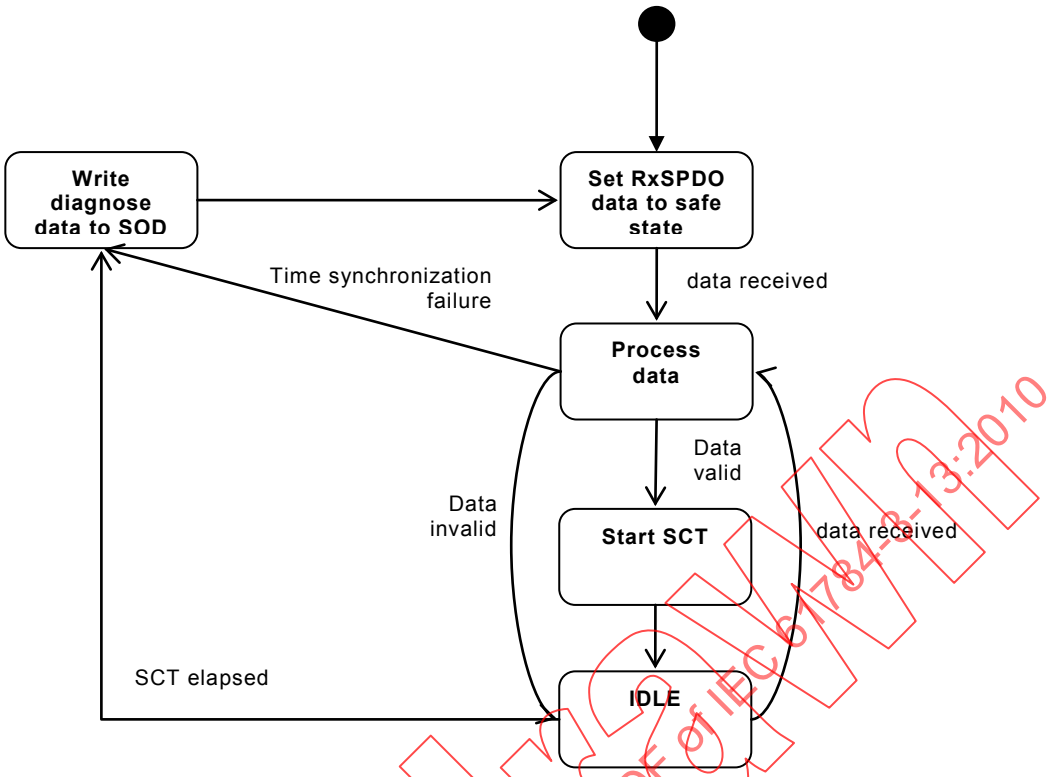
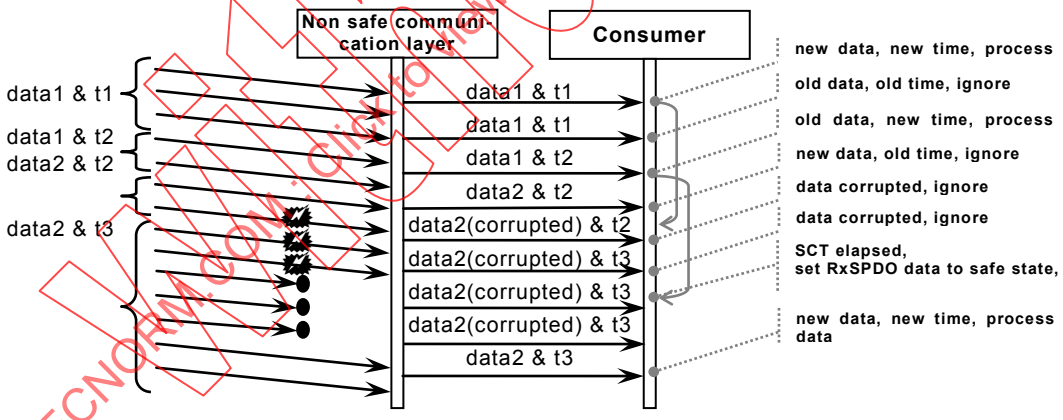


Figure 45 – State diagram RxSPDO

The Safety Process Data Object Consumer sequence diagram is specified by Figure 46.



Key
See Table 173.

Figure 46 – SPDO communication consumer

Table 173 specifies the item used in Figure 46.

Table 173 – SPDO communication consumer item description

Item	Description	Min value	Max value	SOD
SCT	Safety control time [CT]	1	UNSIGNED32	Object 1400h -- 17FEh sub index 02h (see 7.5.4.14)

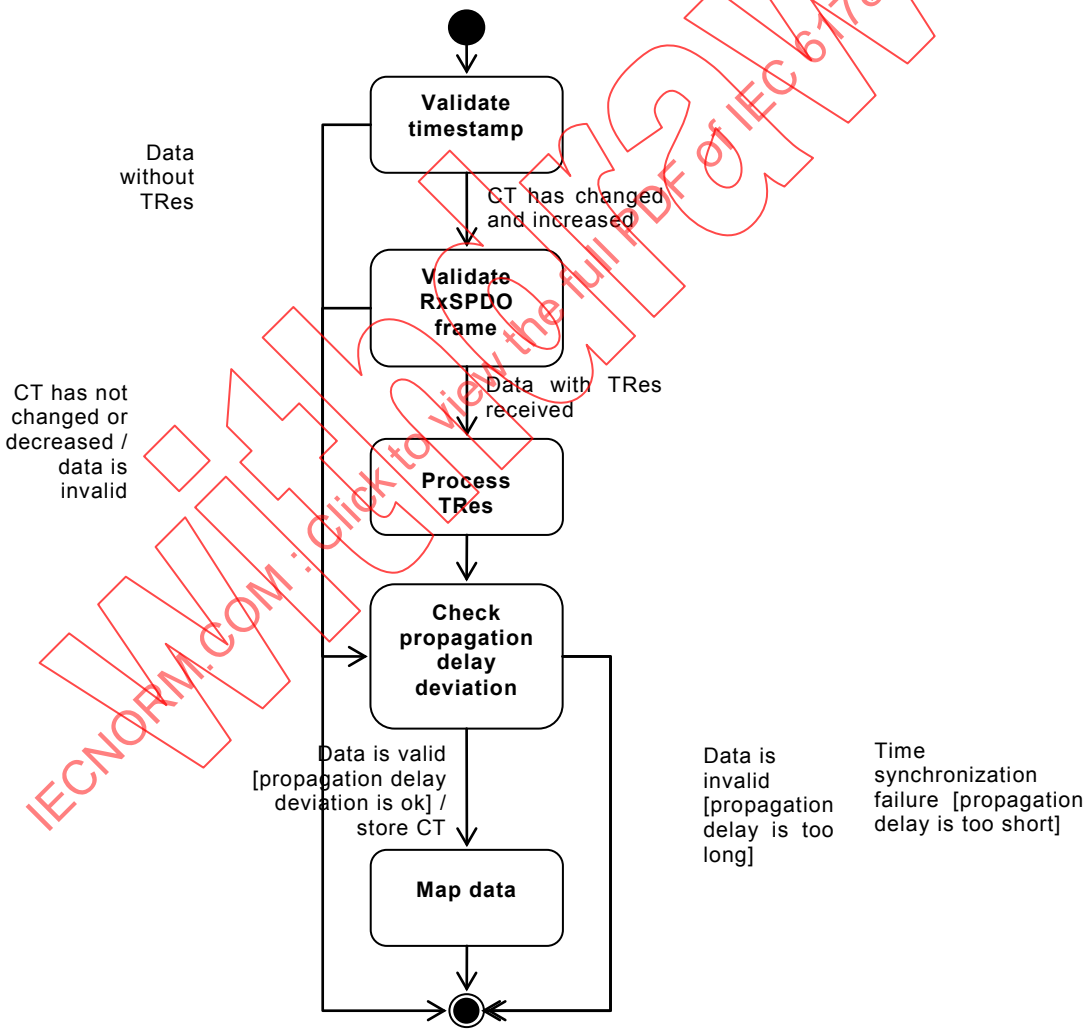
Table 174 specifies the SPDO communication consumer states.

Table 174 – SPDO communication consumer state description

State	Description
IDLE	Waiting for data
Process data	Performing timing validation and time synchronization. Process data according to SPDO mapping
Start SCT	Reset the SCT
Write diagnose data to SOD	The diagnose data has to be available for the user. This is made through the SOD
Set RxSPDO data to safe state	All data related to the producer will be set to zero

7.7.1.2.2 Process data

The process data state diagram is specified by Figure 47.



Key
See Table 175 and Table 176.

Figure 47 – State diagram process data

Table 175 specifies items used in Figure 47.

Table 175 – SPDO communication consumer telegram validation item description

Item	Description	Min value	Max value	SOD
CT	Consecutive Time field	0	65 535	-
TRes	Time synchronization response, consists of TR (Time Request Distinctive Number Field) and TADR (Time Request Address) within the safety PDU	-	-	-

Table 176 specifies the process data states.

Table 176 – SPDO communication consumer telegram validation state description

State	Description
Validate timestamp	If the timestamp (CT) is equal to or less than the timestamp of the previously valid telegram, the telegram shall be ignored
Validate RxSPDO PDU frame	Checks if the PDU contains time synchronization data for this node
Process TRes	See 7.7.2.7
Check propagation delay deviation	Checks timestamp validity
Map data	Writes the data into the SOD according to the RxSPDO mapping

7.7.2 Time synchronization and validation

7.7.2.1 General

To verify the performance of the non safety communication layer, FSCP 13/1 shall use a combined time synchronization and validation sequence. This shall be done cyclically.

NOTE Cyclically because of tolerances of the hardware clocks within the safety nodes.

Figure 48 specifies the basic sequences for the time synchronization and validation.

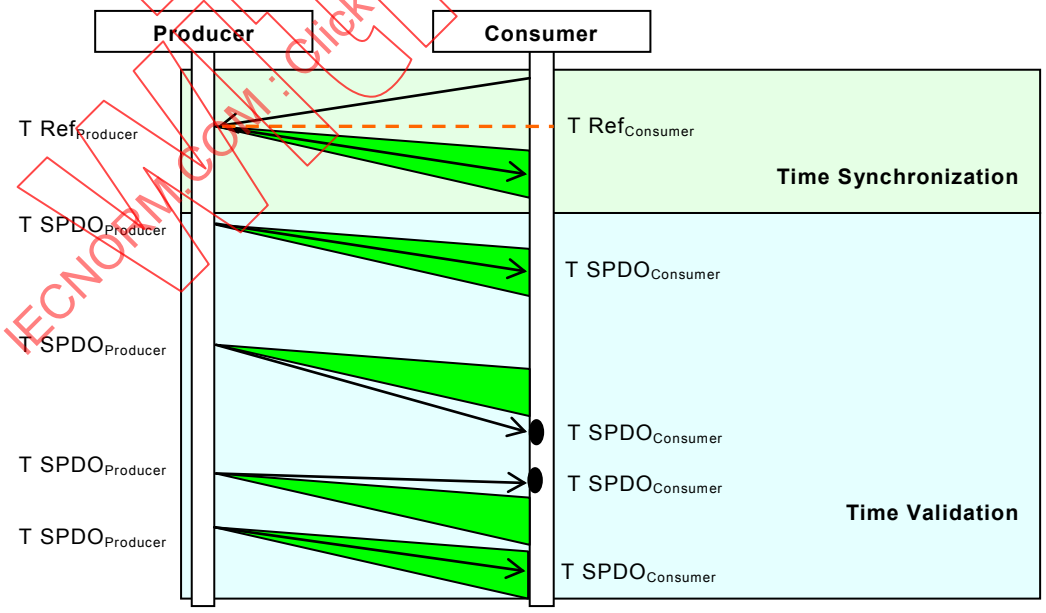
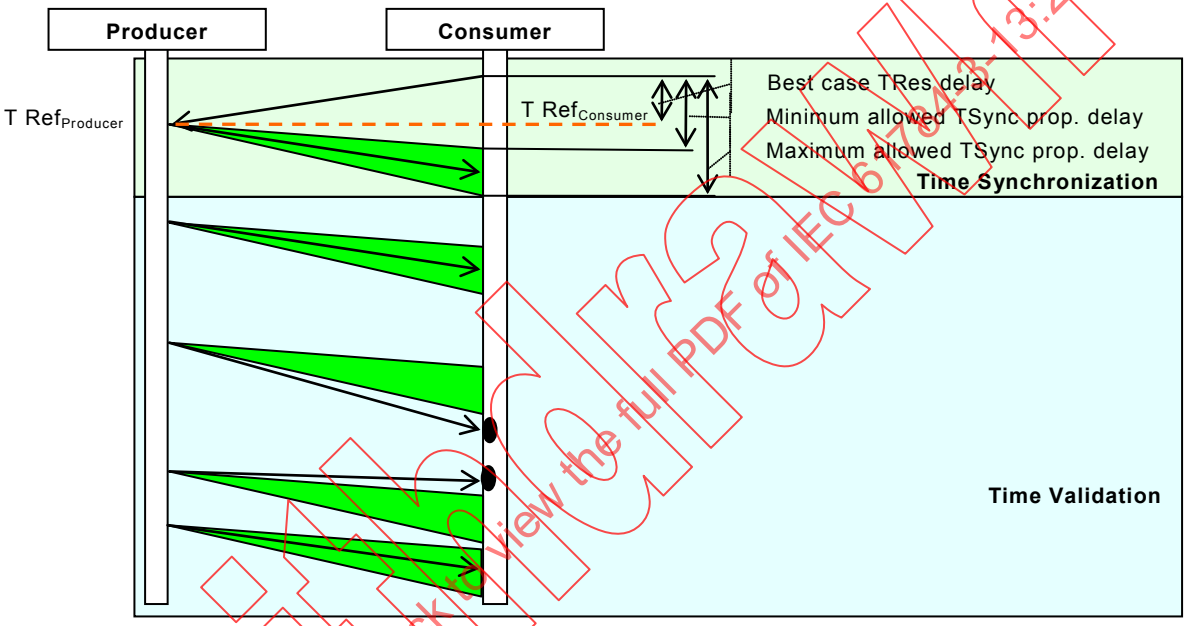


Figure 48 – Time synchronization and validation

7.7.2.2 Time synchronization

The consumer shall start the sequence by sending a set of Time Requests (TReq) to the producer. This shall be done by using a TxSPDO. When receiving the first TReq, the producer shall create a "new data event" (see 7.7.1.1) to reply to the request as soon as possible with the corresponding set of Time Responses (TRes). Then the first received TRes within the consumer shall be checked against the minimum and maximum allowed TSync propagation delay. If the delay is shorter than the minimum allowed delay, the consumer shall enter the safe state, because in this case the best case TRes delay parameter is incorrect. If the delay is longer than the maximum allowed delay, the time response shall be ignored. If delay is within the given limits, time synchronization shall be considered successful and the consumer shall memorize the internal T $Ref_{Producer}$ and T $Ref_{Consumer}$ values.

Figure 49 specifies the synchronization sequence.



Key
See Table 177.

Figure 49 – Time synchronization detail

Table 177 specifies items used in Figure 49.

Table 177 – Time synchronization item description

Item	Description	Min value	Max value	SOD
T $Ref_{Producer}$	Value of CT field within the TRes service supplied by the producer	0	65 535	-
best case TRes delay	User defined value to define the best case minimum time needed to transfer the time request from the consumer to the producer and to set up the time response within the producer. This value is used to calculate the SPDO propagation delay. If the user assumes a larger value than the real value, the time validation would not recognize impermissible delays in the SPDO propagation	0	65 535	Object index 1400h -- 17FEh sub index 0Ah (see 7.5.4.14)

Item	Description	Min value	Max value	SOD
T $Ref_{Consumer}$	Time Request start time [CT] + best case TRes delay [CT]. This value is calculated within the consumer and gives a time reference within the consumers CT to T $Ref_{Producer}$	> best case TRes delay	65 535	-
Minimum allowed TSync propagation delay	Minimum allowed propagation delay for time synchronization [CT]	1	65 535	Object index 1400h -- 17FEh sub index 06h (see 7.5.4.14)
Maximum allowed TSync propagation delay	Maximum allowed propagation delay for time synchronization [CT]	1	65 535	Object index 1400h -- 17FEh sub index 07h (see 7.5.4.14)

7.7.2.3 Time validation

After successful time synchronization, the consumer shall receive SPDOs and calculate the propagation delay as specified by Equations (2), (3) and (4).

$$SPDO\ propagation\ delay = T\_SPDO\_to\_Ref_{Consumer} - T\_SPDO\_to\_Ref_{Producer} \quad (2)$$

$$T\_SPDO\_to\_Ref_{Consumer} = T\ SPDO_{Consumer} - T\ Ref_{Consumer} \quad (3)$$

$$T\_SPDO\_to\_Ref_{Producer} = T\ SPDO_{Producer} - T\ Ref_{Producer}) \quad (4)$$

where

$T\ SPDO_{Producer}$ is the value of CT field within the SPDO supplied by the producer;

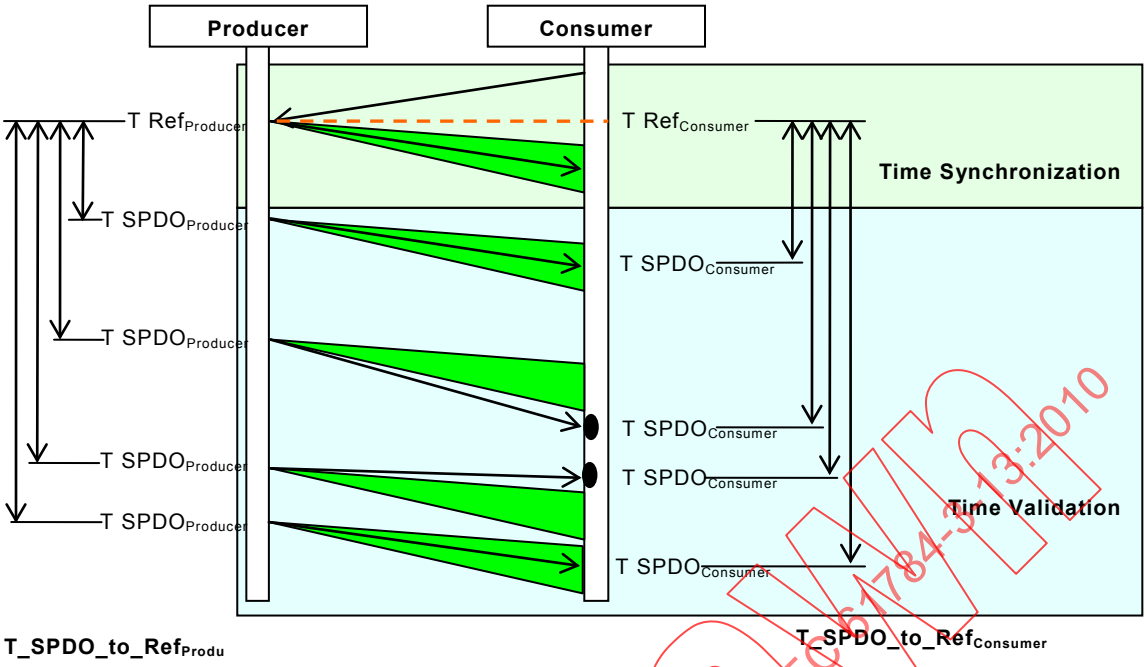
$T\ SPDO_{Consumer}$ is the CT value within the consumer when receiving the SPDO;

$T\ Ref_{Producer}$ is the value of CT field within the TRes service supplied by the producer;

$T\ Ref_{Consumer}$ is the time reference within the consumers CT to T RefProducer see Table 177

The result of Equation (2) shall be checked against the minimum and maximum allowed SPDO propagation delay. If the delay is shorter then the minimum allowed delay, the consumer shall enter the safe state, because in this case the best case TRes delay parameter was set wrong. If the delay is longer than the maximum allowed delay, the SPDO shall be ignored. If delay is within the given limits, the SPDO shall be processed.

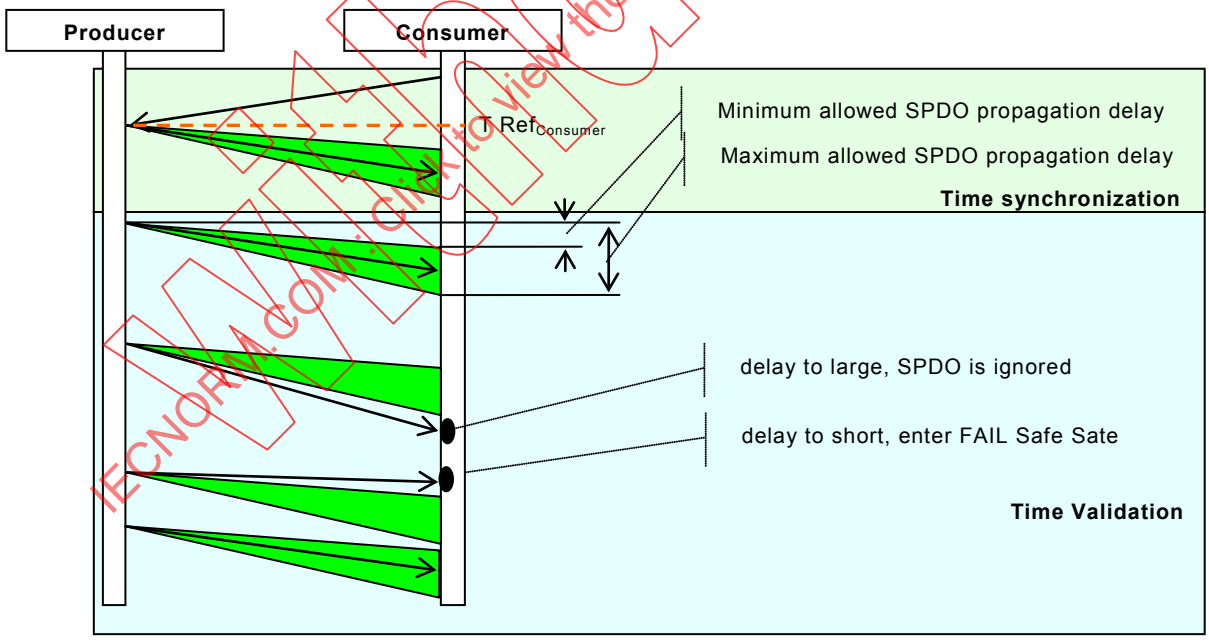
Figure 50 shows how *SPDO propagation delay* is calculated.



Key
See Table 178.

Figure 50 – Calculation of propagation delay

Figure 51 specifies the time validation sequences.



Key
See Table 178.

Figure 51 – Time validation, propagation delay explanation limits

Table 178 specifies items used in Figure 50 and Figure 51.

Table 178 – Time validation item description

Item	Description	Min value	Max value	SOD
T $SPDO_{Producer}$	Value of CT field within the SPDO supplied by the producer	0	65 535	-
T $SPDO_{Consumer}$	CT value within the consumer when receiving the SPDO	0	65 535	-
SPDO propagation delay	Calculated propagation delay using the time references of the previous time synchronization, see Equation (2)	0	65 535	-
Minimum allowed SPDO propagation delay	Minimum allowed propagation delay for SPDO [CT]	1	65 535	Object index 1400h -- 17FEh sub index 8h (see 7.5.4.14)
Maximum allowed SPDO propagation delay	Maximum allowed propagation delay for SPDO [CT]	1	65 535	Object index 1400h -- 17FEh sub index 9h (see 7.5.4.14)

7.7.2.4 Time synchronization operation

To increase the immunity of FSCP 13/1 against data loss within the non safe communication layer, the TReq and TRes services shall be sent as bundle of single services.

The consumer shall send m time requests (TxSPDO) to the producer containing the TR (Time Request Distinctive Number). This number shall be incremented with each time request. The producer shall receive the time request from a specific node and shall start answering the time request. The response to a time request shall be to fill the TADR (SADR of the requesting node) and the TR field in the cyclic TxSPDO telegram from the producer.

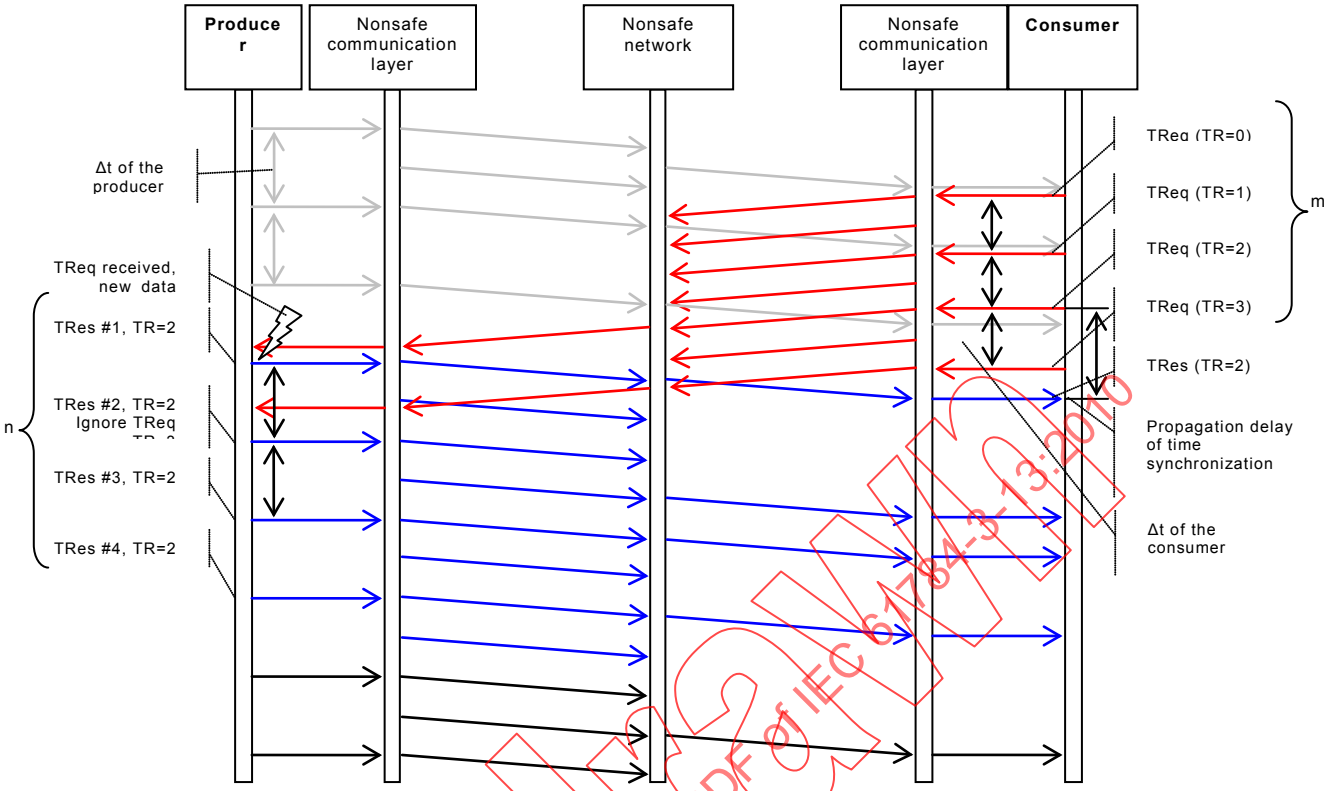
The producer shall repeat the response for a received time request n times although new time requests were already received. This ensures that the consumer which initiates the time request receives the correct answer. During processing the Time Response, the producer shall ignore all other time requests corresponding to this TxSPDO.

If the consumer does not receive the time response within time td (see Table 179), the next set of time requests shall be sent. The number of time responses per time request, the number of requests per consecutive request set, td and the timer ts (see Table 179) which re-initiates a time request shall be adjusted within the SOD. If the propagation delay is too long the TRes shall be ignored, if it is too short, the safe state shall be entered (see 7.7.2.1). If a valid TRes was not received within the Time request cycle, a time synchronization failure shall occur.

The time synchronization for each producer shall start when the consumer is Operational and receives the first RxSPDO from the corresponding producer.

Within a Consumer / Producer relationship, equal CT values shall be mandatory.

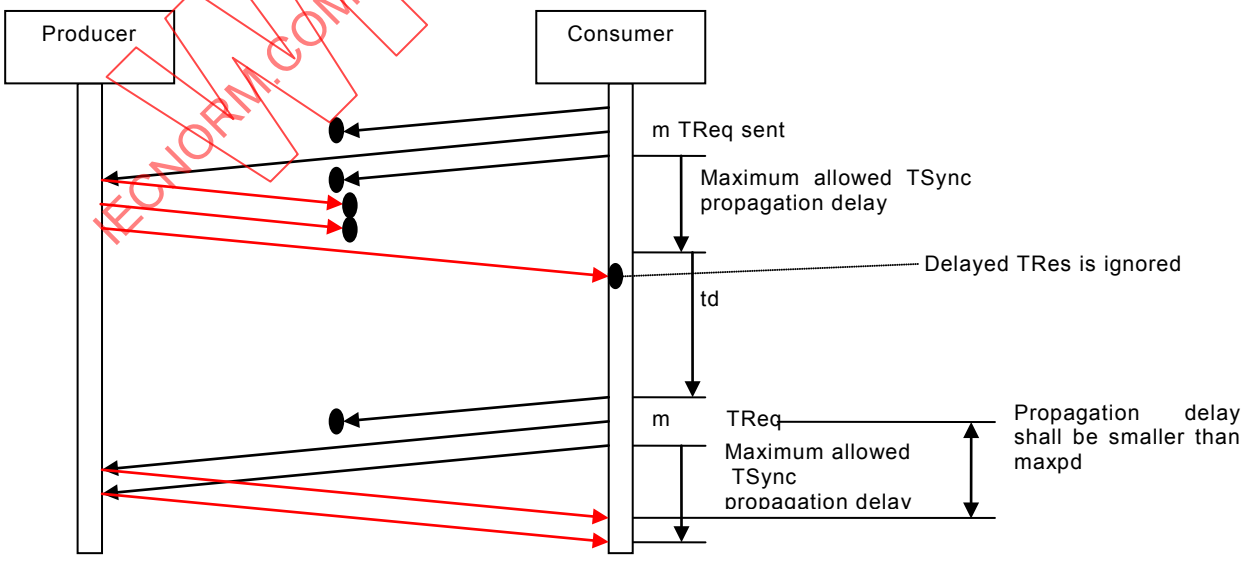
Figure 52 specifies time request and response. The consumer shall send a time request to one of its producers. This request is repeated m times. The producer shall answer the first time request, and it receives and repeats the answer n times. If a producer receives a time request during the time it is already answering another time request on the corresponding TxSPDO, the new time request shall be ignored. Figure 52 also shows that the nonsafe communication layer may also drop or delay messages due to different network cycle times or other effects.



Key
See Table 179.

Figure 52 – Time synchronization on a nonsafe network

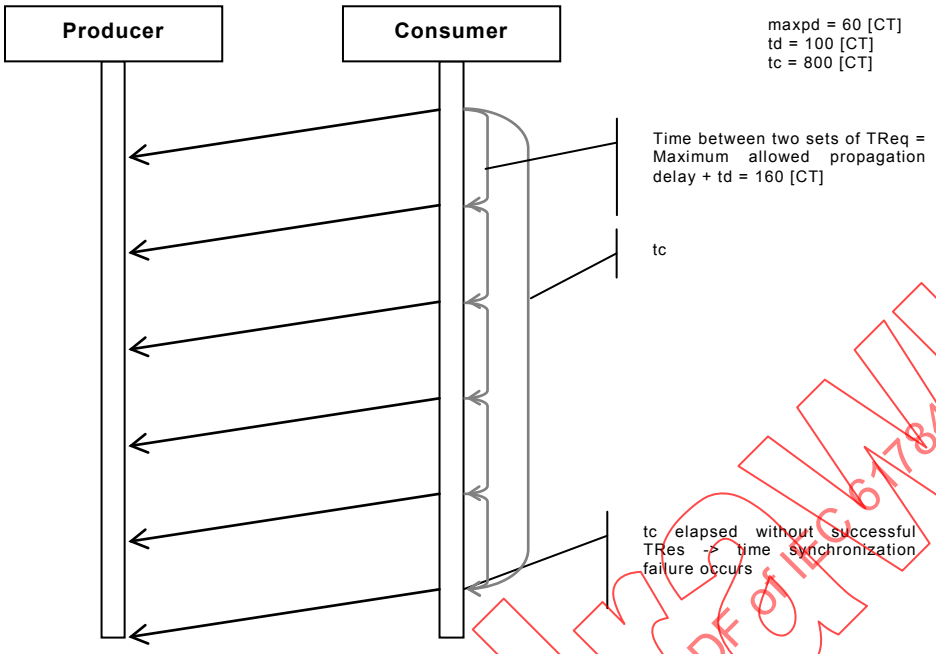
Time synchronization is specified by Figure 53. The consumer shall send a set of m time requests (TReq) and then wait for the response. If the response time is shorter than the minimum allowed propagation delay, the safe state shall be entered. If the response (TRes) is within the maximum and minimum allowed propagation delay, it shall be valid. If no response is received within the maximum allowed propagation delay, the consumer shall wait until td has elapsed and shall send the next set of m TReq.



Key
See Table 179.

Figure 53 – Explanation of time synchronization

Figure 54 specifies a time synchronization failure. The consumer sends time requests to the producer. If a valid time response is not received within the entire time synchronization cycle time, a synchronization failure shall occur. This also ensures that the related outputs will enter the safe state.



Key
See Table 179.

Figure 54 – Time synchronization failure

Table 179 specifies items used in Figure 52, Figure 53 and Figure 54.

Table 179 – Extended time synchronization item description

Item	Description	Min value	Max value	SOD
n	Number of TRes from Producer	1	255	Object index 1C00h -- 1FFEh sub index 03h (see 7.5.4.16)
TR	Time Request Distinctive Number Consumer: To be incremented each TReq	0	63	-
m	Number of consecutive TReq from consumer	1	63	Object index 1400h -- 17FEh sub index 03h (see 7.5.4.14)
td	Time delay between two TReq blocks [CT]	0	UNSIGNED32	Object index 1400h -- 17FEh sub index 04h (see 7.5.4.14)
ts	Time delay for new synchronization [CT]. The delay between successful time synchronization and the next TReq	1	UNSIGNED32	Object index 1400h -- 17FEh sub index 05h (see 7.5.4.14)
tc	Time request cycle [CT]	1	UNSIGNED32	Object index 1400h -- 17FEh sub index 0Bh (see 7.5.4.14)
maxpd	Maximum allowed TSync propagation delay	1	65535	Object index 1400h -- 17FEh sub index 07h (see 7.5.4.14)

7.7.2.5 Time synchronization frequency

The frequency of the time synchronization is calculated by Equation (5).

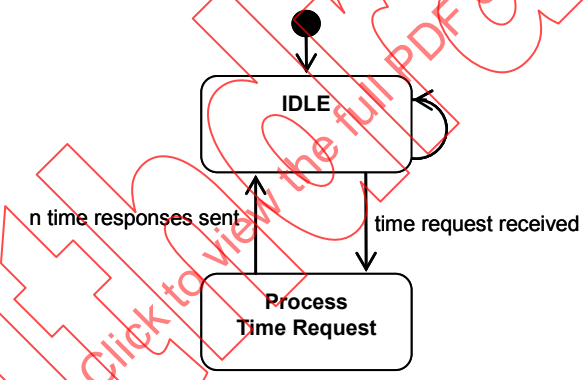
$$TimeRequestCycle\,[\mathrm{CT}] = \frac{SCT\,[\mathrm{CT}] - Max.\ allowed\ SPDO\ PropagationDelay\,[\mathrm{CT}]}{TimeAccuracy\,[1]} \quad (5)$$

where

Element	Min value	Max value	SOD
Time request cycle [CT]	1	UNSIGNED32	Object index 1400h -- 17FEh sub index Bh (see 7.5.4.14)
SCT – Safety control time [CT]	1	UNSIGNED32	Object 1400h -- 17FEh sub index 2h (see 7.5.4.14)
Max. allowed SPDO propagation delay [CT]	1	65 535	Object index 1400h -- 17FEh sub index 9h (see 7.5.4.14)
TimeAccuracy: Overall Time Accuracy of Consumer and Producer, this term is mainly determined by the system, the quartz crystal, etc.	-	-	-

7.7.2.6 Time synchronization producer

The time synchronization producer state diagram is specified by Figure 55.



Key
See Table 180 and Table 181.

Figure 55 – State diagram time synchronization producer

Table 180 specifies items used in Figure 55.

Table 180 – Time synchronization producer item description

Item	Description	Min value	Max value	SOD
n	Number of TRes from producer	1	255	Object index 1C00h -- 1FFEh sub index 03h (see 7.5.4.16)

Table 181 specifies the time synchronization producer states.

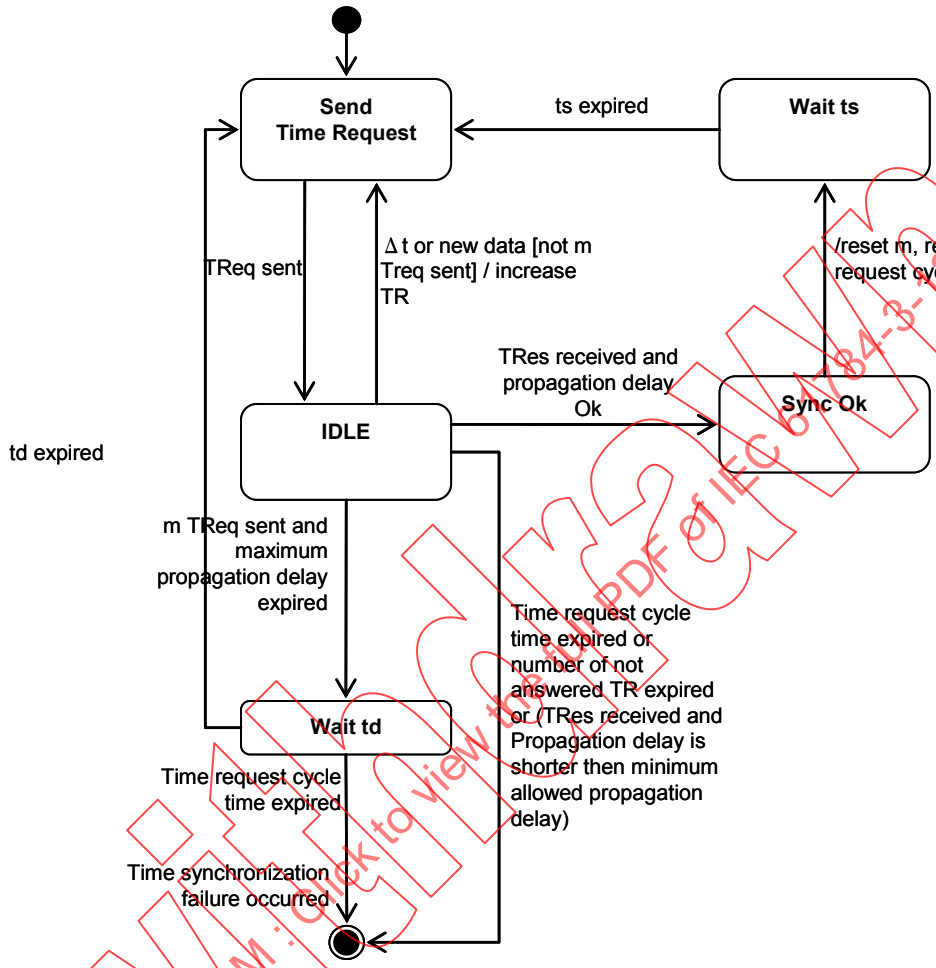
Table 181 – Time synchronization producer state description

State	Description
IDLE	The producer waits for a time request from any consumer
Process Time Request	After a time request from a consumer, the producer starts answering this request

NOTE The response to a Time Request is not a separate telegram. It is just part of the normal TxSPDO.

7.7.2.7 Time synchronization consumer

The time synchronization consumer state diagram is specified by Figure 56.



Key
See Table 182 and Table 183.

Figure 56 – State diagram time synchronization consumer

Table 182 specifies items used in Figure 56.

Table 182 – Time synchronization consumer item description

Item	Description	Min value	Max value	SOD
TReq	Time synchronization request	-	-	-
m	Number of consecutive TReq from consumer	1	63	Object index 1400h -- 17FEh sub index 03h (see 7.5.4.14)
TR	Time Request Distinctive Number Field	0	63	-
Δt	TxSPDO refresh prescale [CT]	1	32 767	Object index 1400h -- 17FEh sub index 02h (see 7.5.4.14)
td	Time delay between two TReq blocks [CT]	0	UNSIGNED32	Object index 1400h -- 17FEh sub index 04h (see 7.5.4.14)

Item	Description	Min value	Max value	SOD
ts	Time delay for new synchronization [CT]. The delay between successful time synchronization and the next TReq	1	UNSIGNED32	Object index 1400h -- 17FEh sub index 05h (see 7.5.4.14)
minpd	Minimum allowed propagation delay for time synchronization [CT]	1	65 535	Object index 1400h -- 17FEh sub index 06h (see 7.5.4.14)
maxpd	Maximum allowed propagation delay for time synchronization [CT]	1	65 535	Object index 1400h -- 17FEh sub index 07h (see 7.5.4.14)
tc	Time request cycle [CT]	1	UNSIGNED32	Object index 1400h -- 17FEh sub index 0Bh (see 7.5.4.14)

Table 183 specifies the time synchronization consumer states.

Table 183 – Time synchronization consumer state description

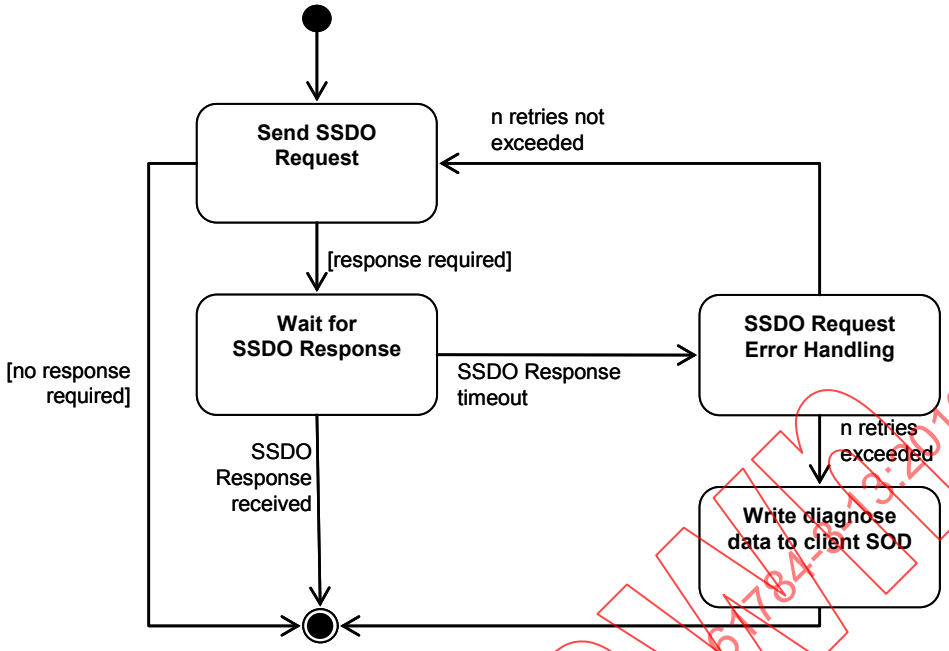
State	Description
Send Time Request	Sending the time request to the corresponding producer
IDLE	Waiting for TRes
Wait td	Waiting until td expired. All received TRes are ignored
Wait ts	Waiting until ts expired. All received TRes are ignored
Sync Ok	Synchronization Ok. Reset m: reset time request cycle time. Reset number of not answered TR

7.7.3 Safety Service Data Object (SSDO)

7.7.3.1 SSDO client

The client shall send an SSDO request telegram to the server. If a response is received or a response is not required, the client shall continue (more SSDO telegrams, etc.). If the response is timed out, the client shall retry to send the telegram n times. The number of retries as well as the timeout time for an SSDO Response telegram shall be set within the SOD of the client.

The SSDO client state diagram is specified by Figure 57.



Key
See Table 184 and Table 185.

Figure 57 – State diagram SSDO client

Table 184 specifies items used in Figure 57.

Table 184 – SSDO client item description

Item	Description	Min value	Max value	SOD
n	Max. count of retries	0	UNSIGNED8	Object index 1201h sub index 02h (see 7.5.4.12)
	Timeout of SSDO Response [CT]	1	UNSIGNED32	Object index 1201h sub index 01h (see 7.5.4.14)

Table 185 specifies the SSDO client states.

Table 185 – SSDO client state description

State	Description
Send SSDO Request	Sending the SSDO telegram to the server
Wait for SSDO Response	Waiting for the response from the server
SSDO Request Error Handling	If the maximum number of retries was not reached, the telegram will be sent again otherwise a communication error occurs
Write diagnose data to client SOD	The diagnose data has to be available for the user. This is made through the SOD of the client

7.7.3.2 SSDO server

If the server receives an SSDO request telegram from the client, the telegram shall be processed according to the ID field. During processing, an SSDO response shall be generated and sent back to the client.

The SSDO server state diagram is specified by Figure 58.

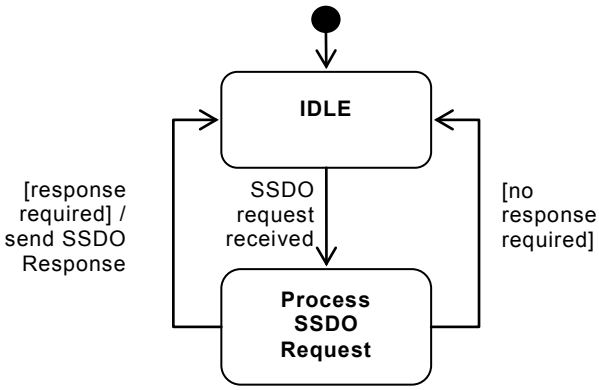


Figure 58 – State diagram SSDO server

Table 186 specifies the SSDO server states.

Table 186 – SSDO server state description

State	Description
IDLE	Waiting for SSDO Access Request telegrams
Process SSDO Request	Process the telegram according to the SSDO Access Command. Generates SSDO Response

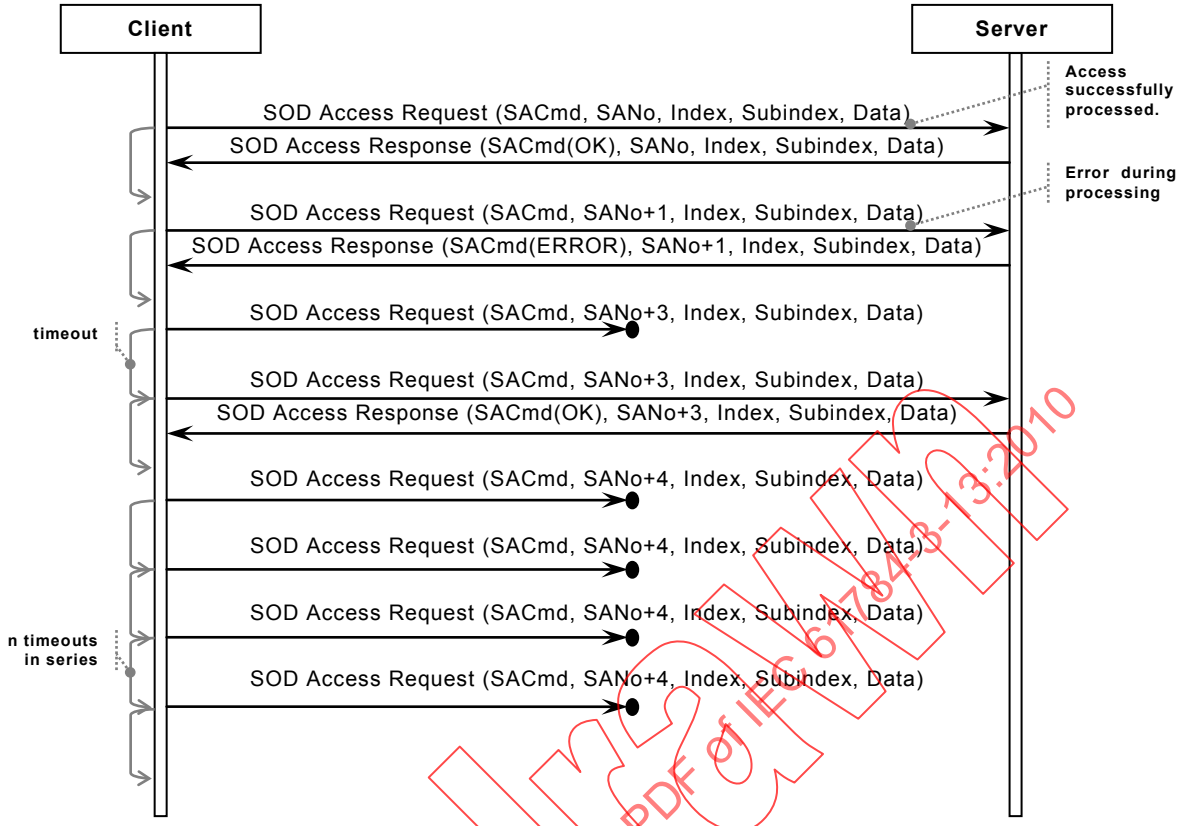
7.7.4 SOD access

7.7.4.1 General

The SOD Access shall be based on the SSDO services. The SSDO service shall be used by the client to access the SOD (Safety Object Dictionary) within the server. Write access to the SOD shall be only permitted in Pre-Operational state.

7.7.4.2 SOD access (expedited)

The SOD access (expedited) sequence diagram is specified by Figure 59.



Key
See Table 187.

Figure 59 – Expedited SOD access

Table 187 specifies items used in Figure 59.

Table 187 – SOD access item description

Item	Description	Min value	Max value	SOD
n	Max. count of retries in series	0	UNSIGNED8	Object index 1201h sub index 02h (see 7.5.4.12)
Timeout	Timeout of SOD Access response	1	UNSIGNED32	Object index 1201h sub index 01h (see 7.5.4.12)

7.7.4.3 SOD download access with segmentation

7.7.4.3.1 Client state diagram

Segmented SOD Access shall be used to download large amounts of data to the SN (e.g. the entire SOD during parameterization). It shall consist of the initialization telegram, the segment download telegram and the end telegram. Each telegram requires a special response. The initialization telegram shall carry the size of the data to be downloaded as payload. The segment download telegrams shall carry data only. The end telegram shall carry the last data segment as payload. If one telegram fails, the entire download shall be repeated.

The segmented SOD download access client state diagram is specified by Figure 60.

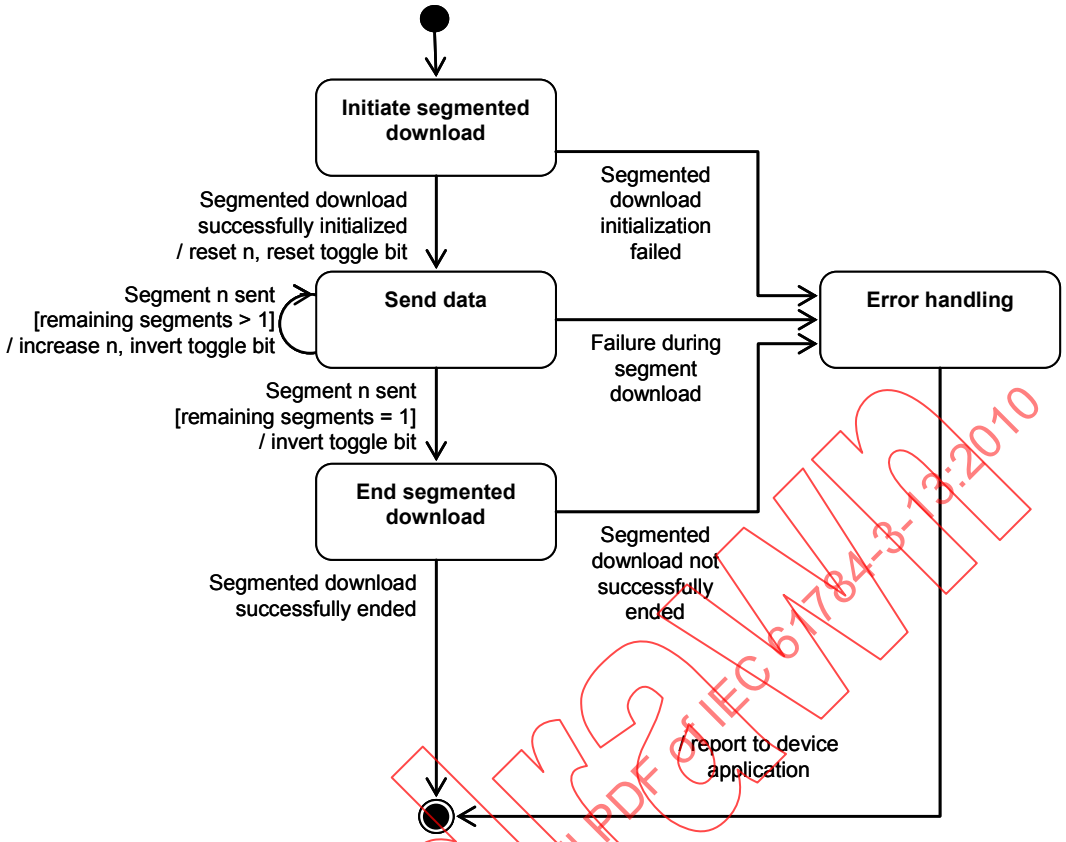
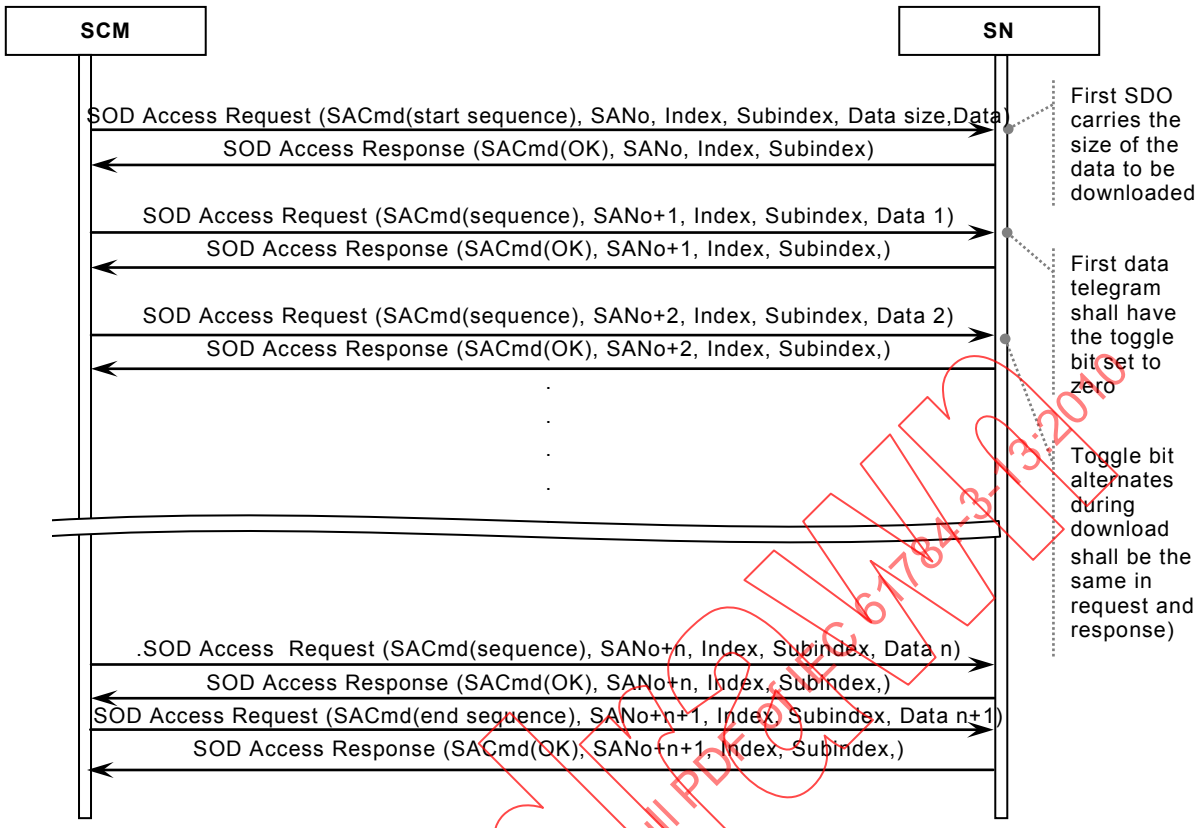


Figure 60 – State diagram segmented SOD download access client

The segmented SOD download access sequence diagram is specified by Figure 61.



Key
See Table 188.

Figure 61 – Segmented SOD download access

Table 188 specifies items used in Figure 61.

Table 188 – Segmented SOD access client item description

Item	Description	Min value	Max value	SOD
n	current segment number	-	-	-
Data size	4 octet value which identifies the size of data being downloaded with the segmented SOD Access	1	UNSIGNED32	-

Table 189 specifies the segmented SOD download access client states.

Table 189 – Segmented SOD download access client state description

State	Description
Initiate sequenced download	Starts the SOD Access with the initialization telegram
Send data	Starts the SOD Access with a segment download telegram
End sequenced download	Starts the SOD Access with the end download telegram
Error handling	Reports the communication error to the application

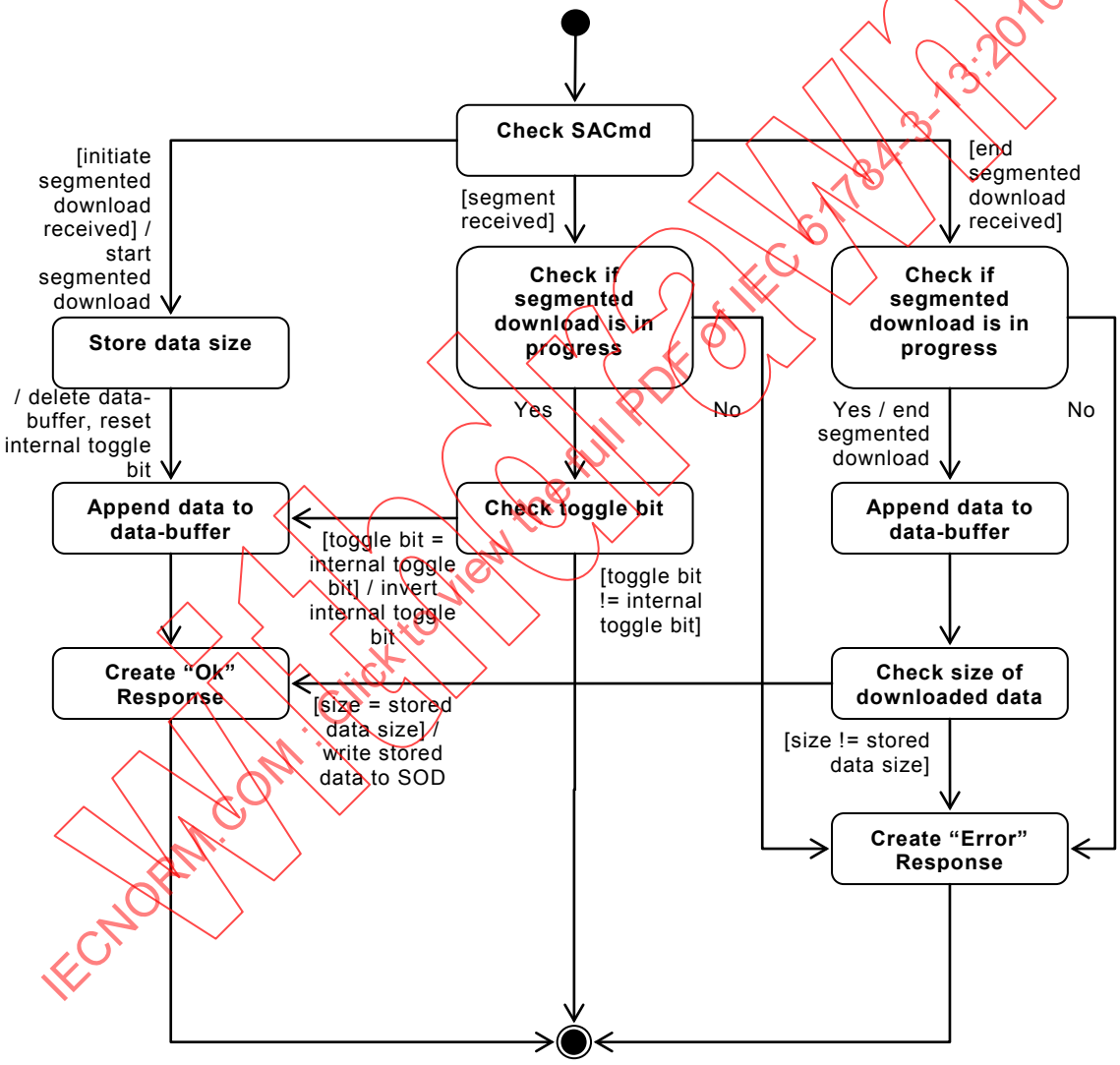
7.7.4.3.2 Server state diagram

The server which receives the data via the segmented download shall verify the correctness of the telegrams. If a download is initiated, the server shall store the data size which shall be received during the entire download.

When the download is complete, this data size shall be cross-checked with the size of the received data. If the received data is not equal to the expected data size, the server shall respond with an error telegram. If the server receives “segment” or “end segmented download” telegrams, it also shall check if the download was initiated. Otherwise, the data shall be ignored and an error response shall be generated.

During the segmented download, the toggle bit of the “segment” telegrams shall be inverted for each new telegram. If the server receives two or more consecutive telegrams with the same toggle bit, only the data of the first telegram shall be stored, all other telegrams shall be ignored.

The Segmented SOD Download Access Server state diagram is specified by Figure 62.



Key
See Table 190 and Table 191.

Figure 62 – State diagram segmented SOD download access server

Table 190 specifies items used in Figure 62.

Table 190 – Segmented SOD access server item description

Item	Description	Min value	Max value	SOD
toggle bit	The toggle bit is used to identify new segments	0	1	-
data size	4 octet value which identifies the size of data being downloaded with the segmented SOD Access	1	UNSIGNED32	-
size	Size of the received data	0	UNSIGNED32	-
SACmd	Safety Access Command	-	-	-

Table 191 specifies the segmented SOD access server states.

Table 191 – Segmented SOD access server state description

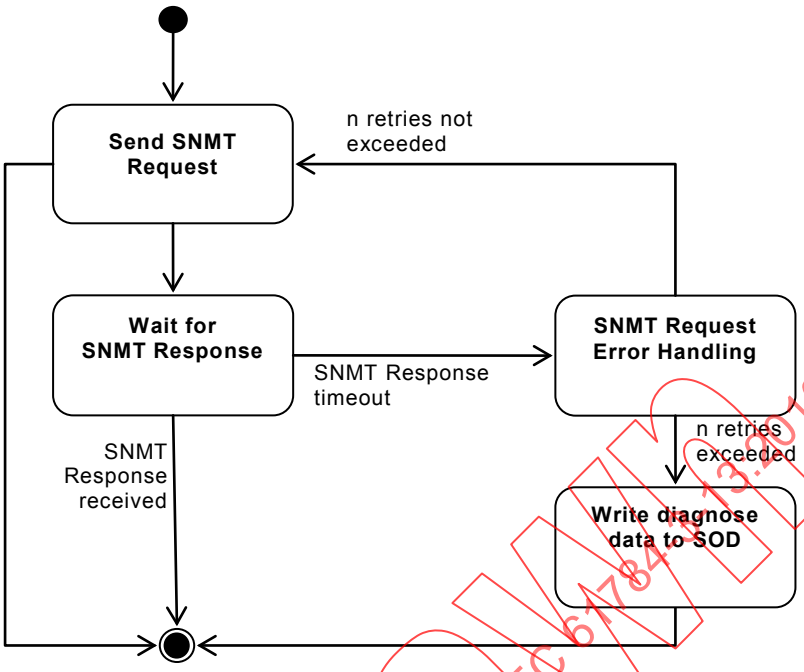
State	Description
Check SACmd	Checks which type of telegram was received
Store data size	Stores the expected data size for cross-check
Append data to data-buffer	Writes the received data into a temporary buffer
Create “Ok” Response	If the telegram was valid, the server sends a response to the client which indicates that the telegram was processed
Check if segmented download is in progress	If a “segment” or “end segmented download” telegram was received, the server has to check if the segmented download was initialized
Check toggle bit	Each “segment” telegram carries a toggle bit which has to be inverted for each new telegram
Check size of downloaded data	At the end of the segmented download, the server checks if the expected data size is equal to the received data size. If not, an error response is generated
Create “Error” response	Creates an error response which tells the client that the segmented download failed

7.7.5 Safety Network Management Object (SNMT)

7.7.5.1 SNMT master

The master shall send an SNMT Request telegram to the slave. If a response is received or a response is not required, the client may continue (more SNMT telegrams, etc.). If the response times out, the client shall retry to send the telegram n times. The number of retries as well as the timeout time for an SNMT Response telegram shall be set within the SOD of the client.

The SNMT master state diagram is specified by Figure 63.



Key
See Table 192 and Table 193.

Figure 63 – State diagram SNMT master

Table 192 specifies items used in Figure 63.

Table 192 – SNMT master item description

Item	Description	Min value	Max value	SOD
n	Max. count of retries	0	UNSIGNED8	Object index 1202h sub index 02h (see 7.5.4.13)
	Timeout of SNMT Response [CT]	1	UNSIGNED32	Object index 1202h sub index 01h (see 7.5.4.13)

Table 193 specifies the SNMT master server states.

Table 193 – SNMT master state description

State	Description
Send SNMT Request	Sending the SNMT telegram to the slave
Wait for SNMT Response	Waiting for the response from the server
SNMT Request Error Handling	If the maximum number of retries was not reached the telegram will be sent again otherwise a communication error occurs
Write diagnose data to SOD	The diagnose data has to be available for the user. This is made through the SOD of the master

7.7.5.2 SNMT slave

The slave receives an SNMT request telegram from the master. The telegram shall be processed according to the ID field. During processing, an SNMT Response shall be generated and sent back to the master.

The SNMT slave state diagram is specified by Figure 64.

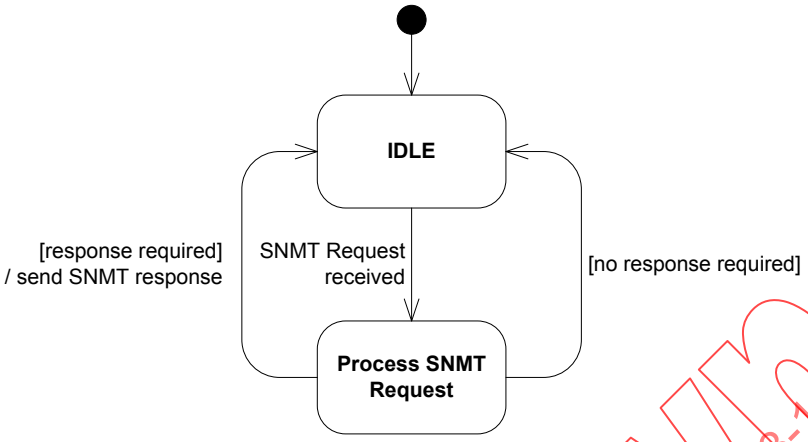


Figure 64 – State diagram SNMT slave

Table 194 specifies the SNMT slave states.

Table 194 – SNMT slave state description

State	Description
IDLE	Wait for SNMT Request telegrams
Process SNMT Request	Process the telegram according to the SNMT Request. Generates SNMT Response

7.7.6 SN power up

7.7.6.1 General

At power up, the SN shall start the self initialization. An SN may contain a flash memory where all data is stored. This data shall be loaded during the self initialization. After this, the SN shall enter the Pre-Operational state. In this state the SN may get parameters from the SCM. The SN also shall send cyclic messages to the SCM to inform the SCM about its status. When the SN switches from Pre-Operational to Operational, it shall store all parameters within the flash memory (this shall be an optional feature).

The SN power up state diagram is specified by Figure 65.

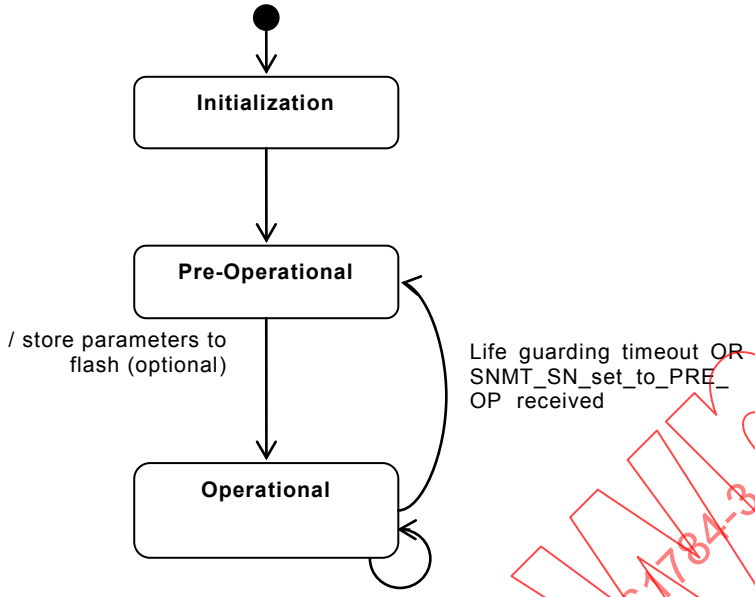


Figure 65 – State diagram SN power up

Table 195 specifies the SN power up states.

Table 195 – SN power up state description

State	Description
Initialization	Loading parameters from flash (optional)
Pre-Operational	Receiving parameters from SCM, UDID verification, etc. In this state all application data are set to the safe state (default of the object, see 7.5.2.9)
Operational	Sending and receiving SPDOs, time requests, etc. To keep the SN in Operational, it needs to receive a cyclic signal from the SCM. If the signal is timed out (timeout to be set within the SOD), the SN switches to Pre-Operational

The states Initialization, Pre-Operational and Operational are also referred to as communication states.

7.7.6.2 States and communication object relation

Table 196 specifies the relation between communication states and communication objects. Services on the listed communication objects shall only be executed if the device is in the appropriate Operational state.

Table 196 – State and communication object relation

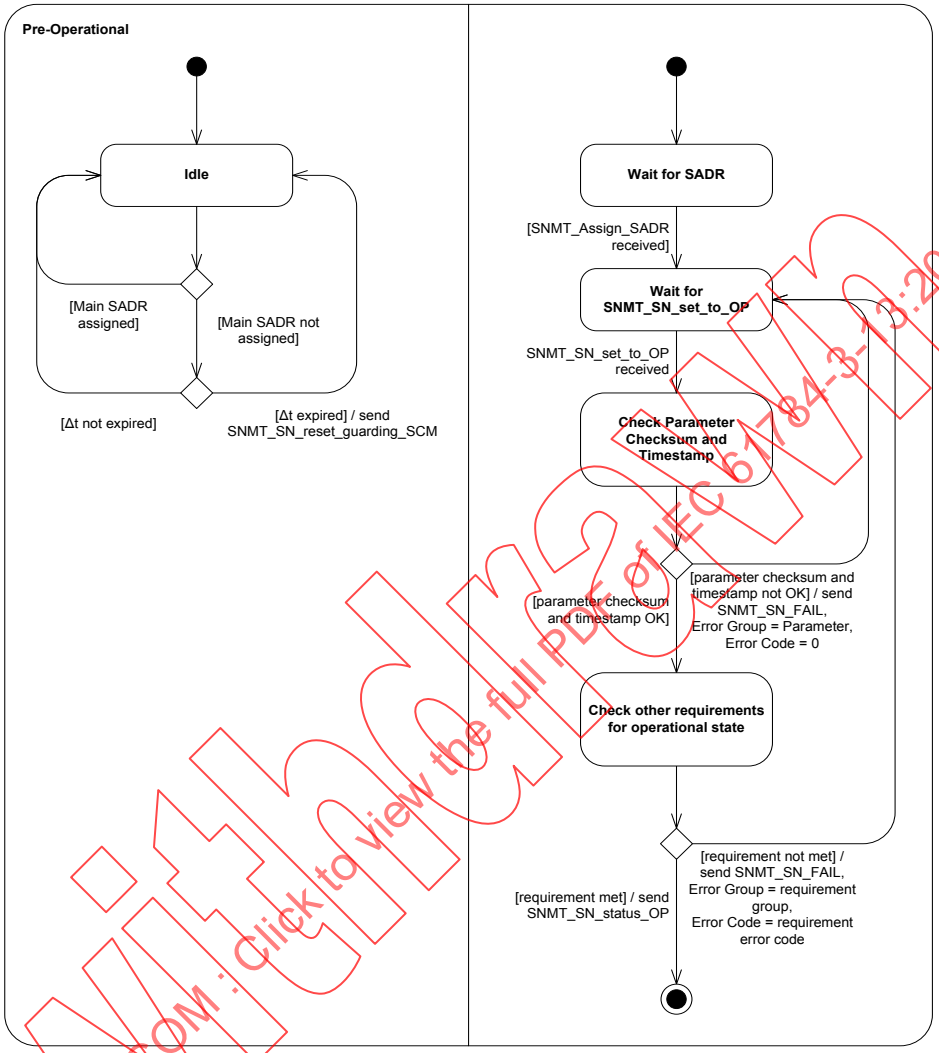
	Initialization	Pre-Operational	Operational
SPDO			X
SSDO		X	X
SNMT		X	X

7.7.6.3 Pre-Operational

When the SN enters the Pre-Operational state, the SNMT and SOD Access Rx services shall be started. In this state all application data shall be set to the safe state. The SNMT services are required for several purposes, e.g. receiving UDID request telegrams. With SOD Access

Commands, the SN may get new parameters from the SCM. Before being able to switch to Operational, the SN at least shall receive its “Main SADR” and the “UDID of the SCM”.

The SN Pre-Operational state diagram is specified by Figure 66.



Key
See Table 197 and Table 198.

Figure 66 – State diagram SN Pre-Operational

Table 197 specifies items used in Figure 66.

Table 197 – SN Pre-Operational state item description

Item	Description	Min value	Max value	SOD
Δt	Refresh time of SNMT_SN_reset_guarding_SCM signal [µs]	1	UNSIGNED32	Object index 100Dh (see 7.5.4.6)

Table 198 specifies the SN Pre-Operational states.

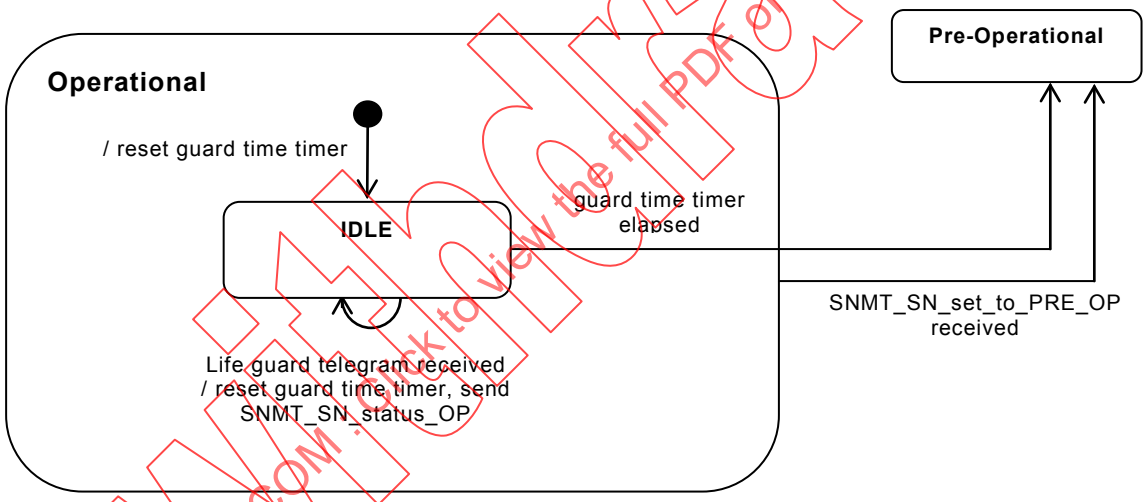
Table 198 – SN Pre-Operational state description

State	Description
Idle	Receiving parameters from SCM. UDID verification. Etc.
Wait for SADR	The SN waits to get its Main SADR from the SCM
Wait for SNMT_SN_set_to_OP	The SN waits to be switched to Operational from the SCM
Check Parameter Checksum and Timestamp	Checks if the parameter checksum as well as the parameter timestamp match the local settings. In case the checksum calculation takes a long time, the SN may reply to the SCM using the service SNMT_SN_busy. The SCM then has to send the service SNMT_SN_set_to_OP again
Check other requirements for Operational state	There might be several reasons for not being able to switch to Operational state. Within this state, the module firmware is requested if switching to Operational is allowed. If not the reason is reported to the SCM

7.7.6.4 Operational

When the SN enters the Operational state, Life Guarding shall be started. The SN needs to receive cyclic telegrams from the SCM to remain in Operational state. If this telegram is timed out, the SN shall fall back into Pre-Operational state.

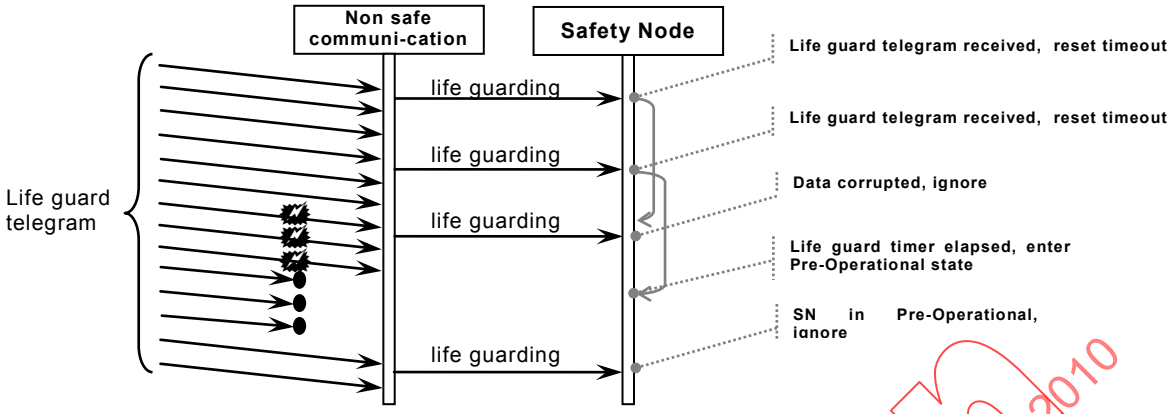
The SN Operational state diagram is specified by Figure 67.



Key
See Table 199 and Table 200.

Figure 67 – State diagram SN Operational

The Life Guarding sequence diagram is specified by Figure 68.



Key
See Table 199.

Figure 68 – Life Guarding telegram

Table 199 specifies items used in Figure 67 and Figure 68.

Table 199 – SN Operational state item description

Item	Description	Min value	Max value	SOD
	Guard time timer	-	-	Object index 100Ch sub index 01h and 02h (see 7.5.4.5)

Table 200 specifies the SN Operational states.

Table 200 – SN Operational state description

State	Description
IDLE	Waits for Life Guarding telegram

7.7.7 SN power down

All safety outputs shall be set to safe state (0 V).

7.7.8 SN recovery after Restart / Error

If an error occurred or the SN was restarted, the SN shall enter the Pre-Operational state. In this state, the SN shall send a cyclic message (with no response) to the SCM. The SCM now shall check the SN again, verify all parameters, UDID, DVI and then set the SN to Operational again.

7.7.9 SCM power up

7.7.9.1 General

After power-on, the SCM loads its parameters from a non-volatile memory. Then it switches to Operational. The Operational super-state is described later on. The SCM can be switched into the Stopped state using an SNMT service. In Stopped, the SCM does not manage any safety nodes. If safety nodes are being configured with an external tool, the SCM shall be in state Stopped.

The SCM power up state diagram is specified by Figure 69.

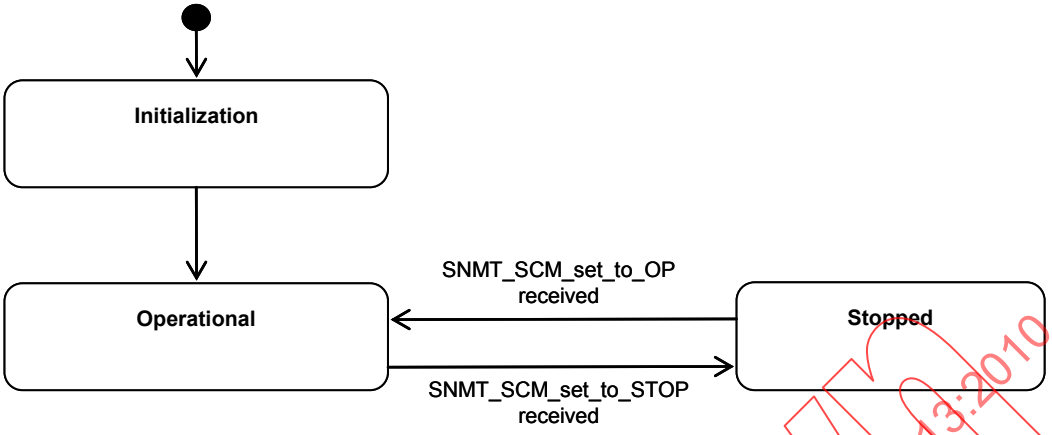


Figure 69 – State diagram SCM power up

Table 201 specifies the SCM power up states.

Table 201 – SCM power up state description

State	Description
Initialization	The SCM loads its parameters from a non-volatile memory
Operational	In Operational state all safety nodes are configured and started
Stopped	The SCM waits for the signal to switch back to Operational

7.7.9.2 States and communication object relation

Table 202 specifies the relationship between communication states and communication objects. Services on the listed communication objects shall only be executed if the device is in the appropriate communication state.

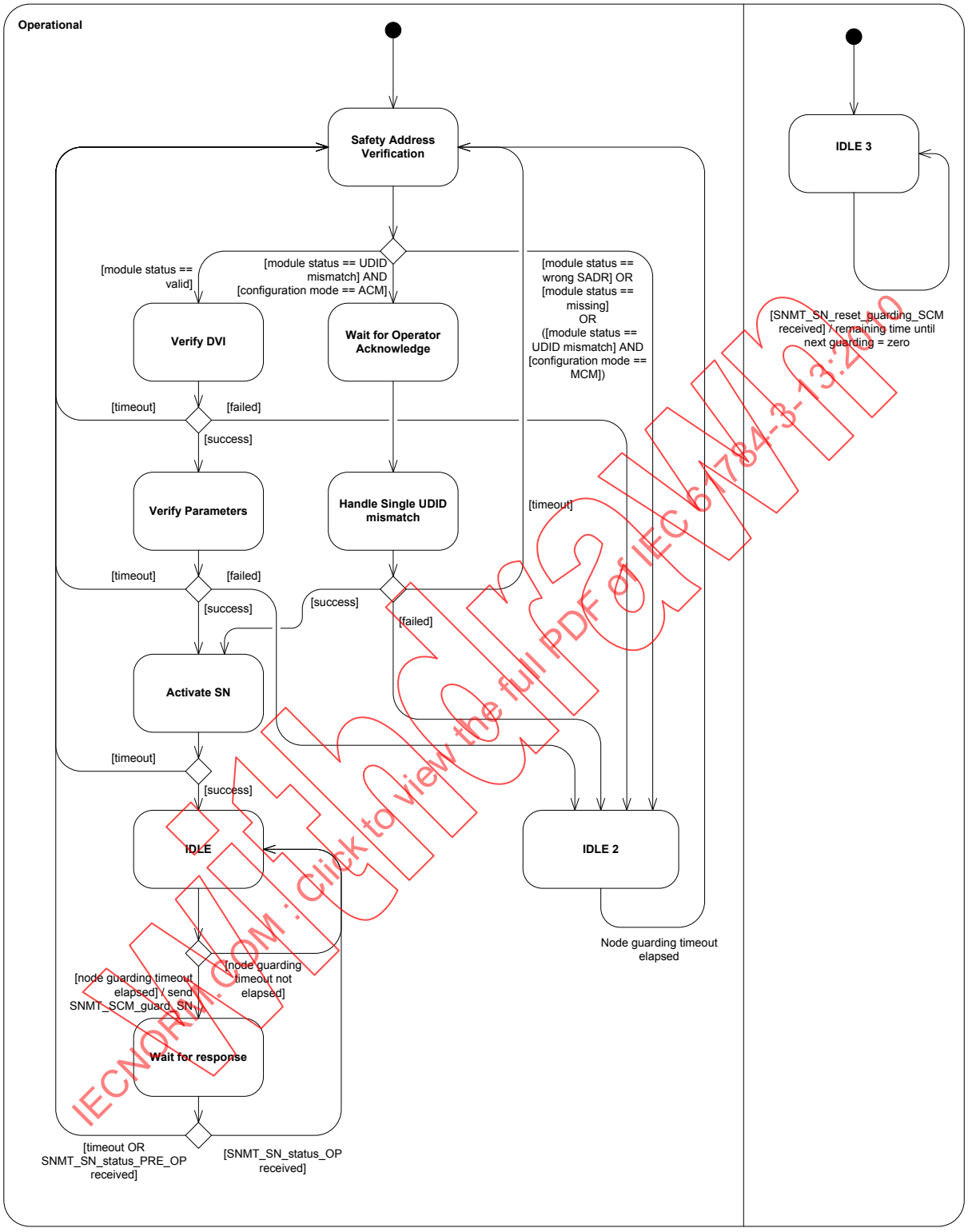
Table 202 – State and communication object relation

	Initialization	Stopped	Operational
SSDO		X	X

7.7.9.3 Operational

When the SCM enters state Operational, it shall start the UDID/SADR verification of all configured nodes. A UDID mismatch shall be handled separately. The parameters for all valid nodes shall be verified against the list of CRCs and Timestamps within the SOD (Object indices C400h - C7FEh). If a node is valid (correct UDID, SADR and DVI) and its parameters are also valid, it shall be set to Operational. Operational nodes get a cyclic message to keep them in Operational state.

The “SCM Operational” state diagram is performed for each SN in parallel and is specified by Figure 70.



Key
See Table 203 and Table 204.

Figure 70 – State diagram SCM Operational

Table 203 specifies items used in Figure 70.

Table 203 – SCM Operational state item description

Item	Description	Min value	Max value	SOD
	Guarding Timeout	1	UNSIGNED32	Object index 100Ch sub index 01h (see 7.5.4.5)

Table 204 specifies the SCM Operational states.

Table 204 – SCM Operational state description

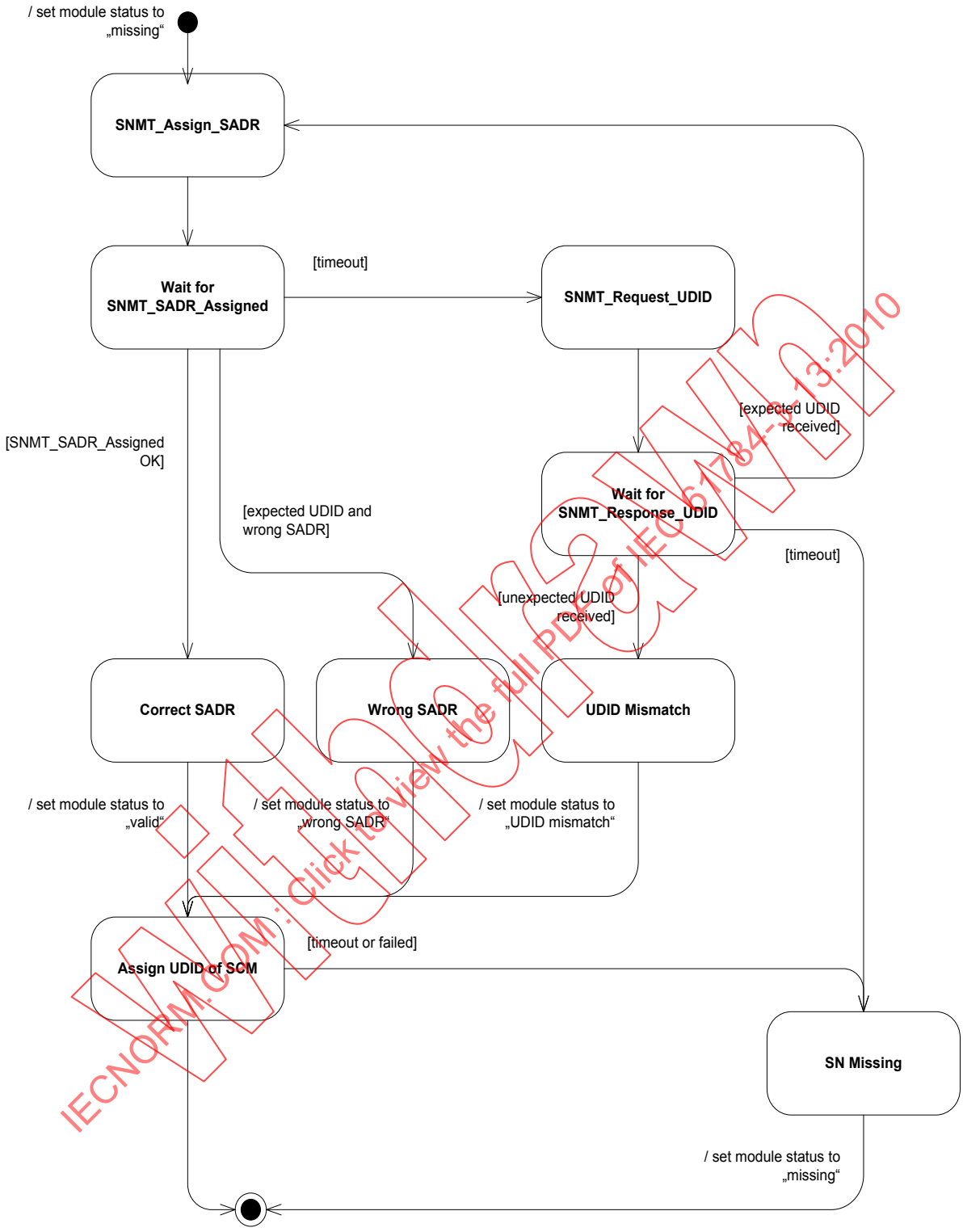
State	Description
Safety Address Verification	The SCM verifies the UDID of the node. Also the correct SADR will be assigned to the node
Verify DVI	The DVI has to be checked to ensure that the new node meets the technical requirements
Verify Parameters	Parameter check and eventually re-parameterization of the SN
Activate SN	After successful parameterization, the SCM switches the SN to Operational
IDLE	Waits until next SNMT_SCM_guard _SN needs to be sent
Wait for response	Waits for the response of the SNMT_SCM_guard_SN service
Wait for Operator Acknowledge	If an unknown UDID was detected during start-up or operation, an operator-acknowledge is needed to accept and verify the new node. This state is only entered in Automatic Configuration Mode (ACM)
Handle Single UDID Mismatch	If a new SN was detected (replacement, commissioning, etc.), it is verified and checked to see if it meets the requirements or not
IDLE 2	If the node is not present or does not meet the technical requirements (module was replaced with a wrong type), the node will not be handled for a node guarding interval. This ensures, that nodes, which are not present or nodes which were replaced with a different module type will get handled again, but do not produce too much bus activity
IDLE 3	Sets the remaining time until the next node guarding to zero in case an SNMT_SN_reset_guarding_SCM telegram is received. After a power up of a module, the module sends this type of telegram to the SCM. This ensures that the SCM recognizes (by the response to the guarding telegram) that the module is down and needs to be restarted

7.7.10 Address verification

The Address verification shall be used to verify the correctness of the safety network. All device configuration changes are detected due to the uniqueness of the UDID.

This state machine shall apply in any configuration mode (ACM or MCM).

The “SCM address verification” state diagram is specified by Figure 71.



Key
See Table 205 and Table 206.

Figure 71 – State diagram SCM address verification

Table 205 specifies items used in Figure 71.

Table 205 – Address verification item description

Item	Description	Min value	Max value	SOD
	module status 0...missing 1...invalid 2...wrong SADR 3...UDID mismatch 4...wrong parameters 5...valid 6...OK	0	5	Object index C400h -- C7FEh sub index 05h (see 7.5.4.18)
	expected UDID	1	OCTET_STRING8	Object index CC01h -- CCFFh sub index 1 -- 253 (see 7.5.4.20)
	assigned SADR	1	1 023	Object index CC01h -- CCFFh sub index 1 -- 253 (see 7.5.4.20)

Table 206 specifies the SCM address verification states.

Table 206 – Address verification state description

State	Description
SNMT_Assign_SADR	Assigns the SADR to a configured node
Wait for SNMT_SADR_Assigned	The response contains the actual SADR. Some nodes might not be able to change the SADR because it is to be set using hardware switches
SNMT_Request_UDID	Requests the UDID of a configured node
Wait for SNMT_Response_UDID	The response from the node contains its UDID
Correct SADR	The SADR could be assigned. The node is valid
Wrong SADR	The UDID is known but the SADR could not be assigned. The node is invalid
UDID Mismatch	The node does not have the expected UDID. An operator-acknowledge is needed to accept and start the node
SN missing	The configured SN could not be reached. The node might be defective or not in the network. The node is missing
Assign UDID of SCM	Sends UDID of SCM to sn via the corresponding service

7.7.11 Commissioning mode

FSCP 13/1 defines no explicit commissioning mode. The first time the SCM is started, all nodes shall be handled as nodes with a UDID mismatch.

7.7.12 Handle single UDID mismatch

7.7.12.1 General

This subclause specifies the handling of nodes with UDID mismatch. If the non matching node (e.g. new node) meets the requirements (DVI, assignment of SADR, etc.), no re-configuration of the SCM shall be needed when replacing nodes.

The “SCM handle single UDID mismatch” state diagram is specified by Figure 72.

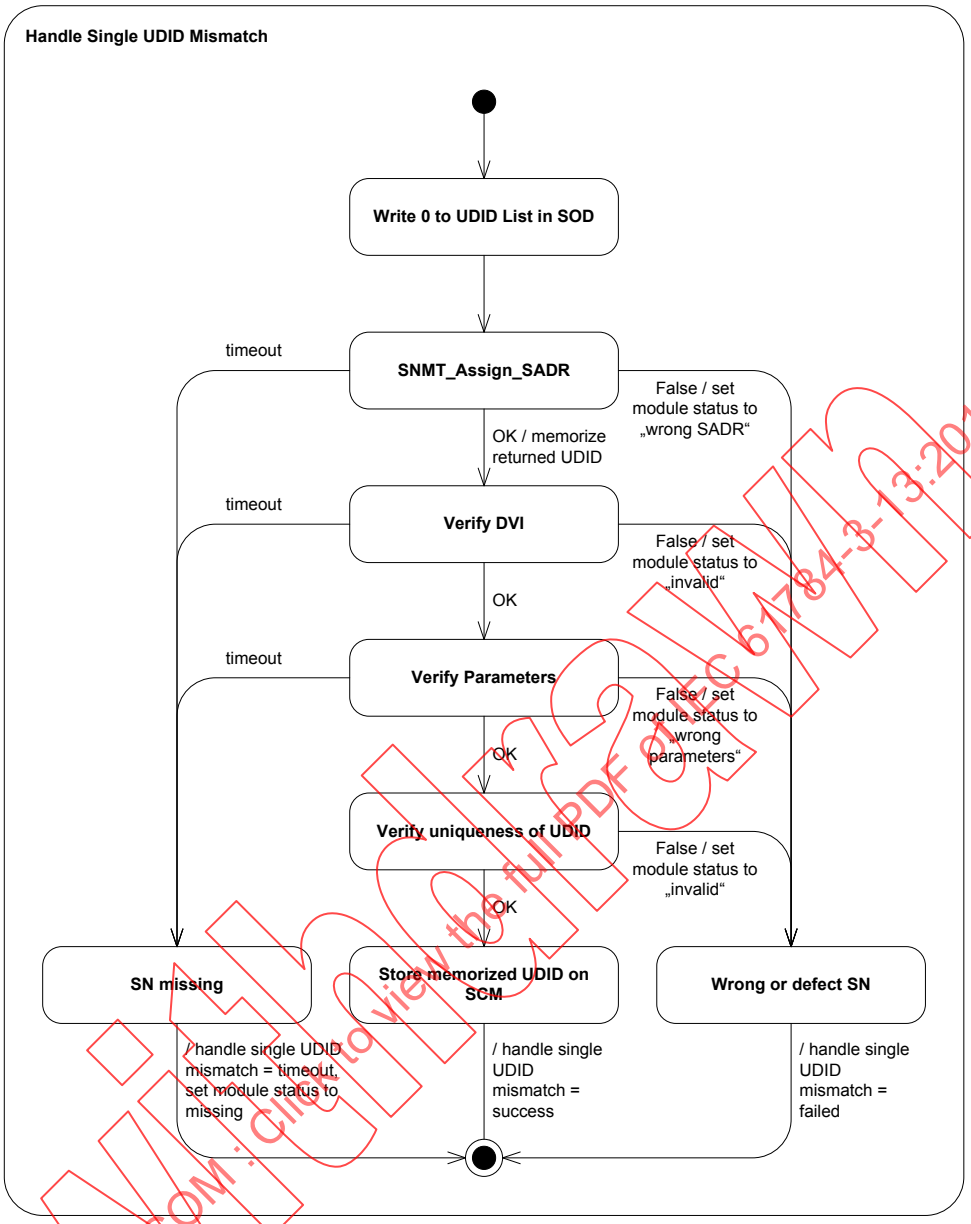


Figure 72 – State diagram SCM handle single UDID mismatch

Table 207 specifies the SCM handle single UDID mismatch states.

Table 207 – SCM handle single UDID mismatch state description

State	Description
Write 0 to UDID list in SOD	The SCM deletes the “old” stored UDID in the SOD (Object 0xCC01 -- 0xCFFF). 0 is no valid UDID
SNMT_Assign_SADR	The SCM writes the configured SADR to the module
Verify DVI	The DVI has to be checked to ensure that the new node meets the technical requirements
Verify Parameters	The parameters which are stored on the node are cross checked with the parameters stored on the SCM. The node will get new parameters if necessary/possible
Verify uniqueness of the UDID	The responded UDID is checked against the UDID list within the SOD (Object 0xC001-0xCFFF)

State	Description
Store memorized UDID on SCM	The new UDID is stored on the SCM to be able to recognize the node at the next start-up
SN missing	The SN did not answer one or more telegrams from the SCM. Therefore the SN will be handled like a missing node
Wrong or defect SN	The node does not meet the technical requirements specified in the SCM Possible reasons are: the DVI does not meet the requirements, it is not possible to change the SADR of the device (this can happen if the SADR is to be set with hardware switches), the parameters stored on the node are not equal to the parameters stored on the SCM but a download is not possible (this can happen if the SCM does not hold the parameters for a node and only knows the required checksum), the UDID is not unique within the SDN (this normally should never happen because the UDID is designed to be globally unique).

7.7.12.2 Verify parameters

The parameters (SOD) stored on the SN shall be compared with the parameters stored on the SCM.

It is not necessary to store all parameters for all nodes on the SCM. Alternatively the required parameter checksum shall be stored on the SCM. In this case, the SCM shall verify the correctness of the parameters on the node. If the parameters are not correct (the checksum does not match), the SCM will not be able to re-configure such nodes.

The “SCM verify parameters” state diagram is specified by Figure 73.

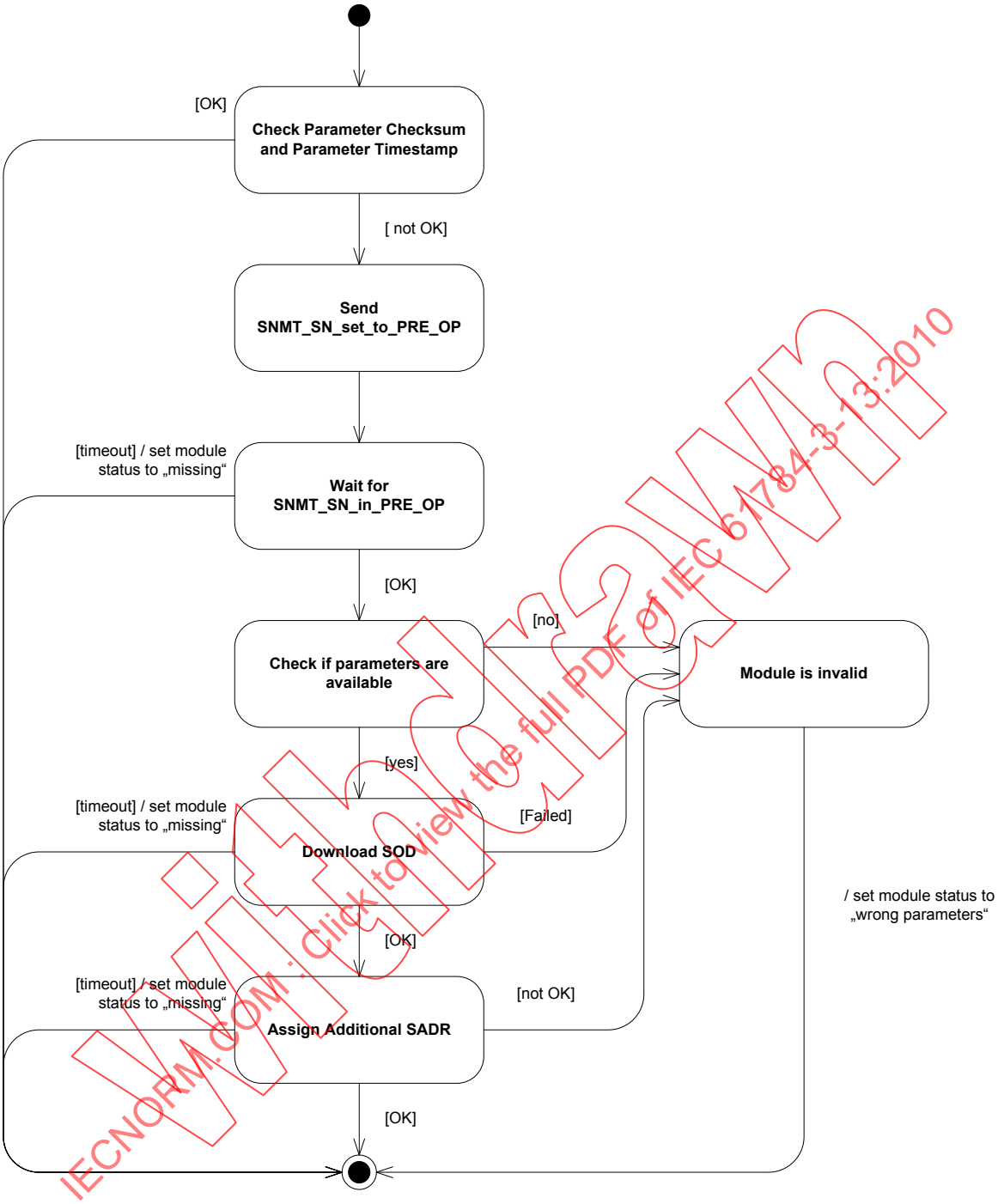


Figure 73 – State diagram SCM verify parameters

Table 208 specifies the SCM verify parameters states.

Table 208 – SCM verify parameters state description

State	Description
Check Parameter Checksum and Parameter Timestamp	The SCM checks if the current parameters for the node are the same which are stored on the SCM
Send SNMT_SN_set_to_PRE_OP	Resets the SN
Wait for SNMT_SN_in_PRE_OP	Waits for the reply of the reset service
Check if parameters are available	If the checksum on the node is different from the checksum on the SCM, the SCM checks if parameters for re-configuration of the node are available
Download SOD	The SCM reconfigures the node
Module is invalid	If the parameters on the node are not equal to the parameters stored on the SCM and reconfiguration of the module is not possible, the module is invalid and may not be set to Operational
Assign additional SADRs	The SCM assigns the SADR for the optional TxSPDOs 2 -- 1 023 of the corresponding SN

7.7.13 Activate SN

The SCM shall try to set the SN to Operational. The SN ether switches to Operational or an error occurred during switching to Operational. An error shall be reported with the SNMT_SN_FAIL telegram. The SCM receives this telegram and shall report it to the application. The vendor specific application shall decide how to cope with the situation. It shall send an acknowledgement telegram SNMT_SN_ACK if the error can get acknowledged, or shall keep the SN in Pre-Operational if the problem cannot be solved.

The “activate SN” state diagram is specified by Figure 74.

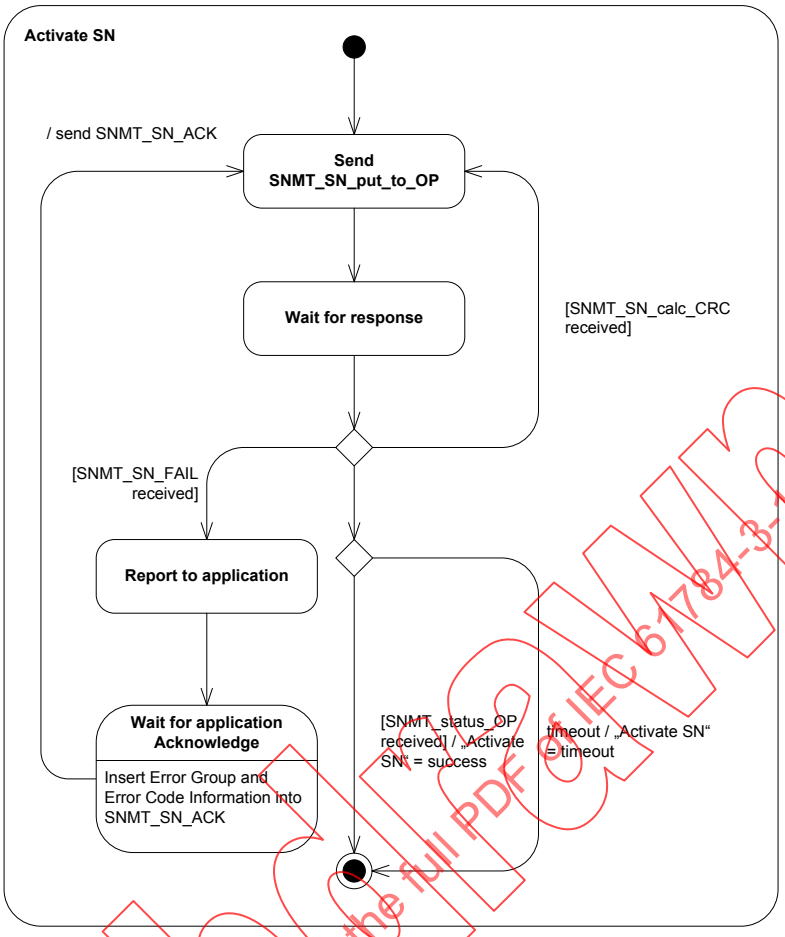


Figure 74 – State diagram activate SN

Table 209 specifies the activate SN states.

Table 209 – Activate SN state description

State	Description
Send SNMT_SN_put_to_OP	The SCM sends an OP request to the CN
Wait for response	The SCM waits for the response from the SN
Report to application	In case of an error the error is reported to the application
Wait for application Acknowledge	The application may set the error group and the error code information for the acknowledgement of the error

7.7.14 Device exchange

Exchanged devices shall send the SNMT_SN_in_PRE_OP message to the SCM. The SCM shall handle those exchanged nodes as UDID-mismatch nodes. After operator acknowledge, they shall be programmed and set to Operational.

8 Safety communication layer management

8.1 General

This clause specifies the parameterization interface of SOD.

The parameterization of FSCP 13/1 devices shall be done using a software tool which was developed and assessed according to IEC 61508. A configuration tool always shall be part of a complete safety related automation system. In general, there are two ways to access the SOD of an FSCP 13/1 device:

— using FSCP 13/1 services, or

— using a vendor specific interface for direct access (e.g. serial line).

The scope of this part is limited to using FSCP 13/1 services.

8.2 Goals of parameterization

The goals of FSCP 13/1 parameterization are:

— configuration of the parameters of FSCP 13/1 devices on a configuration device, and

— safety parameter download to FSCP 13/1 devices using FSCP 13/1 services and protocols.

8.3 Initial configuration of a device

8.3.1 General

A new FSCP 13/1 device (SN) shall have all parameters set to factory default. Therefore the SADR is set to zero, which is an invalid value.

To be able to access the SOD of the SN using FSCP 13/1 services, an SADR shall be assigned. To be able to do that, the safety parameterization software tool (SPST) shall act like an SCM within the FSCP 13/1 network. The only difference is that the SPST does not have an SADR.

Using the SNMT services described in 7.4, the tool is able to set the SADR of the SN. Then the SOD entries can be accessed/modified using the SOD Access Request service described in 7.3.

After the SN is successfully configured, it needs to be switched to Operational to permanently store the parameters. The permanent storage of parameters shall be optional and therefore might not be available on all devices.

Initial configuration can be done in one of two ways:

— by only configuring the SCM (see 8.3.2), or

— configuring each SN (see 8.3.3).

It is also possible that the configuration of an FSCP 13/1 network is a mix of these configuration methods.

8.3.2 SD setup by only configuring the SCM

It is possible to configure an entire FSCP 13/1 network by just configuring the SCM. An SCM can be configured by accessing and modifying its parameters within the SOD of the SN which serves as an SCM.

Optionally, the SCM shall hold a parameter set for each SN within the SD. The parameter set contains:

— parameters,

— length of the parameters in octets, and

— checksum and timestamp for the parameters.

After the download is finished and all parameters are correct within the SOD of the SN which hosts the SCM, the SCM can be started using the SNMT service described in 7.3. Then the SCM shall start configuring all SNs for which parameters are available.

After the SCM has switched to Operational, the software tool shall not act like an SCM anymore. It is not permitted to have more than one SCM within one SD. If it is necessary to do further configurations, the SCM shall be stopped first.

8.3.3 SD setup configuring each SN

If the SCM does not support parameter storage for the SNs, the SNs shall be configured separately. Therefore, in this case it shall be mandatory that the SNs support permanent data storage. After an SN is configured, the total parameter checksum as well as the parameter timestamp shall be read from the SN (index 1018h, see 7.5.4.7). These two parameters shall be stored on the SCM, otherwise the SCM will not be able to start-up the SN.

NOTE The disadvantage of such configurations is that if the SN is changed, the SCM is not able to re-configure the SN. Therefore the SN remains in Pre-Operational state and human intervention is required.

8.4 Avoiding of parameterize the wrong device

The safety parameterization software tool (SPST) is not only responsible for downloading the parameters, but also for the correctness of the data on the SN. Therefore the SPST shall verify that parameters were downloaded correctly. If the tool does not have a mechanism to prevent data corruption on the PC, every single parameter shall be uploaded and verified. If the tool does have such mechanisms, the checksums (sub index 254) which are supported by the module for this verification may be used. In this way, an entire index can be checked at once.

8.5 Parameter check mechanism

The integrity of parameters within the SOD shall be protected by following mechanisms:

— CRC-8 for each index (to be found in sub index 254),

— CRC-16 over all sub index 254 (to be found within the DVI object), and

— timestamp of parameter (to be found within the DVI object).

9 System requirements

9.1 Indicators and switches

There are no specific indicators and switches requirements for FSCP 13/1.

9.2 Installation guidelines

Relevant installation guidelines are specified by IEC 61918. Additionally 6.1.3.3.2 and 6.1.3.3.3 apply.

9.3 Safety function response time

The safety function response time is the worst case elapsed time following an actuation of a safety sensor (for example switch, pressure transmitter, light curtain) connected to a fieldbus, before the corresponding safe state of its safety actuator(s) (for example relay, valve, drive) is achieved in the presence of errors or failures in executing the safety function.

The demand (actuation) on a safety function is caused either by an analog signal crossing a threshold or by a digital signal changing state.

Figure 75 shows an example of typical components making up a safety function response time.

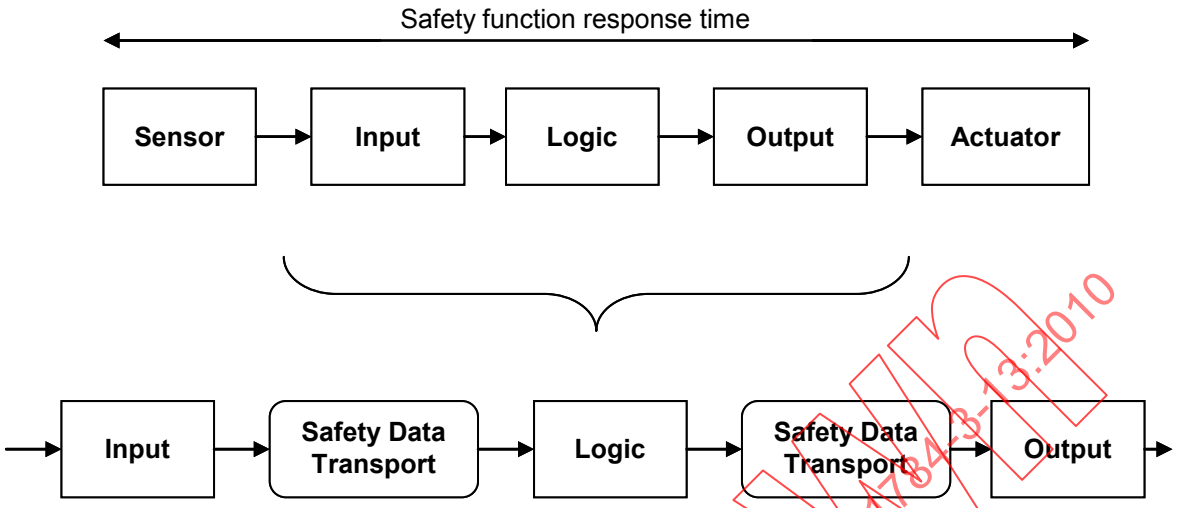


Figure 75 – Safety function response time

As shown in Figure 75, the safety function response time is calculated using Equation (6):

$$t_{SFRT} = t_{Sensor} + t_{Input} + 2 \times t_{NRT} + t_{Controller} + t_{Output} + t_{Actuator} \quad (6)$$

where

t_{SFRT} is the safety function response time;

t_{NRT} is the network reaction time, refer to Equation (7);

t_{Sensor} is the sensor reaction time;

t_{Input} is the input reaction time;

$t_{Controller}$ is the controller reaction time;

t_{Output} is the output reaction time;

$t_{Actuator}$ is the actuator reaction time.

All but the network reaction time are outside the scope of this part.

The network reaction time is calculated by Equation (7).

$$t_{NRT} = t_{SCT} + t_{maxSPD} + 2 \times t_{TRC} \times c_{clockTol} \quad (7)$$

where

t_{NRT} is the network reaction time;

t_{SCT} is the safety control time (see SCT_U32, Table 117);

t_{maxSPD} is the maximum SPDO propagation delay (see MaxSPDOPropagationDelay_U16, Table 124);

t_{TRC} is the time request cycle (see TimeRequestCycle_U32, Table 126);

$c_{clockTol}$ is the tolerance of the hardware clocks.

9.4 Duration of demands

The demand has to exist at least for the duration of the network reaction time (see 9.3) to guarantee that the functional safety communication system detects the demand.

9.5 Constraints for calculation of system characteristics

9.5.1 General

Several constraints shall be used for calculating safety related characteristics of systems using FSCP 13/1.

9.5.2 Number of sinks limit

FSCP 13/1 implementations are limited to a maximum of 1 023 producing and 1 023 consuming nodes. The number of producing nodes and the network cycle time have to be considered to not exceed the message rate limit (see 9.5.3).

The residual error rate values given in A.3 and A.4 are valid for a single point to point relationship. To calculate the residual error rate for a specific safety function the formula (1) in IEC 61784-3:2010 shall be applied.

9.5.3 Message rate limit

The message rate shall be limited to 10 000 safety messages per second.

9.5.4 Message payload limit

The message payload of one safety PDU shall not exceed 254 octets. For additional restrictions see A.3 and A.4.

9.5.5 Residual error rate

FSCP 13/1 fulfils the requirements of IEC 61508 / SIL 3 for reliability and safety. For SIL 3, an overall system residual error rate $< 10^{-7}$ shall be achieved. The safety related network shall not use up more than 1 % of the system residual error rate, therefore the network residual error rate shall be below 10^{-9}.

Implementations according to this part provide a network residual error rate below 10^{-9}, see A.3 and A.4 for prove.

9.6 Maintenance

9.6.1 Diagnostic information

Diagnostic information according to 7.5.4 shall be stored in the object 1001h, 1002h, 1003h by the SN. This information can be read by non safety related applications.

9.6.2 Replacement of safety related devices

9.6.2.1 SN replacement

After replacing the affected SNs, the maintenance technician has to decide.

— If only one SN is replaced, is the SCM able to handle the replacement automatically? Only a simple confirmation by the maintenance technician is requested.

— If more than one SN is replaced, the configuration (see 6.2.3.3) shall be repeated and verification of the safety functionality of the replaced SNs shall be done (see 6.2.3.4).

9.6.2.2 Replacement of SN running the SCM

If the configuration and the application with the corresponding parameters are available after the replacement, the SCM replacement has no influence on the safety functions.

If any data is lost, the steps according to 6.2.3 shall be repeated. In accordance with 6.2.3.4, a complete verification shall be done (as it is required during commissioning).

9.6.3 Modification

After modification, the steps according to 6.2.3 shall be repeated. In accordance with 6.2.3.4, a complete verification shall be done (as it is required during commissioning).

9.6.4 Machine part changing

All changeable machine parts shall be verified against the safety specification for the application at least one time according to the procedure listed in 6.2.3.4. After verification, the SCM knows the configuration for all changeable machine parts; this means that after changing machine parts, no further steps are required.

Open input situations due to missing producers after machine part changes shall be handled by the application.

9.6.5 Firmware update of safety related nodes

A firmware update for a safety related node is a vendor specific task. There is no service within FSCP 13/1 to support this task.

9.6.6 Machine check due to service interval

FSCP 13/1 requires no special check.

9.7 Safety manual

Vendors of devices which implement a safety communication layer compliant to this part shall supply a safety manual which follows the requirements in IEC 61508 which shall at least inform the user:

a) of constraints for calculation of system characteristics, see 9.5,

b) of their responsibilities in the proper parameterization of the device, and

c) about preconditions to be fulfilled for programming and parameterization modes, see 6.2.2.2.

10 Assessment

10.1 General

Assessment of conformity of an FSCP 13/1 implementation or device by a competent assessment body is highly recommended. Each node which is within an interoperable FSCP 13/1 communication network shall provide compatibility according to this part. Only the compatibility of devices guarantees the availability and safety of FSCP 13/1 networks.

Assessment flow of FSCP 13/1 devices is specified by Figure 76.

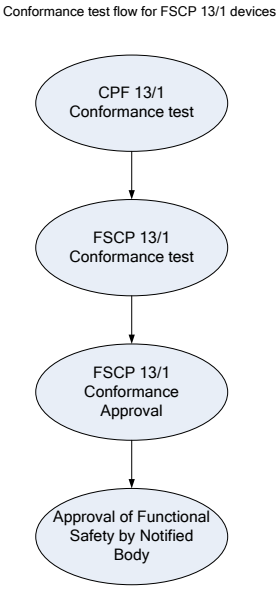


Figure 76 – Assessment flow of devices

10.2 CP 13/1 assessment

For FSCP 13/1 assessment, the underlying communication layer CP 13/1 for all non safety communications is required. For successful interactions between different safety nodes, all mandatory CP 13/1 services shall be implemented according to IEC 61784-3 CP 13/1 and IEC 61158-5-13.

10.3 FSCP 13/1 conformance test

The FSCP 13/1 conformance test ensures that an FSCP 13/1 device conforms to all safety services and measurements according to this part. The test should be done by a competent assessment body or by an assessed test tool. The result is a notice of assessment and a detailed test report which is needed for obtaining the functional safety assessment from a competent assessment body.

10.4 Approval of functional safety by competent assessment body

If machinery needs official approval (e.g. for IEC 61508 or other safety related standards), all FSCP 13/1 devices being used need this approval too. According to the standards, functional safety should be assessed by a competent assessment body.

10.5 Summary

After all three assessments, the devices may be marked with the official FSCP 13/1 Logo.

Annex A
(informative)

Additional information for functional safety communication profiles of CPF 13

A.1 Hash function calculation

The following code example shows a possible way to calculate the CRC-8 and CRC-16 used by FSCP 13/1.

```
/*****************************************************************************
POWERLINK Safety_CRC.c - Reference implementation for POWERLINK Safety CRC-
Calculations
This programm should be a reference implementation for the two POWERLINK Safety
CRC-Calculations
It is possible to compare the implemented CRC's with the mentioned reference values
from this
programm.
This programm is not optimized for performance.

The programm depends on an 8 Bit/Byte machine.
unsigned int has to have 16 bits or more.

Summary:
1. Append zero bits to the data stream
2. Devide the complete data stream by using CRC arithmetic
3. The reminder is the checksum

POWERLINKsafety V1.0
CRC Referenz calculation
Review: Bernecker + Rainer Industrie-Elektronik Ges.m.b.H.
*****************************************************************************/

/* the polynomial is specified without its top bit, the top bit is contained
implicitly in the algorithm because 1 xor 1 is always 0 (register optimized) */
#define c_poly1 0x2F            /* x8+x5+x3+x2+x+1 = 0x12F */
#define c_poly2 0xD5            /* x8+x7+x6+x4+x2+1 = 0x1D5 */
#define c_poly3 0x5935          /* x16+x14+x12+x11+x8+x5+x4+x2+1  0x15935*/
#define c_poly4 0x755B          /* x16+x14+x13+x12++x10+x8+x6+x4+x3+x+1 0x1755B */

/* prototype of left shifting by one bit */
unsigned char
   uc_array_shift_left(unsigned char *, unsigned int);
/* prototype CRC8 */
unsigned char
   uc_POWERLINK Safety_CRC8(unsigned char, unsigned char, unsigned char *);
/* prototype CRC16 */
unsigned int
   ui_POWERLINK Safety_CRC16(unsigned int, unsigned int, unsigned char *);
/* test program */
/* the length of sub frame depands from length of payload */
#define payload 8                   /* length of payload in reference sub frame */
#define c_sub_length1 (payload + 4) /* length of reference sub frame 1 w/o CRC  */
#define c_sub_length2 (payload + 5) /* length of reference sub frame 2 w/o CRC  */

/*first part of POWERLINK Safety short telegram    */
unsigned char uc_subframe_1[c_sub_length1 + 2];

/*second part of POWERLINK Safety short telegram   */
unsigned char uc_subframe_2[c_sub_length2 + 2];

int main(void)
```

```
{
unsigned int result;
/* reference data stream for first sub frame */
                                /*MSB=bit 7 --- LSB bit 0                */
  uc_subframe_1[ 0] = 0x23;     /*  1. byte address 0-7                   */
  uc_subframe_1[ 1] = 0xC8;     /*  2. byte address 8-9, ID               */
  uc_subframe_1[ 2] = 0x08;     /*  3. byte LE                            */
  uc_subframe_1[ 3] = 0x34;     /*  4. byte CT 0-7                        */
  uc_subframe_1[ 4] = 0x11;     /*  5. byte DB 1                          */
  uc_subframe_1[ 5] = 0x22;     /*  6. byte DB 2                          */
  uc_subframe_1[ 6] = 0x33;     /*  7. byte DB 3                          */
  uc_subframe_1[ 7] = 0x44;     /*  8. byte DB 4                          */
  uc_subframe_1[ 8] = 0x55;     /*  9. byte DB 5                          */
  uc_subframe_1[ 9] = 0x66;     /* 10. byte DB 6                          */
  uc_subframe_1[10] = 0x77;     /* 11. byte DB 7                          */
  uc_subframe_1[11] = 0x88;     /* 12. byte DB 8                          */

/* reference data stream for second sub frame */
                                /*MSB=Bit 7 --- LSB BIT 0                */
  uc_subframe_2[ 0] = 0x22;     /*  1. byte address xor domain 0-7        */
  uc_subframe_2[ 1] = 0xC8;     /*  2. byte address xor domain 8-9, ID    */
  uc_subframe_2[ 2] = 0x12;     /*  3. byte CT 8-15                       */
  uc_subframe_2[ 3] = 0x56;     /*  4. byte TADR 0-7                      */
  uc_subframe_2[ 4] = 0x30;     /*  5. byte TADR 8-9, TR                  */
  uc_subframe_2[ 5] = 0x11;     /*  6. byte DB 1                          */
  uc_subframe_2[ 6] = 0x22;     /*  7. byte DB 2                          */
  uc_subframe_2[ 7] = 0x33;     /*  8. byte DB 3                          */
  uc_subframe_2[ 8] = 0x44;     /*  9. byte DB 4                          */
  uc_subframe_2[ 9] = 0x55;     /* 10. byte DB 5                          */
  uc_subframe_2[10] = 0x66;     /* 11. byte DB 6                          */
  uc_subframe_2[11] = 0x77;     /* 12. byte DB 7                          */
  uc_subframe_2[12] = 0x88;     /* 13. byte DB 8                          */

/* calculation of CRCs with reference data stream */
result = uc_POWERLINK Safety_CRC8(c_sub_length1, c_poly1, &uc_subframe_1[0]);
/* result = 0x3c */
uc_subframe_1[c_sub_length1]   = result;          /* 13. byte CRC-8 bit 0-7    */
result = uc_POWERLINK Safety_CRC8(c_sub_length2, c_poly2, &uc_subframe_2[0]);
/* result = 0x4f */
uc_subframe_2[c_sub_length2]   = result;          /* 14. byte CRC-8 bit 0-7    */
result =  ui_POWERLINK Safety_CRC16(c_sub_length1, c_poly3, &uc_subframe_1[0]);
/* result = 0x0374 */
  uc_subframe_1[c_sub_length1]   = result >> 8;    /* 13. byte CRC-16 bit 15-8  */
  uc_subframe_1[c_sub_length1+1] = result;         /* 14. byte CRC-16 bit 0-7   */

result =  ui_POWERLINK Safety_CRC16(c_sub_length2, c_poly4, &uc_subframe_2[0]);
/* result = 07031 */
  uc_subframe_2[c_sub_length2]   = result >> 8;    /* 14. byte CRC-16 bit 15-8  */
  uc_subframe_2[c_sub_length2+1] = result;         /* 15. byte CRC-16 bit 0-7   */

return 0;
}


/*****************************************************************************
function uc_POWERLINK Safety_CRC8()            version 1.0
date 2005-04-22
maxmuim datalength of 14 byte
return: unsinged char checksum
parameter:  1 byte frame length
1 byte polynominal
1 pointer start adress of frame array
*****************************************************************************/
unsigned char uc_POWERLINK Safety_CRC8(unsigned char uc_sub_length,
unsigned char uc_poly,
unsigned char *uc_subframe)
{
  unsigned char uc_checksum;                    /* reminder as crc checksum     */
  unsigned char uc_i;                           /* counter variable             */
  unsigned int ui_stream_le;                    /* stream length                */
```

```
  unsigned char uc_data[15];                    /* include the complete stream
                                                    with appended zeros         */
                                                /*array is not optimized!       */

for (uc_i = 0; uc_i < uc_sub_length; uc_i++)
  {
  uc_data[uc_i]=uc_subframe[uc_i];          /*copy the frame in stream and */
  }
uc_data[uc_sub_length] = 0;                 /*append the zeros*/
ui_stream_le = uc_sub_length + 1;
/* CRC calculation by shifting and xor */
for (uc_i=0; uc_i < uc_sub_length * 8; uc_i++)
  {
  if (uc_array_shift_left( &uc_data[0], ui_stream_le))
      {
      /* XOR when bit falling out is not zero */
      uc_data[0] = uc_data[0] ^ uc_poly;
      }
  }

/* result*/
uc_checksum = uc_data[0];
return (uc_checksum);
}

/*****************************************************************************
function uc_POWERLINK Safety_CRC16()          version 1.0
date 2005-04-22
maxmuim datalength of 261 byte
return: unsinged int checksum
parameter:  1 integer frame length
1 integer polynominal
1 pointer start adress of frame array
*****************************************************************************/

unsigned int ui_POWERLINK Safety_CRC16(unsigned int ui_sub_length,
unsigned int ui_poly,
unsigned char *uc_subframe)
{
unsigned int ui_crc_sum=0;
unsigned int ui_i;
unsigned int ui_stream_le;                  /* stream length               */
unsigned char uc_data[263];
unsigned char uc_poly[2];


uc_poly[1]= ui_poly;        /*LSB*/
uc_poly[0]= ui_poly >>= 8;  /*MSB*/


for (ui_i = 0; ui_i < ui_sub_length; ui_i++)
  {
  uc_data[ui_i]=uc_subframe[ui_i];          /* copy the frame in stream and */
  }
uc_data[ui_sub_length] = 0;                 /* append the zeros */
uc_data[ui_sub_length+1] = 0;               /* append the zeros */
ui_stream_le = ui_sub_length + 2;
/* CRC calculation by shifting and xor */
for (ui_i=0; ui_i < ui_sub_length * 8; ui_i++)
  {
  if (uc_array_shift_left( &uc_data[0], ui_stream_le))
    {
    /* XOR when bit falling out is not zero */
    uc_data[0] = uc_data[0] ^ uc_poly[0];
    uc_data[1] = uc_data[1] ^ uc_poly[1];
    }
  }

/* result */
ui_crc_sum = uc_data[0];
```

```
ui_crc_sum <<= 8;
ui_crc_sum = ui_crc_sum | uc_data[1];

return(ui_crc_sum);
}


/*****************************************************************************
function uc_array_shift_left()          version 1.0
date 2005-04-22
maxmuim datalength of UINT_MAX
return: unsigned char value of bit shifted out at left
parameter: 1 pointer start adress of stream array
1 integer field length
*****************************************************************************/

unsigned char uc_array_shift_left(unsigned char *uc_info, unsigned int us_fields)
{
signed int si_i=0;              /* count variable */
unsigned char msb=0;            /* buffer */
unsigned char carry=0;          /* carry  */

/* shift entire array from right to left */
for (si_i = us_fields - 1; si_i >= 0; si_i--)
  {
  msb = uc_info[si_i] & 0x80;
  msb >>= 7;

  uc_info[si_i] <<= 1;                  /* shift left */
  uc_info[si_i] |= carry;               /* add carry */
  carry = msb;                          /* msb to next carry */
  }

return (carry);
}
```

A.2 Stochastic errors – general considerations

A.2.1 General

There are several measures that enable the receiving device to identify a stochastic error. To verify the data correctness, the following principles shall be used:

- CRC-calculation,
- comparison of data-information of both PDU parts,
- checking of time stamp, and
- checking of received address (comparison with internal address or address in look-up-table).

The checking of time stamp and received address is not in the scope of this clause.

A.2.2 Error detection mechanisms

The safety PDU consists of two subparts. Each SPDU subpart contains a CRC, which will detect all possible bit errors from zero up to CRCs Hamming distance. Data of SPDU part one, which is defined as payload data and header information, is repeated in the SPDU part two. CRC1 and CRC2 use the same polynomial but their value is not identical generally, because the residual is calculated with a different number of octets from both PDU parts.

Figure A.1 shows the structure of the safety PDU.

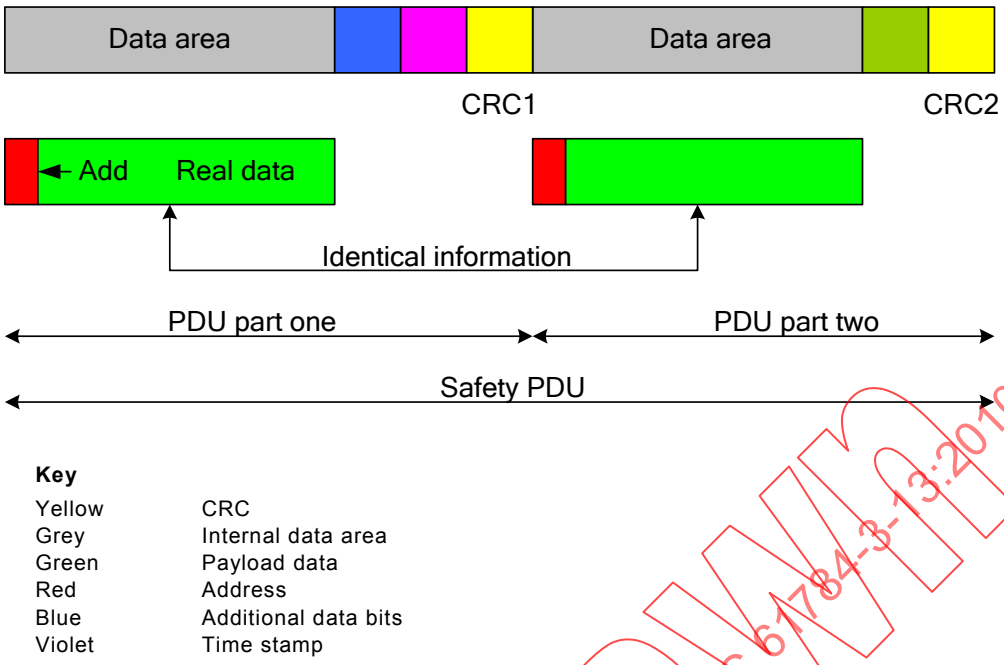


Figure A.1 – Structure of safety PDU

The error detection for each PDU part is done by a CRC. Both CRCs are calculated from all bits of PDU part except the CRC itself. Figure A.2 shows the structure of a PDU part with the CRC.

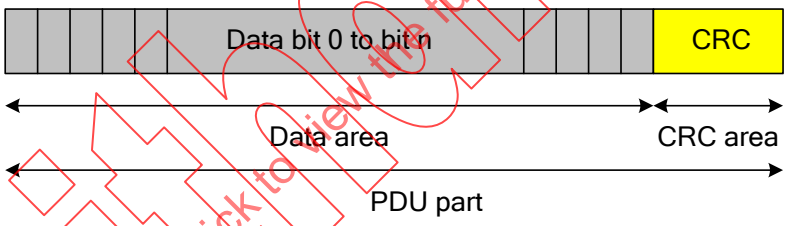


Figure A.2 – Error detection by the use of a CRC

Safety PDUs which transfer 0 to 8 octets use CRC-8. All larger Safety PDUs (for transmission of 9 and more octets) use CRC-16. The properties of selected CRCs are well known (see [48]):

CRC-8: $X^8+X^5+X^3+X^2+X+1$ (97h, omitting X^0, see [48]). The CRC guarantees a Hamming distance of 4 for up to 119 data bits.

CRC-16: $X^{16}+X^{14}+X^{12}+X^{11}+X^8+X^5+X^4+X^2+1$ (AC9Ah, omitting X^0, see [48]). The CRC-16 guarantees a Hamming distance of 5 for up to 241 data bits. For all PDUs from 242 to 2 048 data bits, a Hamming distance of 4 is guaranteed.

If r is the largest exponent number of the CRC polynomial (r is 8 for CRC-8 and 16 for CRC-16), then the selected CRCs have the following additional properties:

— all error effecting an odd number and less than r bits are detected,

— all errors located within an area of r bits are detected,

— the probability of error detection which contains more errors than the guaranteed Hamming distance will be less than 2^{-r} (for CRC-8: 1/256, for CRC-16: 1/65 536).

A.2.3 Calculations

For all stochastic calculations the formula (1) and (D.1) taken from IEC 61784-3:2010 are used:

a) FSCP 13/1 uses redundancy with cross checking (twofold message) therefore formula (D.1) of IEC 61784-3 is used to calculate the residual error probability for cross checking a safety message $R_{BC}(p_e)$, where p_e is the bit error probability. Formula (D.1) of IEC 61784-3 takes into account the probability of error detection by a CRC data integrity assurance (2^{-r} factor).

b) Formula (1) of IEC 61784-3 is used to calculate Λ_{SL} the residual error rate or probability of failure per hour (PFH). To reach SIL 3 according to IEC 61508, this value shall be below 10^{-9}. The symbol R_{SL} of formula (1) is replaced by R_{EMV} of the previous step.

Combining these formulas gives Equation (A.1), which is directly derived from the above formulas and shown here only for convenience. It shall be considered informative.

$$\Lambda_{SL} = 3600 \times v \times 2^{-r} \sum_{k=d}^{8\times(n+5)} \binom{8\times(n+5)}{k} \times \left(p_e^{k}(1-p_e)^{(8\times(n+5)-k)}\right)^2 \quad \text{(A.1)}$$

where

Λ_{SL} Residual error rate or probability of failure per hour (PFH);

v messages per second;

d Hamming distance of CRC used;

n length of payload data in octets (excluding header data);

p_e bit error probability.

The m of formula (1) taken from IEC 61784-3:2010 is 1 for all calculations.

NOTE The hamming distance, d, depends on the CRC used and the length of the safety message.

A.3 Stochastic errors (case A)

A.3.1 General

This subclause shows the residual error rate (Λ_{SL}) calculation for the black channel approach as claimed in IEC 61784-3.

A.3.2 Constraints

All calculations and conclusions in A.3 are valid only if the following constraints are fulfilled:

— the bit error probability is 10^{-2} in accordance to IEC 61784-3,

— the message rate is limited to 10 000 messages per second, and

— the permissible payload data range is 9–25 octets.

A.3.3 Residual error rate

Using Equation (A.1) and constraints A.3.2 the worst case Λ_{SL} is 3,117 22×10^{-10}, from this follows that the maximum number of allowed sinks is 3. Figure A.3 shows the residual error rate as a function of payload octets (vertical axis is in logarithmic scaling, period is decimal sign).

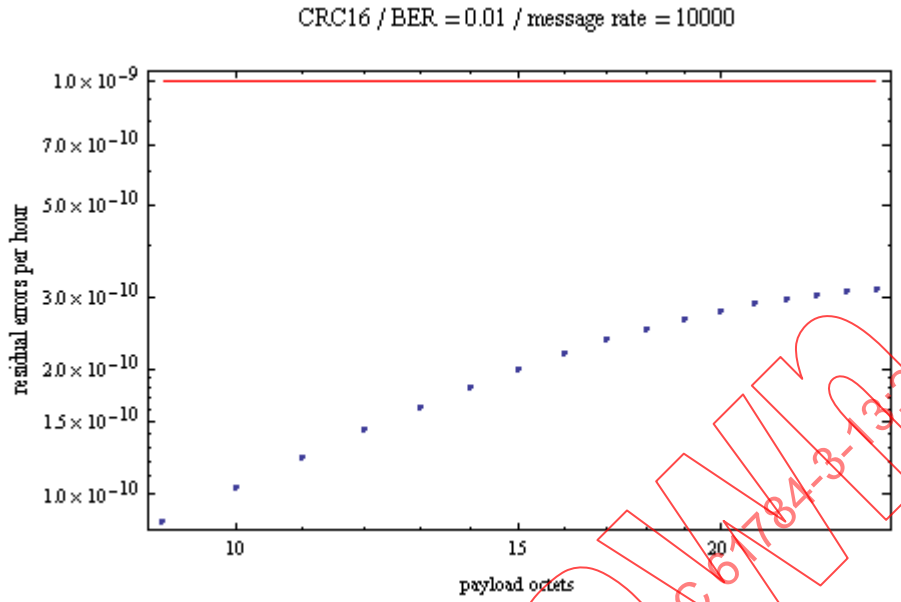


Figure A.3 – Residual errors per hour

A.3.4 Summary

The calculation, considering the constraints A.3.2, proves that the requirement for SIL 3 is fulfilled.

A.4 Stochastic errors (case B)

A.4.1 General

This subclause shows the residual error rate (Λ_{SL}) calculation when CPF 13 is used as the black channel.

A.4.2 Constraints

All calculations and conclusions in A.4 are valid only if following constraints are fulfilled:

— CPF 13 is used exclusively to transport the safety messages,
— the bit error probability is 10^{-3} (see A.4.3 for justification),
— the SPDUs shall not be fragmented for transportation,
— the message rate is limited to 10 000 messages per second, and
— the permissible payload data range is 1–254 octets.

A.4.3 Bit error probability considerations

IEC 61784-3 demands a bit error probability of 10^{-2} to be used in probability calculations unless a better value can be proven.

CPF 13 is used as the black channel therefore the minimum packet size is 512 bits (64 octets). If the bit error probability is 10^{-2}, then one out of 100 bits is destroyed so every transmitted packet of 512 bits length would be corrupt. This shows that a bit error probability of 10^{-2} is not practical.

Considering a bit error probability of 10^{-3}, one out of 1 000 bits is affected so every second transmitted packet of 512 bits length would be corrupt. In this case, the node guarding mechanisms of CPF 13/Type 13 (see IEC 61158-4-13) would force the network into a non operational state where no data communication is possible for any node, therefore no FSCP 13/1 data exchange is possible and the FSCP 13/1 nodes stay in safe state.

Therefore, if a bit error probability of 10^{-3} is observed in a real installation, all safety nodes are in the safe state. Therefore a bit error probability of 10^{-3} can be used for stochastic calculations.

A.4.4 Residual error rate (payload 1—8)

Using Equation (A.1) for 1 to 8 payload octets and constraints A.4.2, the worst case Λ_{SL} is 5,293 58 × 10^{-13}, from this follows that the maximum number of allowed sinks is 1 889.

A.4.5 Residual error rate (payload 9—254)

Using Equation (A.1) for 9 to 254 payload octets and constraints A.4.2, the worst case Λ_{SL} is 6,730 24 × 10^{-12}, from this follows that the maximum number of allowed sinks is 148. Figure A.4 shows the residual error rate as a function of payload octets (vertical and horizontal axes are in logarithmic scaling, period is decimal sign).

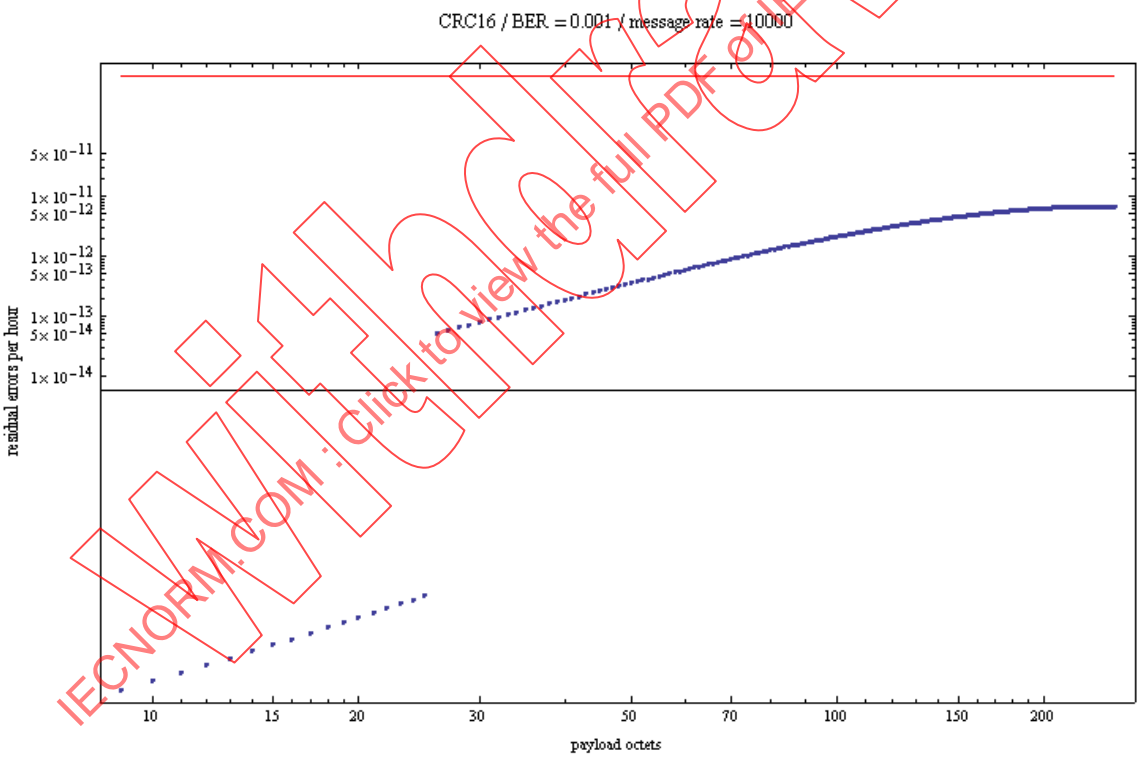


Figure A.4 – Residual errors per hour (payload 9-254)

A.4.6 Summary

The calculation, considering the constraints A.4.2, proves that the requirement for SIL 3 is fulfilled for all length of safety messages.

Annex B
(informative)

Information for assessment of the functional safety communication profiles of CPF 13

Information about test laboratories which test and validate the conformance of FSCP 13/1 products with IEC 61784-3-13 can be obtained from the National Committees of the IEC or from the following organization:

EPSG – Ethernet POWERLINK Standardization Group
Kurfürstenstraße 112
10787 Berlin
GERMANY

Phone: +49 30 85 08 85 -29
Fax: +49 30 85 08 85 -86
E-mail: info@ethernet-powerlink.org
URL: http://www.ethernet-powerlink.org/

Bibliography

[1] IEC 60050 (all parts), *International Electrotechnical Vocabulary*

NOTE See also the IEC Multilingual Dictionary – Electricity, Electronics and Telecommunications (available on CD-ROM and at <http://www.electropedia.org>)

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

[3] IEC/TS 61000-1-2, *Electromagnetic compatibility (EMC) – Part 1-2: General – Methodology for the achievement of the functional safety of electrical and electronic equipment with regard to electromagnetic phenomena*

[4] IEC 61131-2, *Programmable controllers – Part 2: Equipment requirements and tests*

[5] IEC 61131-6[11], *Programmable controllers – Part 6: Functional safety*

[6] IEC 61158 (all parts), *Industrial communication networks – Fieldbus specifications*

[7] IEC 61326-3-1, *Electrical equipment for measurement, control and laboratory use – EMC requirements – Part 3-1: Immunity requirements for safety-related systems and for equipment intended to perform safety related functions (functional safety) – General industrial applications*

[8] IEC 61326-3-2, *Electrical equipment for measurement, control and laboratory use – EMC requirements – Part 3-2: Immunity requirements for safety-related systems and for equipment intended to perform safety related functions (functional safety) – Industrial applications with specified electromagnetic environment*

[9] IEC 61496 (all parts), *Safety of machinery – Electro-sensitive protective equipment*

[10] IEC 61508-1:2010[12], *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 1: General requirements*

[11] IEC 61508-4:2010[12], *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 4: Definitions and abbreviations*

[12] IEC 61508-5:2010[12], *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 5: Examples of methods for the determination of safety integrity levels*

[13] IEC 61508-6:2010[12], *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 6: Guidelines on the application of IEC 61508-2 and IEC 61508-3*

[14] IEC 61511 (all parts), *Functional safety – Safety instrumented systems for the process industry sector*

[15] IEC 61784-1, *Industrial communication networks – Profiles – Part 1: Fieldbus profiles*

[16] IEC 61784-4[13], *Industrial communication networks – Profiles – Part 4: Secure communications for fieldbuses*

[17] IEC 61784-5 (all parts), *Industrial communication networks – Profiles – Part 5: Installation of fieldbuses – Installation profiles for CPF x*

[18] IEC 61800-5-2, *Adjustable speed electrical power drive systems – Part 5-2: Safety requirements – Functional*

[19] IEC/TR 62059-11, *Electricity metering equipment – Dependability – Part 11: General concepts*

[20] IEC 62061, *Safety of machinery – Functional safety of safety-related electrical, electronic and programmable electronic control systems*

[21] IEC/TR 62210, *Power system control and associated communications – Data and communication security*

---

11 In preparation.

12 To be published.

13 Proposed new work item under consideration.

[22] IEC 62280-1, *Railway applications – Communication, signalling and processing systems – Part 1: Safety-related communication in closed transmission systems*

[23] IEC 62280-2, *Railway applications – Communication, signalling and processing systems – Part 2: Safety-related communication in open transmission systems*

[24] IEC 62443 (all parts), *Industrial communication networks – Network and system security*

[25] ISO/IEC Guide 51:1999, *Safety aspects — Guidelines for their inclusion in standards*

[26] ISO/IEC 2382-14, *Information technology – Vocabulary – Part 14: Reliability, maintainability and availability*

[27] ISO/IEC 2382-16, *Information technology – Vocabulary – Part 16: Information theory*

[28] ISO/IEC 7498 (all parts), *Information technology – Open Systems Interconnection – Basic Reference Model*

[29] ISO 10218-1, *Robots for industrial environments – Safety requirements – Part 1: Robot*

[30] ISO 12100-1, *Safety of machinery – Basic concepts, general principles for design – Part 1: Basic terminology, methodology*

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

[32] ISO 13849-2, *Safety of machinery – Safety-related parts of control systems – Part 2: Validation*

[33] ISO 14121, *Safety of machinery – Principles of risk assessment*

[34] ANSI/ISA-84.00.01-2004 (all parts), *Functional Safety: Safety Instrumented Systems for the Process Industry Sector*

[35] VDI/VDE 2180 (all parts), *Safeguarding of industrial process plants by means of process control engineering*

[36] GS-ET-26[14], *Grundsatz für die Prüfung und Zertifizierung von Bussystemen für die Übertragung sicherheitsrelevanter Nachrichten*, May 2002. HVBG, Gustav-Heinemann-Ufer 130, D-50968 Köln ("*Principles for Test and Certification of Bus Systems for Safety relevant Communication*")

[37] ANDREW S. TANENBAUM, *Computer Networks*, 4th Edition, Prentice Hall, N.J., ISBN-10:0130661023, ISBN-13: 978-0130661029

[38] W. WESLEY PETERSON, *Error-Correcting Codes*, 2nd Edition 1981, MIT-Press, ISBN 0-262-16-039-0

[39] BRUCE P. DOUGLASS, *Doing Hard Time*, 1999, Addison-Wesley, ISBN 0-201-49837-5

[40] *New concepts for safety-related bus systems*, 3rd International Symposium "Programmable Electronic Systems in Safety Related Applications ", May 1998, from Dr. Michael Schäfer, BG-Institute for Occupational Safety and Health.

[41] DIETER CONRADS, *Datenkommunikation*, 3rd Edition 1996, Vieweg, ISBN 3-528-245891

[42] German IEC subgroup DKE AK 767.0.4: *EMC and Functional Safety*, Spring 2002

[43] NFPA79 (2002), *Electrical Standard for Industrial Machinery*

[44] GUY E. CASTAGNOLI, *On the Minimum Distance of Long Cyclic Codes and Cyclic Redundancy-Check Codes*, 1989, Dissertation No. 8979 of ETH Zurich, Switzerland

[45] GUY E. CASTAGNOLI, STEFAN BRÄUER, and MARTIN HERRMANN, *Optimization of Cyclic Redundancy-Check Codes with 24 and 32 Parity Bits*, June 1993, IEEE Transactions On Communications, Volume 41, No. 6

[46] SCHILLER F and MATTES T: *An Efficient Method to Evaluate CRC-Polynomials for Safety-Critical Industrial Communication*, Journal of Applied Computer Science, Vol. 14, No 1, pp. 57-80, Technical University Press, Łódź,Poland, 2006

[47] SCHILLER F and MATTES T: *Analysis of CRC-polynomials for Safety-critical Communication by Deterministic and Stochastic Automata*, 6th IFAC Symposium on Fault

[14] This document has been one of the starting points for this part. It is currently undergoing a major revision.

Detection, Supervision and Safety for Technical Processes, SAFEPROCESS 2006, pp. 1003-1008, Beijing, China, 2006

[48] PHILIP KOOPMAN, TRIDIB CHAKRAVARTY, *Cyclic Redundancy Code (CRC) Polynomial Selection for Embedded Networks*

[49] HEINZ GALL, THOMAS STEFFENS, KLAUS KEMP, *Anwendung der Bussysteme in der Anlagensicherheit der Chemie-Industrie*, TÜV Anlagentechnik GmbH

[50] DIETMAR LOCHMANN, *Digitale Nachrichtentechnik, Signale, Codierungen, Übertragungssysteme, Netze*, 2002, Verlag Technik, Berlin, ISBN 3-341-01321-0

[51] Ethernet POWERLINK Standardization Group, *Ethernet POWERLINK V2.0 Communication Profile Specification, Draft Standard, Version 1.0.0*

[52] Ethernet POWERLINK Standardization Group, *Ethernet POWERLINK V1.0 Profile Specification, Draft Standard Proposal, Version 1.07b*

SOMMAIRE

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

RÉSEAUX DE COMMUNICATION INDUSTRIELS – PROFILS –

Partie 3-13: Bus de terrain de sécurité fonctionnelle – Spécifications supplémentaires pour CPF 13

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI 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, la CEI – 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 la CEI"). 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 la CEI, participent également aux travaux. La CEI 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 la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI 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 la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) La CEI 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 la CEI. La CEI 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 la CEI, 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 la CEI, 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 la CEI ou de toute autre Publication de la CEI, 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.

La Norme internationale CEI 61784-3-13 a été établie par le sous-comité 65C: Réseaux de communication industriels, du comité d'études 65 de la CEI: Mesure, commande et automation dans les processus industriels.

La présente version bilingue (2012-02) correspond à la version anglaise monolingue publiée en 2010-06.

Le texte anglais de cette norme est issu des documents 65C/591A/FDIS et 65C/603/RVD.

Le rapport de vote 65C/603/RVD donne toute information sur le vote ayant abouti à l'approbation de cette norme.

La version française de cette norme n'a pas été soumise au vote.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Une liste de toutes les parties de la série CEI 61784-3, présentées sous le titre général *Réseaux de communication industriels – Profils – Bus de terrain de sécurité fonctionnelle*, est disponible sur le site web de la CEI.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous "http://webstore.iec.ch" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

IMPORTANT – Le logo "colour inside" qui se trouve sur la page de garde 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.

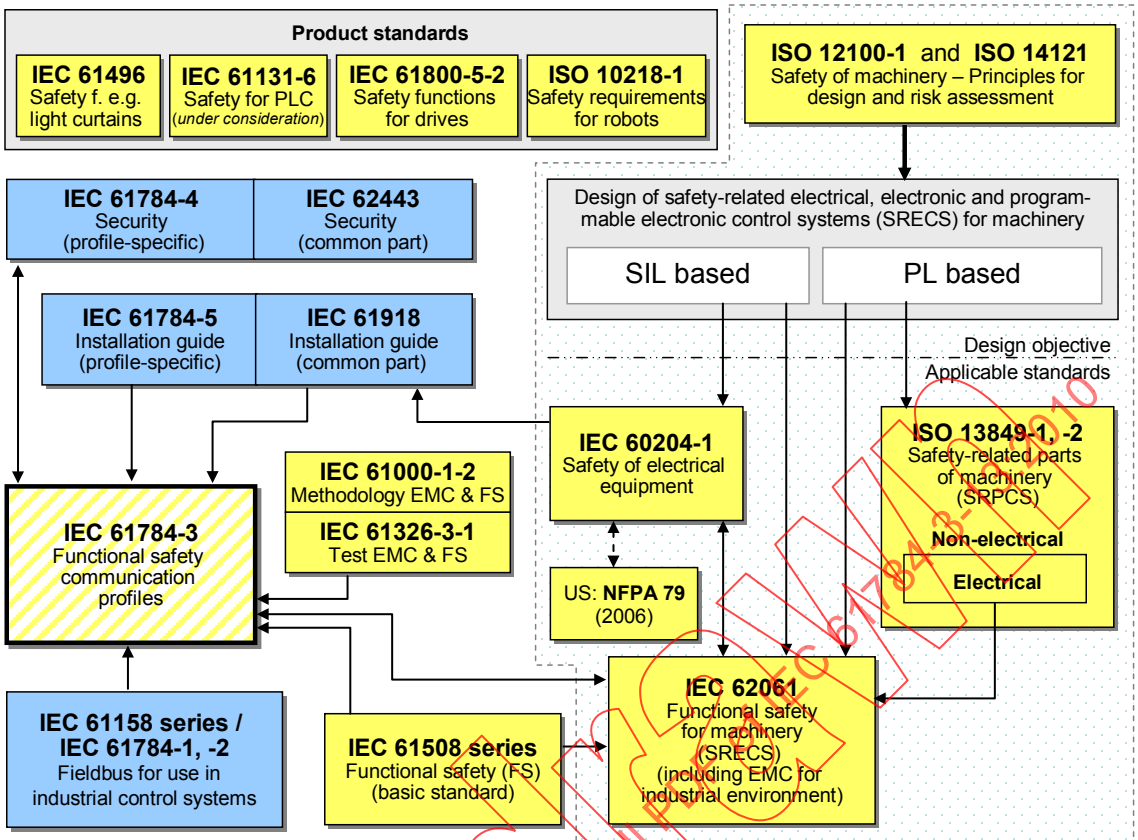
0 Introduction

0.1 Généralités

La norme CEI 61158 relative aux bus de terrain, ainsi que ses normes associées CEI 61784-1 et CEI 61784-2, définissent un ensemble de protocoles de communication qui assurent la commande répartie d'applications automatisées. La technologie de bus de terrain est désormais reconnue et bien éprouvée. Ainsi, de nombreuses améliorations des bus de terrain se développent pour traiter de domaines non encore normalisés tels que les applications en temps réel relatives à la sécurité et de sûreté.

La présente norme définit les principes pertinents applicables aux communications en termes de sécurité fonctionnelle en référence à la série CEI 61508, et spécifie plusieurs couches de communication de sécurité (profils et protocoles correspondants) basés sur les profils de communication et les couches de protocoles de la CEI 61784-1, la CEI 61784-2 et la série CEI 61158. Elle ne couvre pas les aspects relatifs à la sécurité électrique et à la sécurité intrinsèque.

La Figure 1 illustre les relations entre la présente norme et les normes pertinentes relatives à la sécurité et au bus de terrain dans un environnement machines.



Key

(yellow) safety-related standards

(blue) fieldbus-related standards

(dashed yellow) this standard

Légende

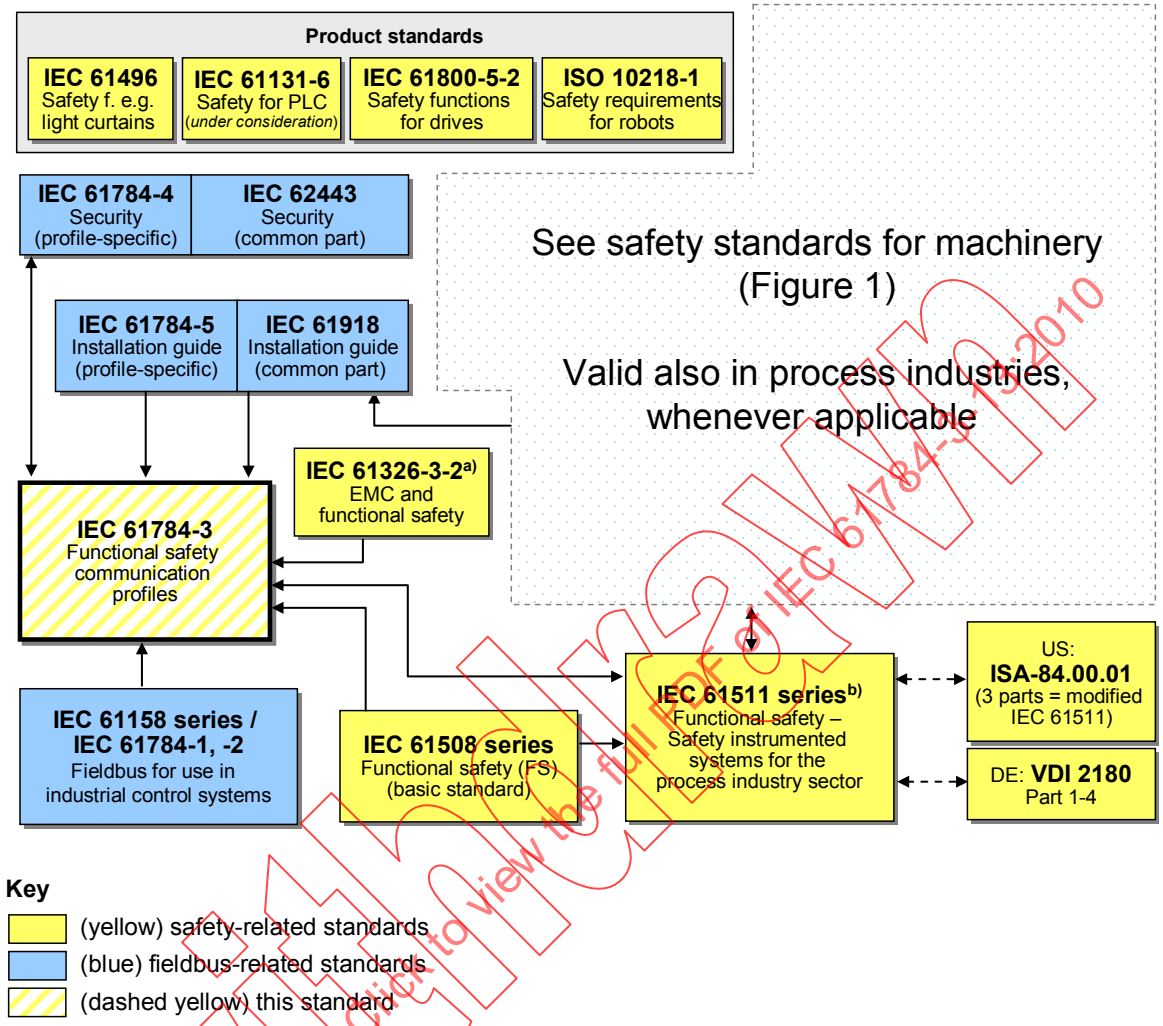
Anglais	Français
Product standards	Normes de produits
Safety function, e.g. light curtains	Fonction de sécurité, par exemple rideaux de lumière
Safety for PLC (under consideration)	Sécurité relative aux automates programmables (à l'étude)
Safety functions for drives	Fonctions de sécurité applicables aux entraînements
Safety requirements for robots	Exigences de sécurité applicables aux robots
Safety of machinery - ... assessment	Sécurité des machines – principes généraux de conception et appréciation du risque
Security (profile-specific)	Sûreté (spécifique au profil)
Security (common part)	Sûreté (partie commune)
Design of safety-related .... for machinery	Conception des systèmes de commande électriques, électroniques et électroniques programmables relatifs à la sécurité pour les machines
SIL based	Basé sur SIL
PL based	Basé sur PL
Installation guide (profile-specific)	Guide d'installation (spécifique au profil)
Installation guide (common part)	Guide d'installation (partie commune)

Anglais	Français
Design objective	Objectif de conception
Applicable standards	Normes applicables
Safety of electrical equipment	Sécurité des équipements électriques
Safety-related parts of machinery	Parties des systèmes de commande relatives à la sécurité
Non-electrical	Non électrique
Electrical	Electrique
Methodology EMC & functional safety	Méthodologie en matière de compatibilité électromagnétique & sécurité fonctionnelle
Test EMC & functional safety	Essai CEM et sécurité fonctionnelle
Functional safety communication profiles	Profils de communication de sécurité fonctionnelle
IEC 61158 series / Fieldbus for use in industrial control systems	Série CEI 61158 / Bus de terrain pour utilisation dans des systèmes de commande industriels
IEC 61508 series, Functional safety (basic standard)	Série CEI 61508 Sécurité fonctionnelle (norme de base)
Functional safety for machinery .... for industrial environment)	Sécurité fonctionnelle des systèmes de commande électriques, électroniques et électroniques programmables (y compris les interférences électromagnétiques dans l'environnement industriel)
Key	Légende
(yellow) safety-related standards	(jaune) normes relatives à la sécurité
(blue) fieldbus-related standards	(bleu) normes relatives au bus de terrain
(dashed) yellow) this standard	(jaune pointillé) la présente norme
IEC	CEI

NOTE Les paragraphes 6.7.6.4 (complexité élevée) et 6.7.8.1.6 (faible complexité) de la CEI 62061 précisent la relation entre le niveau de performance PL (catégorie) et le niveau d'intégrité de sécurité SIL.

Figure 1 – Relations entre la CEI 61784-3 et d'autres normes (machines)

La Figure 2 illustre les relations entre la présente norme et les normes pertinentes relatives à la sécurité et aux bus de terrain dans un environnement de transformation.



Légende

Anglais	Français
Product standards	Normes de produits
Safety function, e.g. light curtains	Fonction de sécurité, par exemple barrières photo électriques
Safety for PLC (under consideration)	Sécurité relative aux automates programmables (à l'étude)
Safety functions for drives	Fonctions de sécurité applicables aux entraînements
Safety requirements for robots	Exigences de sécurité applicables aux robots
Security (profile-specific)	Sûreté (spécifique au profil)
Security (common part)	Sûreté (partie commune)
Installation guide (profile-specific)	Guide d'installation (spécifique au profil)
Installation guide (common part)	Guide d'installation (partie commune)
See safety standards for machinery (Figure 1)	Voir normes de sécurité pour les machines (Figure 1)
Valid also in process industries, whenever applicable	Valable également dans les industries de transformation, le cas échéant
Functional safety communication profiles	Profils de communication de sécurité fonctionnelle

Anglais	Français
IEC 61326-3-2 a) EMC and functional safety	CEI 61326-3-2 a) CEM & sécurité fonctionnelle
IEC 61158 series/ IEC 61784-1-2, Fieldbus for use in industrial control systems	Série CEI 61158/ CEI 61784-1,-2 Bus de terrain pour utilisation dans des systèmes de commande industriels
IEC 61508 series, Functional safety (basic standard)	Série CEI 61508 Sécurité fonctionnelle (norme de base)
IEC 61511 seriesb) Functional safety–safety instrumented systems for the process industry sector	Série CEI 61511b) sécurité fonctionnelle – systèmes instrumentés de sécurité pour le secteur des industries de transformation
US: ISA 84.00.1 (3 parts = modified IEC 61511)	US: ISA 84.00.1 (3 parties = CEI 61511 modifiée)
DE : VDI 2180 Part 1 –4	DE : VDI 2180 Parties 1 à 4
Key	Légende
(yellow) safety-related standards	(jaune) normes relatives à la sécurité
(blue) fieldbus-related standards	(bleu) normes relatives au bus de terrain
(dashed) yellow) this standard	(jaune pointillé) la présente norme

[a] Pour des environnements électromagnétiques spécifiés; autrement, la CEI 61326-3-1.

[b] EN ratifiée.

Figure 2 – Relations entre la CEI 61784-3 et d'autres normes (transformation)

Les couches de communication de sécurité mises en œuvre dans le cadre de systèmes relatifs à la sécurité conformément à la série CEI 61508, assurent la confiance nécessaire à accorder à la transmission de messages (information) entre deux participants ou plus sur un bus de terrain dans un système sécuritaire, ou une fiabilité suffisante dans le comportement de sécurité en cas d'erreurs ou de défaillances du bus de terrain.

Les couches de communication de sécurité spécifiées dans la présente norme permettent de garantir cette assurance en utilisant un bus de terrain dans des applications nécessitant une sécurité fonctionnelle jusqu'au niveau d'intégrité de sécurité (SIL) spécifié par son profil de communication de sécurité fonctionnelle correspondant.

La revendication du SIL qui en résulte pour un système dépend de la mise en œuvre du profil de communication de sécurité fonctionnelle retenu au sein du système – la mise en œuvre du profil de communication de sécurité fonctionnelle dans un dispositif normal ne suffit pas à le qualifier de dispositif de sécurité.

La présente norme décrit:

— les principes de base de mise en œuvre des exigences de la série CEI 61508 pour les communications de données relatives à la sécurité, y compris les défauts de transmission potentiels, les mesures correctives et les considérations concernant l'intégrité des données;

— la description individuelle des profils de sécurité fonctionnelle pour plusieurs familles de profils de communication dans les CEI 61784-1 et CEI 61784-2;

— les extensions de la couche de sécurité aux sections relatives au service et aux protocoles de communication de la série CEI 61158.

0.2 Déclaration de droits de propriété

La commission électrotechnique internationale (CEI) attire l'attention sur le fait qu'il est déclaré que la conformité avec le présent document peut impliquer l'utilisation de brevets concernant les profils de communication de sécurité fonctionnelle pour la famille 13 comme suit, où la notation [xx] désigne le détenteur des droits de propriété:

AT 31/2007	[BR]	Anordnung und ein Verfahren zur sicheren Datenkommunikation über ein nicht sicheres Netzwerk
DE 102004055978.3	[BR]	Verfahren zur Zeitsynchronisation innerhalb eines sicherheitsgerichteten Netzwerkes
DE 102004055685.7	[BR]	Verfahren zur Abgrenzung eines sicheren Netzwerkes
DE 102004055684.9	[BR]	Verfahren zur Absicherung des Datentransfers in einem sicheren Netzwerk mit CRC's variabler Länge
EP 08150038	[BR]	Arrangement and a method for safe data communication via a non-safe network
US 11/970178	[BR]	Arrangement and a method for safe data communication via a non-safe network

La CEI ne prend pas position eu égard à la preuve, la validité et la portée de ces droits de propriété.

Les détenteurs de ces droits de propriété ont donné l'assurance à la CEI qu'ils consentent à négocier des licences avec des demandeurs du monde entier, en des termes et à des conditions raisonnables et non discriminatoires. De ce fait, les déclarations des détenteurs desdits droits de brevets sont enregistrées auprès de la CEI.

Des informations peuvent être obtenues auprès de:

[BR] Bernecker + Rainer
Industrie-Elektronik Ges.m.b.H.
B&R Strassse 1
5142 Eggelsberg
AUTRICHE

Tél.: +43 7748 6586– 0
Télécopie: +43 7748 6586 – 26

L'attention est par ailleurs attirée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues autres que ceux identifiés ci-dessus. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

RÉSEAUX DE COMMUNICATION INDUSTRIELS – PROFILS –

Partie 3-13: Bus de terrain de sécurité fonctionnelle – Spécifications supplémentaires pour CPF 13

1 Domaine d'application

La présente partie de la série CEI 61784-3 spécifie une couche de communication sécuritaire (services et protocole) fondée sur la CPF 13 de la CEI 61784-2 et le type 13 de la CEI 61158. Elle identifie les principes applicables aux communications de sécurité fonctionnelle définies dans la CEI 61784-3, et appropriés à cette couche de communication de sécurité.

NOTE 1 Elle ne couvre pas les aspects relatifs à la sécurité électrique et à la sécurité intrinsèque. La sécurité électrique concerne les dangers tels que les chocs électriques. La sécurité intrinsèque concerne les dangers associés aux atmosphères explosibles.

La présente partie [1] définit les mécanismes de transmission des messages propres à la sécurité entre les participants d'un réseau réparti, en utilisant la technologie de bus de terrain conformément aux exigences de la série CEI 61508 [2] concernant la sécurité fonctionnelle. Ces mécanismes peuvent être utilisés dans diverses applications industrielles, telles que la commande de processus, l'usinage automatique et les machines.

La présente partie fournit des lignes directrices tant pour les développeurs que pour les évaluateurs de dispositifs et systèmes conformes.

NOTE 2 La revendication du SIL qui en résulte pour un système dépend de la mise en œuvre du profil de communication de sécurité fonctionnelle retenu au sein du système – la mise en œuvre du profil de communication de sécurité fonctionnelle, conforme à la présente partie, dans un dispositif normal ne suffit pas à le qualifier de dispositif de sécurité.

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application 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).

CEI 61131-3, *Programmable controllers – Part 3: Programming languages* (disponible uniquement en anglais)

IEC 61158-3-13, *Industrial communication networks – Fieldbus specifications – Part 3-13: Data-link layer service definition – Type 13 elements* (disponible uniquement en anglais)[3]

IEC 61158-4-13, *Industrial communication networks – Fieldbus specifications – Part 4-13: Data-link layer protocole specification – Type 13 elements (*disponible uniquement en anglais*)*

IEC 61158-5-13, *Industrial communication networks – Fieldbus specifications – Part 5-13: Application layer service definition – Type 13 elements (*disponible uniquement en anglais*)*

---

1 Dans les pages suivantes de la présente norme, "la présente partie" se substitue à "cette partie de la série CEI 61784-3".

2 Dans les pages suivantes de la présente norme, "CEI 61508" se substitue à "série CEI 61508".

3 Les publications monolingues des séries IEC 61158 et IEC 61784 sont actuellement en cours de traduction.

IEC 61158-6-13, *Industrial communication networks – Fieldbus specifications – Part 6-13: Application layer protocole specification – Type 13 elements* (disponible uniquement en anglais)

CEI 61508 (toutes les parties), *Sécurité fonctionnelle des systèmes électriques / électroniques / électroniques programmables relatifs à la sécurité*

IEC 61784-2, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3* (disponible uniquement en anglais)

IEC 61784-3:2010[4], *Industrial communication networks – Profiles – Part 3: Functional safety fieldbuses – General rules and profile definitions* (disponible uniquement en anglais)

IEC 61918, *Industrial communication networks – Installation of communication networks in industrial premises* (disponible uniquement en anglais)

ISO/CEI 19501, *Technologies de l'information – Traitement distribué ouvert – Langage de modélisation unifié (UML), version 1.4.2*

3 Termes, définitions, symboles, abréviations et conventions

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.1.1 Termes et définitions communs

3.1.1.1
disponibilité
probabilité, pour un système automatisé, qu'il ne se produise pas de conditions opérationnelles non satisfaisantes, telles que la perte de production, pendant une période donnée

3.1.1.2
canal noir
canal de communication sans preuve existante de conception ou de validation conformément à la CEI 61508

3.1.1.3
canal de communication
connexion logique entre deux points limites d'un *système de communication*

3.1.1.4
système de communication
disposition de matériels, logiciels et vecteurs de propagation destinée à permettre la transmission de *messages* (couche d'application définie dans l'ISO/CEI 7498) d'une application à une autre

3.1.1.5
connexion
liaison logique entre deux objets d'application de dispositifs identiques ou différents

[4] En cours d'élaboration.

3.1.1.6
contrôle de redondance cyclique (CRC[5])
<valeur> donnée redondante déduite, et enregistrée ou transmise simultanément, d'un bloc de données afin de détecter toute corruption des données

<méthode> procédure utilisée pour calculer les données redondantes

NOTE 1 Les termes « code CRC » et « signature CRC », et les étiquettes telles que CRC1, CRC2, peuvent également être utilisés dans la présente norme pour faire référence aux données redondantes.

NOTE 2 Voir également [37], [38]][6].

3.1.1.7
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

[CEI 61508-4:2010[7]], [CEI 61158]

NOTE 1 Les erreurs peuvent être dues à des erreurs de conception du matériel/logiciel et/ou des informations altérées du fait de perturbations électromagnétiques et/ou autres effets.

NOTE 2 Les erreurs ne produisent pas nécessairement une *défaillance* ou une *panne*.

3.1.1.8
défaillance
cessation de l'aptitude d'une unité fonctionnelle à accomplir une fonction requise ou fonctionnement d'une unité fonctionnelle d'une toute autre manière que celle requise

NOTE 1 La définition de la CEI 61508-4 est identique avec des notes complémentaires.

[CEI 61508-4:2010, modifiée], [ISO/CEI 2382-14.01.11, modifiée]

NOTE 2 Une défaillance peut être due à une *erreur* (par exemple, problème de conception du matériel/logiciel ou rupture de message).

3.1.1.9
panne
condition anormale susceptible de provoquer la réduction ou la perte de capacité d'une unité fonctionnelle à accomplir une fonction requise

NOTE Le VEI 191-05-01 définit la « panne » comme un état caractérisé par l'incapacité à accomplir une fonction requise, à l'exclusion de l'incapacité au cours de la période de maintenance préventive ou autres actions planifiées, ou du fait de l'absence de ressources externes.

[CEI 61508-4:2010, modifiée], [ISO/CEI 2382-14.01.10, modifiée]

3.1.1.10
bus de terrain
système de communication reposant sur le transfert de données en série et utilisé dans des applications d'automatisation industrielle ou de commande de processus

3.1.1.11
système de bus de terrain
système utilisant un *bus de terrain* avec des dispositifs connectés

---

5 CRC = *Cyclic Redundancy Check*

6 Les chiffres entre crochets font référence à la bibliographie.

7 A publier.

3.1.1.12
trame
synonyme discrédité de DLPDU

3.1.1.13
fonction de hachage
fonction (mathématique) de mise en correspondance des valeurs d'un ensemble (éventuellement) très grand de valeurs en une plage de valeurs (habituellement) plus petite

NOTE 1 Les fonctions de hachage peuvent être utilisées pour déterminer l'altération de données.

NOTE 2 Les fonctions de hachage courantes incluent la parité, la somme de contrôle ou le CRC.

[CEI/TR 62210, modifiée]

3.1.1.14
danger
état ou ensemble de conditions d'un système qui, avec d'autres conditions associées, entraîne inévitablement un préjudice pour les personnes, les biens ou l'environnement

3.1.1.15
message
série ordonnée d'octets destinée à communiquer des informations
[ISO/CEI 2382-16.02.01, modifiée]

3.1.1.16
niveau de performance (PL[8])
niveau discret utilisé pour spécifier la capacité des parties relatives à la sécurité des systèmes de commande à accomplir une fonction de sécurité dans des conditions prévisibles
[ISO 13849-1]

3.1.1.17
redondance
existence de moyens, outre les moyens qui se révéleraient suffisants pour qu'une unité fonctionnelle accomplisse une fonction requise ou que des données représentent une information

NOTE La définition de la CEI 61508-4 est identique, avec des exemples et des notes supplémentaires.

[CEI 61508-4:2010, modifiée], [ISO/CEI 2382-14.01.12, modifiée]

3.1.1.18
fiabilité
probabilité qu'un système automatisé puisse accomplir une fonction requise dans des conditions données pendant un intervalle de temps donné (t1, t2)

NOTE 1 On suppose, en général, que le système automatisé est en état d'accomplir la fonction requise au début de l'intervalle.

NOTE 2 le terme «fiabilité» est aussi employé en français pour désigner la performance de fiabilité quantifiée par cette probabilité.

NOTE 3 Au cours de la période MTBF (temps moyen entre défaillances) ou MTTF (durée moyenne de fonctionnement avant défaillance), il y a réduction de la probabilité qu'un système automatisé puisse accomplir une fonction requise dans des conditions données.

NOTE 4 La fiabilité diffère de la disponibilité.

[CEI 62059-11, modifiée]

8 PL = *Performance Level*

3.1.1.19
risque
combinaison de la probabilité d'occurrence d'un dommage ou préjudice et de la gravité de ce dernier

NOTE Pour plus d'informations sur ce concept, se reporter à l'Annexe A de la CEI 61508-5:2010[9].

[CEI 61508-4:2010], [ISO/CEI Guide 51:1999, définition 3.2]

3.1.1.20
couche de communication de sécurité (SCL[10])
couche de communication qui comprend toutes les mesures nécessaires permettant d'assurer la transmission de données en toute sécurité conformément aux exigences de la CEI 61508

3.1.1.21
données de sécurité
données transmises par un réseau de sécurité utilisant un protocole de sécurité

NOTE La couche de communication de sécurité ne garantit pas la sécurité des données proprement dites, mais uniquement la transmission en toute sécurité de ces dernières.

3.1.1.22
dispositif de sécurité
dispositif conçu conformément à la CEI 61508 et qui met en œuvre le profil de communication de sécurité fonctionnelle

3.1.1.23
fonction de sécurité
fonction à réaliser par un système E/E/PE relatif à la sécurité ou par un dispositif externe de réduction de risque, prévue pour assurer ou maintenir un état de sécurité de l'EUC par rapport à un événement dangereux spécifique

NOTE La définition donnée dans la CEI 61508-4 est identique avec ajout d'un exemple et d'une référence.

[CEI 61508-4:2010, modifiée]

3.1.1.24
temps de réponse de la fonction de sécurité
temps écoulé du cas le plus défavorable suite à l'activation d'un capteur de sécurité relié à un bus de terrain, avant que ne soit atteint l'état de sécurité correspondant de son (ses) actionneur(s) de sécurité, du fait d'erreurs ou de défaillances avérées dans le canal de fonction de sécurité

NOTE Ce concept est introduit dans le paragraphe 5.2.4 de la CEI 61784-3:2010[11] et traité dans le cadre des profils de communication de sécurité fonctionnelle définis dans la présente partie.

3.1.1.25
niveau d'intégrité de sécurité (SIL[12])
niveau discret (un sur quatre niveaux possibles), correspondant à une plage de valeurs d'intégrité de sécurité, où le niveau d'intégrité de sécurité 4 est le niveau le plus élevé et le niveau d'intégrité de sécurité 1 est le niveau le plus faible

---

9 A publier.

10 SCL = *Safety Communication Layer*

11 En cours d'élaboration.

12 SIL = *Safety Integrity Level*

NOTE 1 Les mesures cible des défaillances (voir 3.5.17 dans la CEI 61508-4:2010) applicables aux quatre niveaux d'intégrité de sécurité sont spécifiées dans les Tableaux 2 et 3 de la CEI 61508-1:2010[13].

NOTE 2 Les niveaux d'intégrité de sécurité sont utilisés pour spécifier les exigences d'intégrité de sécurité des fonctions équivalentes à attribuer aux systèmes E/E/PE relatifs à la sécurité.

NOTE 3 Le niveau d'intégrité de sécurité (SIL) n'est pas une propriété d'un système, sous-système, élément ou composant. L'interprétation correcte de l'expression « système sécuritaire avec SILn » (où n est égal à 1, 2, 3 ou 4) signifie que le système est potentiellement capable de prendre en charge des fonctions de sécurité avec un niveau d'intégrité de sécurité jusqu'à n.

[CEI 61508-4:2010]

3.1.1.26
mesure de sécurité
<la présente norme> mesure permettant de contrôler les *erreurs* de communication éventuelles, qui est conçue et mise en œuvre conformément aux exigences de la CEI 61508

NOTE 1 Dans la pratique, plusieurs mesures de sécurité sont combinées pour atteindre le niveau d'intégrité de sécurité requis.

NOTE 2 Les *erreurs* de communication et les mesures de sécurité associées sont détaillées en 5.3 et 5.4 de la CEI 61784-3:2010.

3.1.1.27
application relative à la sécurité
programmes conçus conformément à la CEI 61508 pour satisfaire aux exigences SIL de l'application

3.1.1.28
système relatif à la sécurité
système qui exécute des *fonctions de sécurité* conformément à la CEI 61508

3.1.1.29
esclave
entité de communication passive capable de recevoir des messages et de les envoyer en réponse à une autre entité de communication qui peut être un maître ou un esclave

3.1.1.30
horodatage
informations temporelles incluses dans un *message*

3.1.2 CPF 13: Termes et définitions supplémentaires

3.1.2.1
taux d'erreurs sur les bits
probabilité d'une interprétation erronée de bit due au bruit électrique

3.1.2.2
informations de fournisseur de dispositif (DVI[14])
informations d'identification d'un nœud de sécurité

3.1.2.3
adresse de sécurité (SADR[15])
adresse d'un nœud de sécurité dans un domaine de sécurité

---

13 A publier.

14 DVI = *Device Vendor Information*

15 SADR = *Safety Address*

3.1.2.4
gestionnaire de configuration de sécurité (SCM[16])
service chargé de gérer les services relatifs à la sécurité

NOTE Les services sécuritaires gérés sont, par exemple, la configuration, le paramétrage, etc.

3.1.2.5
nœud de sécurité (SN[17])
nœud ayant une fonction de sécurité

3.1.2.6
domaine de sécurité (SD[18])
espace d'adresse qui peut recevoir jusqu'à 1023 nœuds de sécurité

3.1.2.7
passerelle de domaine de sécurité (SDG[19])
passerelle permettant le partage de données entre différents domaines de sécurité

3.1.2.8
numéro de domaine de sécurité (SDN[20])
identifiant d'un domaine de sécurité

3.1.2.9
gestion du réseau de sécurité (SNMT[21])
services permettant de gérer la couche de sécurité

3.1.2.10
dictionnaire d'objets de sécurité (SOD[22])
référentiel de tous les objets de données accessibles au moyen des services du bus de terrain de sécurité fonctionnelle

3.1.2.11
objet de données de processus de sécurité (SPDO[23])
objet permettant l'échange de données de processus entre différents nœuds de sécurité

3.1.2.12
objet de données de service de sécurité (SSDO[24])
objet permettant l'échange de données de service entre des nœuds de sécurité

3.1.2.13
initialisation
état initial de communication d'un nœud de sécurité

16 SCM = *Safety Configuration Manager*

17 SN = *Safety Node*

18 SD = *Safety Domain*

19 SDG = *Safety Domain Gateway*

20 SDN = *Safety Domain Number*

21 SNMT = *Safety Network Management*

22 SOD = *Safety Object Dictionary*

23 SPDO = *Safety Process Data Object*

24 SSDO = *Safety Service Data Object*

3.1.2.14
opérationnel
état de communication d'un nœud de sécurité dans lequel toutes les fonctions de sécurité fonctionnelle opèrent correctement

3.1.2.15
pré-opérationnel
état de communication d'un nœud de sécurité dans lequel toutes les sorties d'application sont à l'état de sécurité

3.1.2.16
identifiant unique de dispositif (UDID[25])
identifiant unique et universel d'un dispositif donné

3.2 Symboles et abréviations

3.2.1 Symboles et abréviations communs

CP	Profil de communication (*Communication Profile*)	[CEI 61784-1]
CPF	Famille de profils de communication (*Communication Profile Family*)	[CEI 61784-1]
CRC	Contrôle de redondance cyclique (*Cyclic Redundancy Check*)	
DLL	Couche de liaison de données (*Data Link Layer*)	[ISO/CEI 7498-1]
DLPDU	Unité de données de protocole de liaison de données (*Data Link Protocol Data Unit*)	
CEM	Compatibilité électromagnétique	
EUC	Equipement commandé (*Equipment Under Control*)	[CEI 61508-4:2010]
E/E/PE	Électrique/électronique/électronique programmable (*Electrical/Electronic/Programmable Electronic*)	[CEI 61508-4:2010]
FAL	Couche Application de bus de terrain (*Fieldbus Application Layer*)	[CEI 61158-5]
SF	Sécurité Fonctionnelle	
FSCP	Profil de communication de sécurité fonctionnelle (*Functional Safety Communication Profile*)	
MTBF	Temps moyen entre défaillances (*Mean Time Between Failures*)	
MTTF	Durée moyenne de fonctionnement avant défaillance (*Mean Time To Failure*)	
PDU	Unité de données de protocole (*Protocol Data Unit*)	[ISO/CEI 7498-1]
PFH	Fréquence moyenne de défaillance dangereuse [h-1] par heure	[CEI 61508-6:2010[26]]
PhL	Couche physique (*Physical Layer*)	[ISO/CEI 7498-1]
PL	Niveau de performance (*Performance Level*)	[ISO 13849-1]
PLC	Automate programmable (*Programmable Logic Controller*)	
SCL	Couche de communication de sécurité (*Safety Communication Layer*)	
SIL	Niveau d'intégrité de sécurité (*Safety Integrity Level*)	[CEI 61508-4:2010]

3.2.2 CPF 13: Symboles et abréviations supplémentaires

ACM	Mode de configuration automatique (*Automatic Configuration Mode*)	
ADR	Informations d'adresse (origine ou destination)	
APDU	Unité de données de protocole d'application (*Application Protocol Data Unit*)	[ISO/CEI 7498-1]
CN	Nœud commandé (*Controlled Node*)	[CEI 61158-3-13]

25 UDID = *Unique Device Identification*

26 A publier.

CT	Champ temps consécutifs (*Consecutive Time*)	
DBx	Octet de données	
DSADR	SADR de destination (*Destination SADR*)	
DVI	Informations de fournisseur de dispositif (*Device Vendor Information*)	
ID	Identifiant de trame	
LE	Champ longueur	
LSB	Octet de poids faible (*Least Significant Byte*)	
MAC	Couche d'accès au support (*Medium Access Layer*)	
MCM	Mode de configuration manuel (*Manual Configuration Mode*)	
MN	Nœud gestionnaire (*Managing Node*)	[CEI 61158-3-13]
MSB	Octet de poids fort (*Most Significant Byte*)	
PDO	Objet de données de processus (*Process data object*)	[CEI 61158-3-13]
RxSPDO	Objet de données de processus de sécurité réception	
SADR	Adresse de sécurité (*Safety Address*)	
SCM	Gestionnaire de configuration de sécurité (*Safety Configuration Manager*)	
SCT	Temps de contrôle de sécurité (*Safety Control Time*)	
SD	Domaine de sécurité (*Safety Domain*)	
SDG	Passerelle de domaine de sécurité (*Safety Domain Gateway*)	
SDN	Numéro de domaine de sécurité (*Safety Domain Number*)	
SDO	Objet de données de service (*Service Data Object*)	[CEI 61158-3-13]
SN	Nœud de sécurité (*Safety Node*)	
SNMT	Gestion de réseau de sécurité (*Safety Network Management*)	
SOD	Dictionnaire d'objets de sécurité	
SPDO	Objet de données de processus de sécurité (*Safety process data object*)	
SPST	Outil logiciel de paramétrage de sécurité (*Safety Parameterization Software Tool*)	
SSDO	Objet de données de service de sécurité (*Safety Service Data Object*)	
SSADR	SADR d'origine (*Source SADR*)	
TADR	SADR supplémentaire pour informations temporelles	
TR	Numéro distinctif de demande de temps	
TReq	Demande de synchronisation temporelle	
TRes	Réponse de synchronisation temporelle	
TÜV	Technischer Überwachungsverein	
TxSPDO	Objets de données de processus de transmission de sécurité (*Safety Transmit Process data objects*)	
UDID	Identifiant unique de dispositif (*Unique Device Identification*)	

3.3 Conventions

3.3.1 Valeurs hexadécimales

Certaines valeurs dans la présente partie sont spécifiées en notation hexadécimale. La lettre "h" doit dans ce cas être jointe à la valeur proprement dite. Les caractères 1, 2, 3, 4, 5, 6, 8, 9, 0, a, b, c, d, e, f, A, B, C, D, E et F doivent être utilisés. Toutes les valeurs en notation hexadécimale doivent décrire des octets complets; par conséquent, une représentation par une valeur hexadécimale est toujours constituée d'un nombre pair de caractères.

EXEMPLE 1 12ABh, 05h, 1234ABCDh

Pour décrire des données dans lesquelles une suite d'octets doit être explicitement définie – pour être indépendante de la représentation exacte de la mémoire (gros-boutiste ou petit-boutiste) – on doit toujours utiliser une notation en séquence hexadécimale. Chaque octet est représenté par une valeur hexadécimale à deux chiffres (sans le suffixe h); les représentations à un seul octet doivent être séparées par un tiret (-).

EXEMPLE 2 00-00-00-00-00-00, AC-3E

3.3.2 Valeurs binaires

Certaines valeurs dans la présente partie sont spécifiées en notation binaire. La lettre "b" doit dans ce cas être jointe à la valeur proprement dite. Les caractères 1 et 0 doivent être utilisés.

EXEMPLE 11001010b

3.3.3 Chiffres en caractères joker (métacaractères)

Si des parties de valeurs constantes sont “sans effet” ou ont une signification supplémentaire hors du cadre du texte descriptif, elles doivent être représentées par un x à la place du chiffre correspondant.

EXEMPLE 12xBh, 1x0xx010b, 123x45

3.3.4 Diagrammes

Le cas échéant, dans l'ensemble de la présente partie, les notations graphiques conformes à l'ISO/CEI 19501 sont utilisées.

4 Vue d’ensemble de FSCP 13/1 (Ethernet POWERLINK safety)

4.1 Profil de communication de sécurité fonctionnelle 13/1

La famille de profils de communication 13 (communément appelée Ethernet POWERLINK[27]) définit des profils de communication sur la base des normes CEI 61158-3-13, CEI 61158-4-13, CEI 61158-5-13 et CEI 61158-6-13.

Le profil de base CP 13/1 est défini dans la CEI 61784-2. Le profil de communication de sécurité fonctionnelle FSCP 13/1 CPF 13 (Ethernet POWERLINK safety) repose sur les profils de base de la famille CPF 13 spécifiés dans la CEI 61784-2 et sur les spécifications de la couche de communication de sécurité définies dans la présente partie.

4.2 Aperçu technique

Le FSCP 13/1 définit un protocole de sécurité ouvert pour des communications à base Ethernet de l'ordre de la microseconde.

Les services et le protocole FSCP 13/1 spécifient une communication de données sécuritaire entre nœuds de sécurité (SN). Le protocole FSCP 13/1 est conçu pour assurer une fonction de sécurité de niveau SIL 3 conformément à la CEI 61508. La configuration du réseau est réalisée au moyen du Gestionnaire de configuration de sécurité (SCM). Des services de Gestion de réseau de sécurité (SNMT) sont utilisés pour le diagnostic au démarrage et dans l'environnement d'exécution des réseaux FSCP 13/1. Le dictionnaire d’objets de sécurité

---

27 Ethernet POWERLINK et Ethernet POWERLINK safety désignent des appellations commerciales de l'organisation à but non lucratif EPSG (Ethernet POWERLINK Standardization Group - Groupe de normalisation de l'Ethernet POWERLINK). .. Cette information est donnée à l’intention des utilisateurs de la présente Norme internationale et ne signifient nullement que la CEI approuve ou recommande le détenteur de la marque ou de l’un quelconque de ses produits. La conformité à la présente norme ne nécessite pas l’utilisation des marques commerciales Ethernet POWERLINK ou Ethernet POWERLINK safety. L’utilisation des marques commerciales Ethernet POWERLINK ou Ethernet POWERLINK safety nécessite l’autorisation d’Ethernet POWERLINK Standardization Group (EPSG).

(SOD) est une base de données fournie par chaque SN contenant des données de configuration et des données utilisateur. Les objets de données de service de sécurité (SSDO) sont utilisés pour l'échange de données spontanées. Les objets de données de processus de sécurité (SPDO) fournissent des services d'échanges de données synchrones.

La communication de données synchrones entre SN est modélisée conformément au modèle éditeur abonné (voir la Figure 3), tandis que la communication des données spontanées utilise le modèle client-serveur (voir la Figure 4).

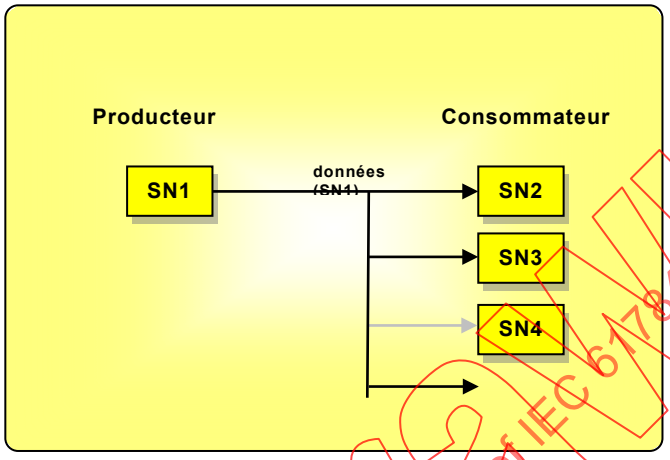


Figure 3 – Exemple de relation producteur/consommateur

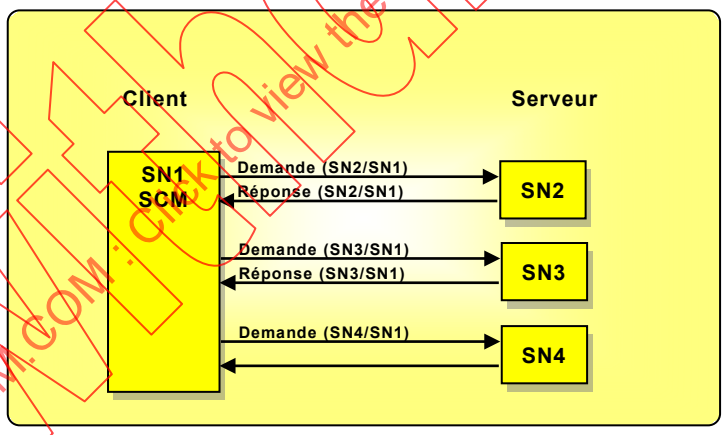


Figure 4 – Exemple de relation client/serveur

5 Généralités

5.1 Documents externes de spécifications applicables au profil

Les documents [51] et [52] peuvent aider à comprendre les concepts du FSCP 13/1.

5.2 Exigences fonctionnelles de sécurité

Les exigences suivantes ont été utilisées pour l'élaboration du FSCP 13/1 et doivent s'appliquer à la conception de dispositifs mettant en œuvre le protocole FSCP 13/1:

- Le protocole FSCP 13/1 doit prendre en charge le niveau d'intégrité de sécurité 3 (SIL 3) conformément à la CEI 61508.

- Les exigences fondamentales de développement du FSCP 13/1 sont spécifiées dans la CEI 61784-3.
- Le FSCP 13/1 utilise des services définis dans la CEI 61158-5-13.
- Le protocole FSCP 13/1 est fondé sur une approche "canal noir"; aucune dépendance sécuritaire ne s'applique au profil de communication CPF 13.
- Les mises en œuvre du FSCP 13/1 doivent être conformes à la CEI 61508.
- Les exigences de la CPF 13 doivent demeurer inchangées en termes de sécurité, sauf spécification contraire dans la présente partie.
- Les communications de sécurité et normale doivent être indépendantes. Cependant, les dispositifs normalisés et les dispositifs FSCP 13/1 doivent pouvoir utiliser le même canal de communication.
- Il doit être possible d'utiliser des dispositifs d'infrastructure de réseaux non sécuritaires pour construire un réseau FSCP 13/1.
- Un domaine de sécurité donné doit prendre en charge jusqu'à 1 023 dispositifs sécuritaires.
- Jusqu'à 1 023 domaines de sécurité doivent pouvoir être pris en charge.
- La largeur de bande utilisée par les messages de sécurité doit être aussi faible que possible; les PDU de sécurité doivent être optimisées pour les dispositifs utilisant moins de données (par exemple des E/S numériques), et
- Sauf spécification contraire dans la présente partie, l'état de sécurité de toute valeur doit être zéro (0).

5.3 Mesures de sécurité

Le profil FSCP 13/1 doit utiliser les mesures de sécurité énumérées dans le Tableau 1 pour la détection d'erreurs de communication dans des données cycliques transmises dans des SPDO (voir 7.2). Les mesures de sécurité doivent être traitées et surveillées au sein de chaque dispositif FSCP 13/1.

Tableau 1 – Erreurs de communication et mesures de détection (cycliques)

Erreurs de communication	Mesures de sécurité							
	Numéro de séquence	Horodatage	Délai d'attente	Authentification de connexion	Message de retour d'informations	Assurance d'intégrité des données	Redondance avec vérification par recoupement	Différents systèmes d'assurance d'intégrité des données
Altération						X	X	
Répétition non prévue		X						
Séquence incorrecte		X						
Perte			X [a]					
Retard inacceptable		X	X					
Insertion			X[b]					
Déguisement			X			X	X	X
Adressage				X				

NOTE Les erreurs "perte " et "insertion" ne seront pas détectées par un horodatage. Une perte de données est détectée dans le délai maximal de réaction par un temporisateur de surveillance. Si un message a été envoyé plusieurs fois (insertion), l'horodatage reste inchangé et le dispositif de réception ignore ce message.

[a] Des PDU manquantes doivent être détectées dans le délai maximal de réaction.
[b] Un seul message doit être accepté au cours d'un créneau défini.

Le profil FSCP 13/1 doit utiliser les mesures de sécurité énumérées dans le Tableau 2 pour la détection d'erreurs de communication dans des données acycliques transmises dans des SSDO (voir 7.3) ou dans des SNMT (voir 7.4). Les mesures de sécurité doivent être traitées et surveillées au sein de chaque dispositif FSCP 13/1.

Tableau 2 – Erreurs de communication et mesures de détection (acycliques)

Erreurs de communication	Mesures de sécurité							
	Numéro de séquence	Horodatage	Délai d'attente	Authentification de connexion	Message de retour d'informations	Assurance d'intégrité des données	Redondance avec vérification par recoupement	Différents systèmes d'assurance d'intégrité des données
Altération					X	X	X	
Répétition non prévue	X							
Séquence incorrecte	X							
Perte	X				X			
Retard inacceptable								
Insertion	X				X			
Déguisement						X	X	X
Adressage				X				
NOTE Les erreurs “perte ” et “insertion” ne seront pas détectées par un horodatage. Une perte de données est détectée dans le délai maximal de réaction par un temporisateur de surveillance. Si un message a été envoyé plusieurs fois (insertion), l'horodatage reste inchangé et le dispositif de réception ignore ce message.								

Pour éviter des erreurs systématiques et stochastiques, les recommandations suivantes doivent être envisagées lors de l'élaboration et de la mise en œuvre du FSCP 13/1 (voir 7.1):

- Le producteur des données de sécurité doit:
  - générer un horodatage,

    NOTE Toute nouvelle SPDU doit contenir un nouvel horodatage.
  - générer des CRC et
  - dupliquer le message de sécurité.
- Le consommateur de données de message de sécurité doit:
  - vérifier la partie une de la PDU avec le CRC,
  - vérifier la partie deux de la PDU avec le CRC,
  - comparer les zones des parties une et deux de la PDU bit par bit,
  - vérifier l'adresse reçue,
  - réinitialiser le chien de garde (compteur) interne si une nouvelle PDU valide a été reçue,
  - si une PDU porte le même horodatage que la dernière PDU reçue ou qu'une ancienne PDU, la PDU doit être ignorée,
  - si aucune PDU valide n'est reçue dans un délai défini (temps de contrôle de sécurité, voir 7.5.4.14), le dispositif doit passer à l'état de sécurité, et
  - si le chien de garde (compteur écoulé) présente un dépassement (absence de réinitialisation dans le délai maximal), le dispositif doit passer à un état de sécurité.

5.4 Structure de la couche de communication de sécurité

Le protocole FSCP 13/1 est placé en couche supérieure sur un canal de communication normalisé, qui doit être considéré comme un canal noir. La Figure 5 illustre la relation entre la couche de communication de sécurité et le canal de communication normalisé.

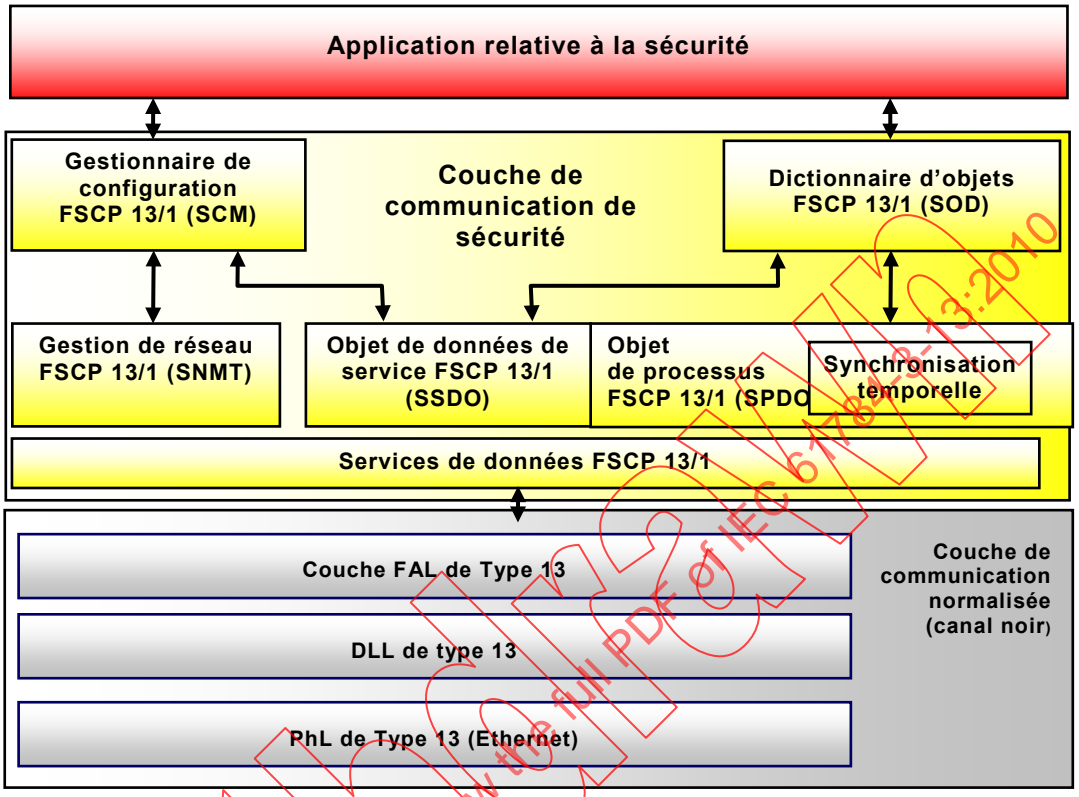


Figure 5 – Structure de la couche de communication

Les données de sécurité sont produites par l'entité d'application sécuritaire. Ces données sont encapsulées dans un message de sécurité par les couches de communication de sécurité et envoyées via le canal de communication normalisé. Une couche de communication de sécurité dans le dispositif de réception décode le message de sécurité et vérifie l'intégrité ainsi que l'actualité des données. Les données sont présentées à l'entité d'application sécuritaire. Si aucune donnée n'est reçue dans un créneau temporel défini, l'entité d'application est informée de cet événement, qui peut être utilisé pour mettre toutes les sorties à l'état de sécurité.

La couche de communication de sécurité assure:

— le transport des données de sécurité (intégrité et surveillance temporelle),
— l'amorçage du réseau,
— le diagnostic de l'environnement d'exécution,
— la mise en service,
— le téléchargement des programmes et microprogrammes,
— le paramétrage, et
— le remplacement du dispositif.

Il est possible d'accéder à toutes ces fonctions et de les contrôler via le canal de communication de sécurité (voir Figure 6).

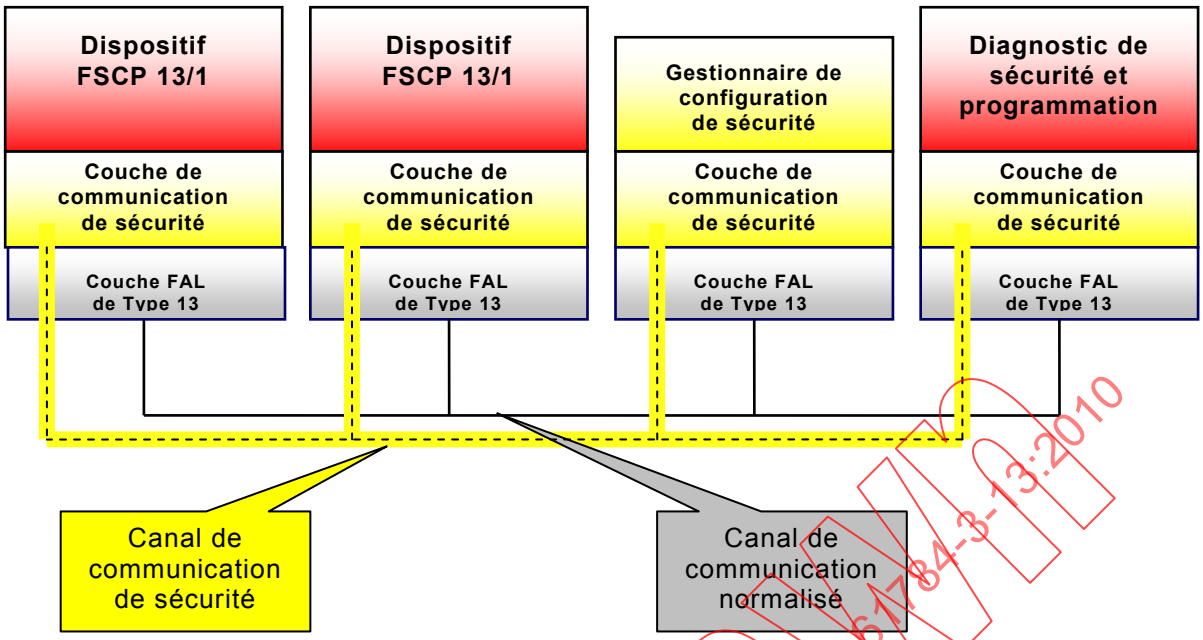


Figure 6 – Canal de communication de sécurité

5.5 Relations avec la FAL (et DLL, PhL)

5.5.1 Généralités

Le FSCP 13/1 doit utiliser les services définis dans la CEI 61158-5-13.

5.5.2 Types de données

Les profils définis dans la présente partie prennent en charge tous les types de données CPF 13 définis dans la CEI 61158-5-13.

6 Services de la couche de communication de sécurité

6.1 Modélisation

6.1.1 Modèle de référence

6.1.1.1 Généralités

Une mise en œuvre conforme du FSCP 13/1 doit fournir les objets d'application suivants:

- la Gestion de réseau de sécurité (SNMT),
- le Gestionnaire de configuration de sécurité (SCM),
- les objets de données de service de sécurité (SSDO),
- le dictionnaire d'objets de sécurité (SOD), et
- les objets de données de processus de sécurité (SPDO).

La Figure 5 illustre les relations entre ces objets d'application.

L'utilisation d'une couche de communication sous-jacente doit être obligatoire.

6.1.1.2 Gestion de réseau de sécurité (SNMT)

Le SNMT doit fournir les services requis pour que le Gestionnaire de configuration de sécurité (SCM) configure et vérifie les nœuds sécuritaires au sein du réseau.

6.1.1.3 Objets de données de service de sécurité (SSDO)

Le SSDO doit fournir les services requis pour l'accès au dictionnaire d'objets de sécurité (SOD) et la prise en charge du SCM.

6.1.1.4 Objets de données de processus de sécurité (SPDO)

Le SPDO doit fournir les services requis pour l'échange de données de sécurité entre certaines entrées dans le SOD des nœuds communicants. Les services de synchronisation temporelle, qui font partie des SPDO, doivent vérifier les performances du réseau entre nœuds communicants.

6.1.1.5 Dictionnaire d'objets de sécurité (SOD)

Le SOD doit décrire le format ainsi que les services d'accès des données spécifiques au dispositif.

6.1.1.6 Gestionnaire de configuration de sécurité (SCM)

Le SCM doit fournir les services requis de démarrage, de paramétrage et toute mesure ultérieure permettant d'assurer un fonctionnement sûr pendant toute la durée du cycle de vie de l'application.

6.1.2 Modèle de communication

Le FSCP 13/1 doit utiliser les services FAL définis dans la CEI 61158-5-13 pour l'échange de données entre nœuds de sécurité (SN); le FSCP 13/1 définit un modèle de communication producteur / consommateur construit en utilisant ces services. Les nœuds FSCP 13/1 doivent être logiquement séparés en nœuds producteurs et consommateurs.

Les nœuds producteurs doivent envoyer des données identifiées par une adresse de sécurité spécifique (SADR). Tout SN dans le domaine de sécurité peut recevoir ces données. Ceci est réalisé au moyen du mécanisme décrit en 7.5. Le modèle de communication permet, au moyen de SADR, d'identifier de manière unique un SN ainsi qu'une PDU de sécurité envoyée à partir d'un nœud spécifique.

La Figure 7 illustre une communication type entre producteur et consommateur. Le nœud producteur SN1 envoie des données. Les données sont identifiées de manière unique par les SADR des producteurs. Chaque nœud dans le domaine de sécurité est capable de recevoir ces données.

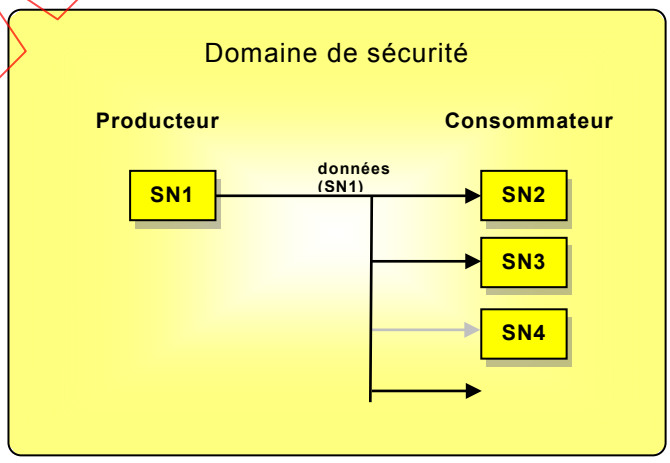


Figure 7 – Communication type entre producteur et consommateur

La Figure 8 présente une version étendue du modèle de communication producteur / consommateur qui permet au producteur de générer des données séparées pour des consommateurs individuels. A cette fin, le producteur doit utiliser des SADR. Le Gestionnaire de configuration de sécurité (SCM) doit s'assurer que les SADR utilisées sont toujours uniques au sein du domaine de sécurité.

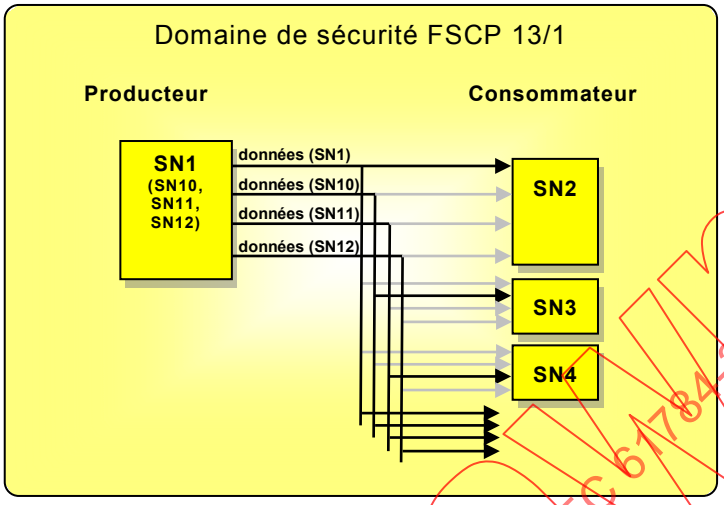


Figure 8 – Communication étendue entre producteur et consommateur

La Figure 9 illustre la communication client / serveur qui doit être utilisée pour les services de configuration et la gestion du réseau. Le SCM (ou un outil de configuration dédié, voir l'Article 8) doit agir comme client. Les SN doivent se limiter au rôle de serveur.

En utilisant le modèle de communication client / serveur, le client (SCM ou outil) envoie une demande au serveur (SN) qui est identifié dans le champ ADR (voir 7.1.2) en utilisant l'adresse du serveur et qui contient également l'adresse du client dans le champ TADR (voir 7.1.8). Le serveur envoie une réponse destinée à l'adresse indiquée dans le champ ADR (voir 7.1.2) en utilisant l'adresse du serveur et contient également l'adresse du client dans le champ TADR (voir 7.1.8).

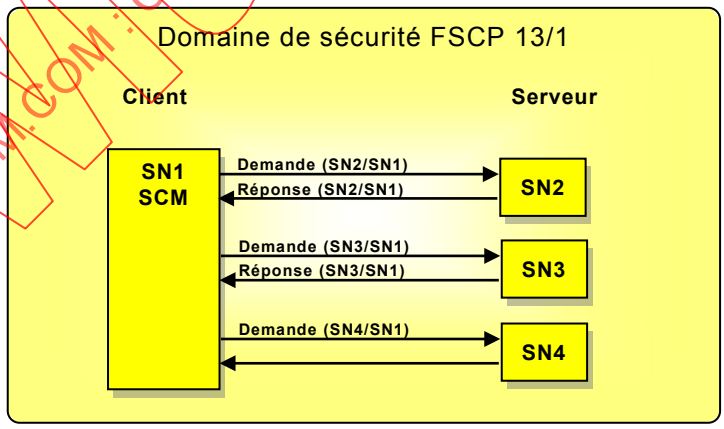


Figure 9 – Communication client / serveur

6.1.3 Rôles des dispositifs et topologie

6.1.3.1 Généralités

Le Tableau 3 définit les rôles des dispositifs FSCP 13/1.

Tableau 3 – Rôles des dispositifs

Rôle du dispositif	Description
Nœud de sécurité (SN)	Un dispositif conforme au FSCP 13/1
Gestionnaire de configuration de sécurité (SCM)	Service centralisé de gestion d'un domaine de sécurité (SD) [a]
Passerelle de domaine de sécurité (SDG: Safety Domain Gateway)	Dispositif FSCP 13/1 permettant l'échange de données entre différents SD
[a] Un domaine de sécurité définit un espace d'adresse isolé pouvant accepter jusqu'à 1 023 SN.	

Un dispositif FSCP 13/1 doit mettre en œuvre le rôle du dispositif d'un SN et peut prendre le rôle de SCM et/ou de SDG.

La Figure 10 donne une vue d'ensemble des relations possibles entre rôles des dispositifs et domaines de sécurité. SD1 est un exemple de domaine de sécurité simple. La communication sécuritaire des dispositifs FSCP 13/1 dans le SD1 est limitée aux SN dans SD1.

L'association de SD2 avec SD3 illustre la possibilité d'établir une communication FSCP 13/1 entre des domaines de sécurité différents. Ceci est réalisé en utilisant une SDG spéciale capable de communiquer avec différents domaines de sécurité. La communication entre domaines est traitée en interne par le dispositif. La capacité SDG est indépendante de la capacité SCM d'un dispositif. La SDG doit avoir une adresse SADR dans SD2 et une adresse SADR dans SD3.

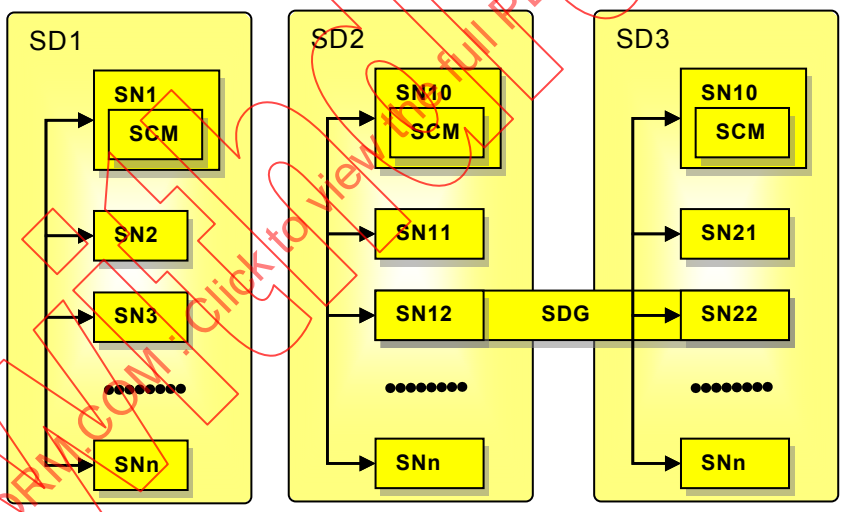


Figure 10 – Vue d'ensemble de la topologie

6.1.3.2 Nœud de sécurité (SN)

Un SN est un nœud dans n'importe quel réseau conforme à la présente partie. Un SN doit être identifié par:

- Une adresse logique unique au sein d'un SD donné, appelée adresse de sécurité (SADR). La plage de numérotation de l'adresse SADR doit être de 1 à 1 023 (0 est réservé et ne doit pas être utilisé). L'adresse SADR doit être définie lors de l'étape de planification de l'application. Des SADR dupliquées dans un SD donné, doivent être détectées par le SCM lors de la vérification du réseau au moment du démarrage du SCM ou du démarrage de tout nœud dans le domaine de sécurité.
- Une adresse physique est globalement unique dans tous les dispositifs conformes au FSCP 13/1, appelée Identifiant unique de dispositif (UDID). Les UDID dupliqués dans un SD donné doivent être détectés par le SCM.

Pour fonctionner, un SN peut nécessiter d'autres données, telles que les paramètres sécuritaires ou les données d'application. Il dépend de la capacité du dispositif dans lequel ces données sont enregistrées:

- Le microprogramme spécifique au dispositif, l'UDID et les informations de fournisseur du dispositif (DVI) (voir 7.5.4.7) doivent être au minimum enregistrés en permanence dans le dispositif. Il ne doit y avoir dans le protocole FSCP 13/1 aucun moyen de modifier ces données via le réseau.
- Des SN simples, sans mémoire permanente, doivent recevoir le SDN, la SADR et tout autre paramètre en provenance d'une SCM après démarrage.

  NOTE 1 Ceci exige un enregistrement permanent des données spécifiques à ce SN dans le SCM.

- Des SN complexes à mémoire permanente ou à commutateur physique permettant le réglage du SDN et de l'adresse SADR, doivent conserver leurs données en permanence sur le dispositif.

  NOTE 2 Le SCM enregistrera uniquement les données qui ne sont pas enregistrées en permanence dans le SN; pour toutes les autres données, le SCM n'effectuera qu'une vérification.

6.1.3.3 Domaine de sécurité (SD)

6.1.3.3.1 Généralités

Un domaine de sécurité doit constituer un espace d'adresse contenant jusqu'à 1 023 SADR (1 à 1 023). Seuls les SN qui partagent le même SD doivent pouvoir communiquer directement entre eux. Les SN sur des SD différents doivent se limiter à une communication via une passerelle de domaine de sécurité.

Un SD est identifié par son numéro de domaine de sécurité (SDN) qui va de 1 à 1 023; 0 est réservé et ne doit pas être utilisé. Le SDN est défini manuellement. On doit s'assurer que des SDN uniques sont utilisés au moment où la configuration de l'application est planifiée.

6.1.3.3.2 Protection du domaine de sécurité

Le risque d'une attaque extérieure du FSCP 13/1 dans les données de processus de la PDU de sécurité (SPDO) est négligeable. Il y a cependant un risque potentiel pour les services de configuration. Lorsque l'Internet est utilisé en relation avec le FSCP 13/1, on doit utiliser une méthode de cryptage fiable, comme par exemple VPN (Virtual Private Network – Réseau privé virtuel), SSL (Secure Socket Layer – couche d'échanges sécurisés) ou IPsec (sécurité du protocole Internet). Le choix d'un système de cryptage approprié est du ressort de la planification du déploiement et ne s'inscrit pas dans le domaine d'application de la présente partie. Dans les réseaux locaux (LAN), on doit déployer des mesures organisationnelles telles que des pare-feux.

La Figure 11 donne un exemple de protection de domaine de sécurité FSCP 13/1.

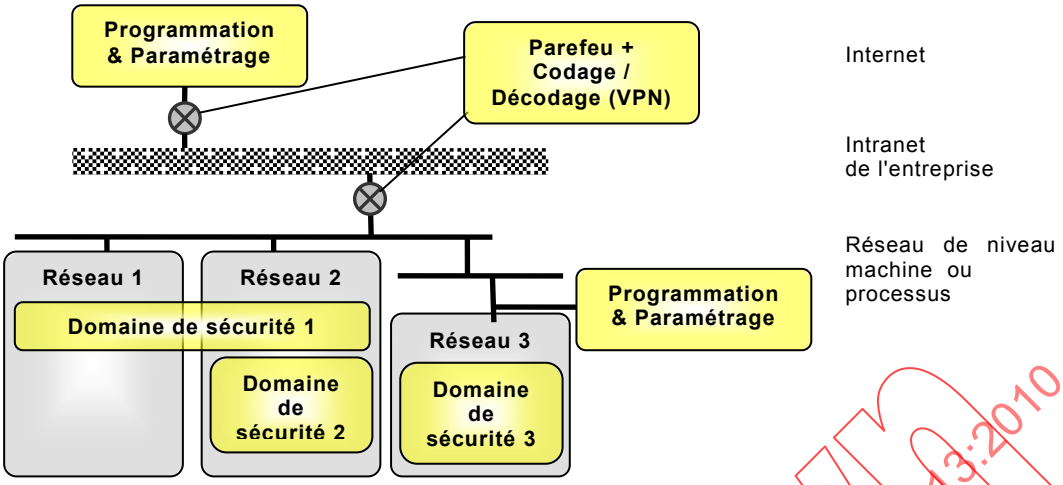


Figure 11 – Protection de domaine de sécurité (exemple)

6.1.3.3.3 Séparation entre domaines de sécurité

L'interaction entre domaines de sécurité ainsi que les risques de confusion au cours de la configuration et du paramétrage sont exclus grâce à l'utilisation de numéros de domaines de sécurité uniques.

NOTE Si la taille de l'Intranet de l'entreprise et l'organisation de la gestion du réseau ne permettent pas l'administration de numéros de domaines de sécurité uniques, il est recommandé de les diviser en plusieurs étendues de réseaux gérables et de les séparer par des méthodes approuvées telles que des pare-feux.

Pour éviter des situations critiques du point de vue de la sécurité, les données de SPDO et de PDO de SSDO doivent en outre être codées de manière à assurer leur unicité au sein d'un domaine de sécurité donné (voir 7.1.10). Par conséquent, des PDO de sécurité qui sont transmis par erreur dans le domaine incorrect peuvent être identifiés et rejetés.

La Figure 12 donne un exemple de séparation entre domaines de sécurité.

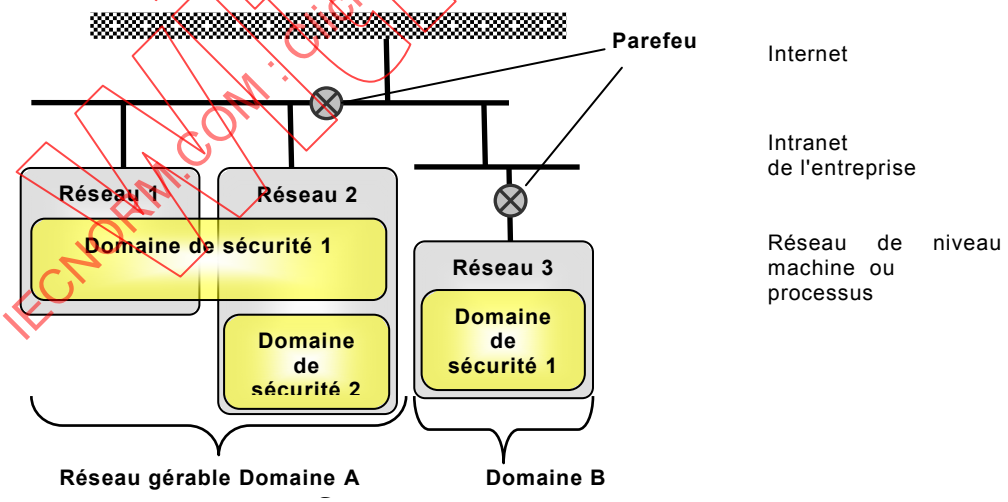


Figure 12 – Séparation entre domaines de sécurité (exemple)

6.1.3.4 Passerelle de domaine de sécurité (SDG)

Il doit exister dans le protocole FSCP 13/1 un rôle du dispositif spécial, capable de communiquer avec différents domaines de sécurité (SD). La communication entre domaines est traitée en interne par la SDG. Dans un SD donné, la SDG agit comme un SN normal.

Une SDG est chargée de:

— acheminer des PDU de SSDO entre domaines de sécurité si un SN souhaite envoyer un SSDO à un SN situé dans un autre domaine de sécurité; ce SSDO est envoyé à la SDG qui l'achemine au nœud de destination et il peut être nécessaire de passer par plusieurs SDG pour atteindre le SN de destination,

— acheminer des données SPDO entre des domaines de sécurité; la SDG peut disposer de moyens de lecture des données contenues dans les SPDO d'un domaine de sécurité donné et envoyer ces données dans un autre domaine de sécurité en les plaçant dans le SPDO de la SDG.

La configuration de la SDG dépend de chaque fournisseur particulier.

L'objectif de la présente partie est de spécifier les mécanismes de communication sécuritaires nécessaires à l'échange de données au sein d'un domaine de sécurité unique. Les communications entre différents domaines de sécurité (et la passerelle de domaine de sécurité en est un moyen) ne s'inscrivent pas dans le domaine d'application de la présente partie.

6.1.3.5 Gestionnaire de configuration (SCM)

Le Gestionnaire de configuration de sécurité (SCM) doit être au centre des responsabilités de gestion FSCP 13/1. Dans un domaine de sécurité, il ne doit être autorisé qu'un seul SN exécutant le service SCM.

Le SCM doit être chargé:

— de l'enregistrement permanent des paramètres spécifiques de SN qui ne sont pas mémorisés dans le SN (voir 6.1.3.2),

— de l'affectation unique et de la vérification de la SADR au sein d'un SD donné,

— de la vérification de l'unicité de l'UDID au sein du SD,

— de la vérification des paramètres prévus et des DVI des SN, et

— de l'envoi d'un signal de sauvegarde périodique à tous les SN dans le SD considéré.

NOTE Ce signal de sauvegarde périodique provenant du SCM est nécessaire dans le SN afin de détecter certaines situations d'échec comme lorsque aucune SCM n'est responsable du SN.

6.2 Modèle de cycle de vie

6.2.1 Généralités

Comme défini dans la CEI 61508, le protocole FSCP 13/1 prend en charge l'application sécuritaire au cours de diverses phases de son cycle de vie. La communication sécuritaire n'est qu'une partie de l'ensemble du concept de sécurité applicable aux machines. Les autres parties sécuritaires de la machine doivent bien entendu être également prises en compte. Le présent paragraphe s'intéresse uniquement à la communication sécuritaire dans le cadre du protocole FSCP 13/1.

6.2.2 Concept, planification et mise en œuvre

6.2.2.1 Agencement de l'application

Les caractéristiques du FSCP 13/1 doivent être prises en compte lorsqu'il s'agit de concevoir l'agencement de l'application. Les possibilités et restrictions suivantes doivent être respectées:

— Domaine de sécurité, attribution de SDN,

— Séparation et protection des domaines de sécurité,

— Adresses de sécurité,

— Gestionnaire de configuration de sécurité, et

— Communication entre les domaines.

6.2.2.2 Programmation et paramétrage

6.2.2.2.1 Généralités

La programmation et le paramétrage doivent être effectués au moyen d'outils sécuritaires. Les fichiers et paramètres de l'application doivent être mémorisés dans le dispositif SCM uniquement ou directement dans le dispositif correspondant en fonction de ses capacités.

La programmation et le paramétrage de l'application sécuritaire doivent comprendre les tâches suivantes:

— l'accès à des SN dans un SD donné est référencé uniquement par l'adresse SADR,

NOTE Ceci permet l'installation de plusieurs applications identiques avec des SDN différents sans avoir à modifier l'application.

— la communication avec d'autres SD n'est possible que par l'intermédiaire du SDG,

— localiser le service SCM,

— répertorier les DVI des dispositifs,

— déployer l'application sécuritaire, et

— spécifier les paramètres sécuritaires (voir 7.5).

Le SCM est le nœud central pour la vérification des dispositifs sécuritaires. Le FSCP 13/1 doit prendre en charge deux stratégies différentes de configuration. Ceci est nécessaire pour respecter toutes les exigences résultant de la large gamme d'applications qui peuvent utiliser le FSCP 13/1. Ces deux stratégies doivent être:

— Le mode de configuration automatique (ACM), et

— Le mode de configuration manuel (MCM).

6.2.2.2.2 Mode de configuration automatique (ACM)

Lorsqu'il utilise le mode de configuration automatique (ACM), le système regroupe automatiquement les UDID des SN connectés. L'ACM doit être utilisé uniquement si toutes les conditions suivantes sont remplies:

— tous les dispositifs sécuritaires peuvent être connectés au moment de la configuration,

— un dispositif (UDID unique) pour chaque SN ayant une SADR spécifique,

— si un dispositif sécuritaire est remplacé, le SCM vérifie que le nouveau dispositif est du même type que celui qui a été remplacé; si cette condition est remplie, le nouvel UDID sera recueilli et l'opérateur doit confirmer le remplacement.

6.2.2.2.3 Mode de configuration manuel (MCM)

Le mode de configuration manuel (MCM) s'applique à tous les types d'applications; cependant, ce mode nécessite une configuration manuelle des dispositifs sécuritaires. De ce fait, ce mode est principalement destiné aux applications présentant les caractéristiques suivantes:

— tous les dispositifs sécuritaires connectables ne peuvent pas être connectés en une seule fois,

EXEMPLE 1 Bras robotisé avec plusieurs outils sécuritaires.

— dispositifs multiples (multiples UDID) pour un SN ayant une adresse SADR spécifique,

EXEMPLE 2 Soit une application de machine modulaire avec plusieurs chaînes de machines dans une installation donnée. Le module A (équivalent à la SADR) de la machine peut être utilisé pour toutes les

chaînes de machines. Pour éviter les problèmes de goulots d'étranglement, il y a plusieurs équipements (c'est-à-dire plusieurs UDID) pour le module A (=SADR) et ces équipements (UDID) sont tous applicables sur une chaîne de machines donnée.

— le remplacement du module doit être réalisé automatiquement sans intervention manuelle.

6.2.3 Mise en service

6.2.3.1 Généralités

La mise en service peut être divisée en trois étapes principales:

a) installation de l'application (voir 6.2.3.2),

b) réglage de la configuration en fonction du mode de configuration (voir 6.2.3.3),

c) vérification de la fonctionnalité sécurité (voir 6.2.3.4).

6.2.3.2 Installation

Installer les applications et les paramètres sur les dispositifs FSCP 13/1. L'une des méthodes suivantes doit être utilisée.

— un outil de programmation conforme FSCP 13/1 utilisant les services FSCP 13/1 spécifiés (voir en 7.3), ou

— un protocole propriétaires de transmission sécuritaire pour l'échange de données du système de programmation vers un dispositif sécuritaire via le réseau ou une connexion de communication directe (par exemple une liaison série).

Au cours de la phase installation du cycle de vie, la capacité du dispositif doit déterminer quelles données doivent être téléchargées et où:

— si le dispositif lui-même détient les données, celles-ci doivent être téléchargées vers le dispositif,

— si le dispositif est incapable de détenir les données, celles-ci doivent être enregistrées sur le SCM. Le SCM doit assurer la mise à jour correcte de tous les dispositifs à la mise sous tension en utilisant les services de configuration FSCP 13/1.

Après installation des applications et des paramètres, le SDN doit être si nécessaire mis à jour (voir 6.1.3.3).

6.2.3.3 Réglage de la configuration

6.2.3.3.1 Réglage de la configuration en ACM

Le réglage de la configuration en ACM doit comprendre les quatre étapes suivantes:

a) mise sous tension de tous les dispositifs sécuritaires,

b) s'assurer que la couche de communication fonctionne pour tous les SN,

c) démarrage du SCM en rassemblant les UDID (sans intervention de l'opérateur),

d) le personnel chargé de la mise en service doit effectuer la reconnaissance de chaque dispositif.

6.2.3.3.2 Réglage de la configuration en MCM

Le réglage de la configuration en MCM doit comprendre les trois étapes suivantes:

a) le personnel chargé de la mise en service doit saisir manuellement la liste des SADR possibles ainsi que les UDID correspondants; pour cela il doit être utilisé un outil de configuration sécuritaire,

b) le SCM doit accepter toutes les combinaisons SADR / UDID configurées,

c) le SCM ne doit pas accepter de discordances et ne doit pas remédier aux éventuelles discordances comme cela a été effectué en ACM. En d'autres termes, toutes les discordances doivent être corrigées par le personnel chargé de la mise en service en saisissant la valeur corrigée au moyen de l'outil de configuration sécuritaire.

6.2.3.4 Vérification

La vérification doit comprendre les cinq étapes suivantes:

a) le personnel chargé de la mise en service doit vérifier la fonctionnalité de la sécurité par rapport à la spécification de sécurité de l'application. En cas de modification ou de remplacement de dispositifs sur le trajet au sein de la couche de communication, le personnel chargé de la mise en service doit passer à l'étape décrite en 6.2.3.3;

b) si la vérification est effectuée au cours de la mise en service, toutes les combinaisons de modules échangeables doivent être vérifiées;

NOTE Cela est notamment important dans le cas de machines modulaires complexes (en combinaison avec le MCM).

c) Si la vérification est effectuée dans le cadre d'une maintenance, seule la fonctionnalité sécuritaire affectée à la tâche de maintenance doit être vérifiée;

d) Lorsque la vérification est achevée, une sauvegarde spécifique à l'installation de la configuration en cours doit être effectuée;

e) En cas de remplacement d'un SCM, un téléchargement des fichiers d'application et des paramètres, ainsi que la configuration doivent être effectués; dans le cas contraire, les étapes de mise en service (installation, configuration et vérification) doivent être recommencées.

6.2.4 Conditions de fonctionnement

6.2.4.1 Transfert des données sécuritaires

Pendant le fonctionnement, le transfert des données sécuritaires est géré par le protocole FSCP 13/1. Le réseau sous-jacent est chargé du transport des données entre les nœuds, avec une qualité de temporisation définie. La perte de synchronisation ou de la qualité du transport dans la couche de communication sous-jacente est détectée par la couche FSCP 13/1 (voir 7.7.2) et entraînera une perte de la disponibilité et non une perte de sécurité (approche "canal noir").

EXEMPLE Voir la Figure 13.

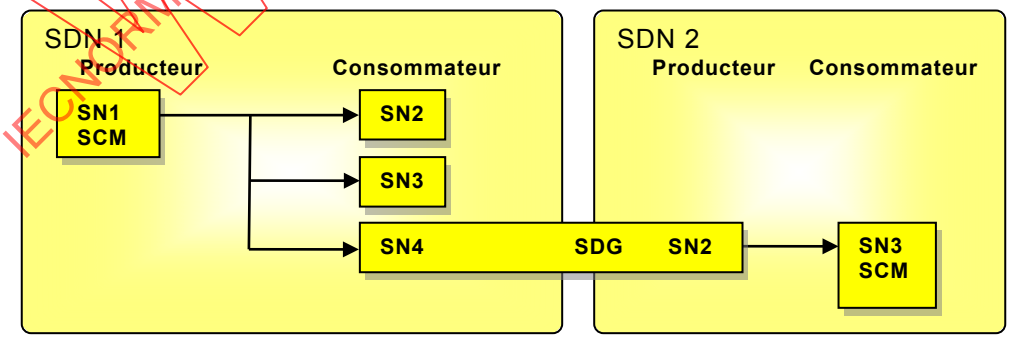


Figure 13 – Exemple de flux de données

6.2.4.2 Synchronisation temporelle et validation

Pour éviter l'absence ou les retards soudains de données, la vérification de la performance du réseau doit être effectuée. Chaque consommateur doit éventuellement envoyer une requête aux producteurs connectés pour leur demander leur temps relatif. En comparant les temps

relatifs, le temps écoulé et les informations temporelles d'éventuels transferts, le consommateur doit vérifier l'opportunité et l'actualité des données reçues.

La vérification des performances du réseau fait partie du transfert de données. En premier lieu, un consommateur envoie une PDU de “Demande de temps” au producteur connecté. Une fois cette PDU reçue, le producteur envoie ses données avec un code supplémentaire qui permet au consommateur d'identifier ce message comme étant une “Réponse de temps”. En mesurant le délai écoulé entre la “demande de temps” et la “réponse de temps”, le consommateur peut vérifier si le réseau est suffisamment rapide pour répondre aux exigences de l'application, en fonction du temps de réaction le plus défavorable.

Une description détaillée est fournie en 7.7.2.

6.2.4.3 Sauvegarde

Le service Sauvegarde (voir 7.5.4.5) doit permettre au FSCP 13/1 de s'assurer que seuls les SN configurés sont opérationnels sur un réseau FSCP 13/1. Un SN attend des signaux de sauvegarde fréquents pour rester à l'état opérationnel. Dans le cas contraire, il doit passer à l'état pré-opérationnel dans lequel il n'y a aucun risque pour le réseau sécuritaire.

La fréquence du signal de sauvegarde doit être spécifique à l'application et elle est en général déterminée par la durée nécessaire pour vérifier les fonctions de sécurité de l'application.

L’utilisation d’une mesure organisationnelle, comme par exemple la mise hors tension de tous les SN en cas de reconfiguration du SCM, permet d'éliminer tout risque dans une telle situation. En conséquence, la fréquence de sauvegarde peut être réglée à une valeur très importante (plusieurs jours) dans de tels cas.

6.2.4.4 Démarrage après mise sous tension ou réinitialisation

Après mise sous tension, le SCM (voir 7.7.9) doit effectuer une vérification du réseau. S'il n'y a aucune modification sur le réseau, aucune tâche particulière n'est requise.

6.2.4.5 Récupération suite à une défaillance du réseau

Suite à une panne du réseau, tous les SN concernés doivent passer à l'état pré-opérationnel. Étant donné que les SN consommateurs correspondants n'obtiendront plus de données, ces nœuds doivent passer à l'état de sécurité.

Après récupération du réseau, tous les SN concernés doivent envoyer au SCM une information SNMT_SN_in_PRE_OP. Le SCM doit ensuite démarrer avec l'initialisation des SN.

6.2.5 Conditions de maintenance

Voir 9.6.

6.3 Couche de communication non sécuritaire

6.3.1 Généralités

Le présent paragraphe énumère les exigences applicables à la couche de communication non sécuritaire pour le transport des PDU de sécurité. Les protocoles de communication conformes au CFP 13 (voir la CEI 61784-2, la CEI 61158-3-13, la CEI 61158-4-13, la CEI 61158-5-13 et la CEI 61158-6-13) satisfont à ces exigences.

6.3.2 Exigences pour le transport des données

6.3.2.1 Généralités

La couche de communication non sécuritaire est chargée du transport des données entre les SN. La coexistence de nœuds non sécuritaires et de nœuds sécuritaires est admise.

6.3.2.2 Déguisement

Pour éviter toute possibilité d'usurpation de données telles que des PDU de sécurité, les nœuds non sécuritaires ne doivent pas:

— comporter un codage tel que décrit à l'Article 7,

— comporter un codage capable de générer des PDU de sécurité selon l'Article 7, ou

— appliquer d'éventuels mécanismes tels que définis à l'Article 7.

Des informations spécifiques concernant le traitement de ces problèmes par les mécanismes du protocole figurent à l'Article 7.

La couche de communication non sécuritaire est autorisée à avoir un accès en lecture seule pour les données dans une PDU de sécurité. Ceci doit être réalisé en permettant l'accès aux données dans les PDU de sécurité sans utiliser les mesures d'intégrité de données spécifiques aux PDU de sécurité, comme par exemple le CRC et la répétition de données de la partie un de la PDU dans la partie deux de la PDU.

6.3.2.3 Modèle de communication

Du point de vue de la CPF 13, la Figure 14 décrit le modèle de communication de sécurité commun des CN pour 0 n à SN.

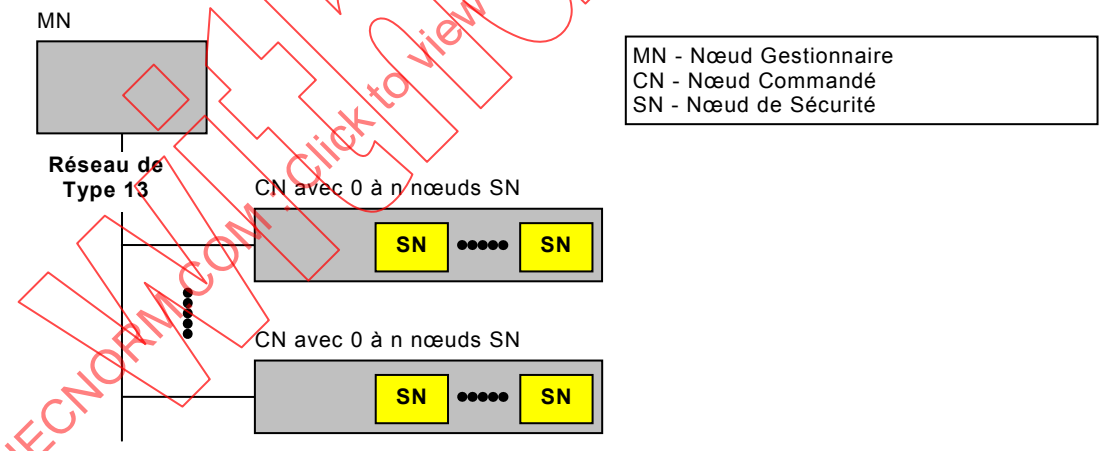


Figure 14 – Modèle de communication

Les fonctionnalités MN et CN sont telles que définies dans la CEI 61158-3-13; elles ne sont impliquées dans aucune fonction de sécurité.

La fonction de sécurité est mise en œuvre par une entité au sein du CN appelée nœud de sécurité (SN). Différentes situations peuvent se présenter:

— CN sans un SN,

— CN avec un SN uniquement,

— CN avec plusieurs dispositifs sécuritaires représentés comme un seul SN. La sécurité du bus du module et des modules supplémentaires doit être assurée par le fournisseur,

— CN à multiples SN, ou

— CN en tant que système modulaire et à multiples SN. La sécurité du bus de connexion du module est assurée par le FSCP 13/1.

6.3.2.4 Transport de SPDO

La couche de communication non sécuritaire doit assurer le transport des données SPDO entre les SN. Les services définis dans la CEI 61158-5-13 doivent être utilisés.

NOTE 1 Les données utiles SPDO sont utilisées pour les données de processus cycliques, comme l'état d'une entrée ou d'une sortie. Les SPDO sont retransmis de manière cyclique au cours d'une certaine période. L'envoi d'un SPDO est uniquement déclenché par un élément temporel et non lié à une modification des données proprement dites.

Les données SPDO doivent être transmises comme des PDU de sécurité dans des PDU de couche d'application de Type 13 (APDU). Les APDU Isoc2 conformes à la CEI 61158-6-13 doivent être utilisées. Un SPDO est généré de manière cyclique par un SN, appelé producteur; ce SPDO peut avoir un ou plusieurs SN cible(s), appelé(s) consommateur(s). Pour le transport d'un SPDO de son producteur à ses consommateurs, des mécanismes de Type 13 doivent être utilisés.

Le principe de base du transport de SPDO est illustré en Figure 15.

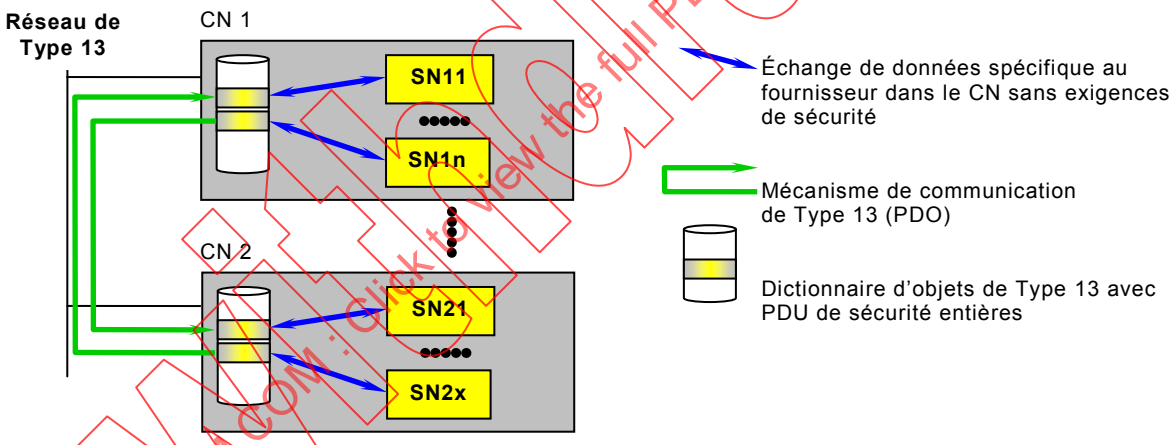


Figure 15 – Transport de SPDO

Le trajet de transmission entre le SN producteur et le SN consommateur est défini de la manière suivante:

— le SN producteur construit une SPDU,

— dans le CN auquel le SN producteur est connecté (CN producteur), une entrée de dictionnaire d'objets de domaine type de données doit être définie pour incorporer la SPDU (dans le trajet de communication, la SPDU ne doit pas être fragmentée),

— le déplacement de données SPDU vers l'entrée du dictionnaire d'objets incombe au CN producteur,

— l'entrée de dictionnaire d'objets doit alors être transmise dans une PDU de couche d'application de Type 13 (APDU), qui est transmise de manière cyclique en utilisant l'ASE d'objet de données de processus (ASE de PDO) comme spécifié dans la CEI 61158-5-13,

— le dictionnaire d'objets du CN de réception (CN consommateur) doit définir un objet de domaine type de données correspondant à la définition de l'entrée du dictionnaire d'objets du producteur. Le CN consommateur doit recevoir l'APDU et copier la PDU de sécurité qui y est contenue vers l'objet domaine mentionné auparavant, et

— au cours de la dernière étape, le CN consommateur est chargé de déplacer les données SPDU de l'entrée du dictionnaire d'objets vers le SN consommateur.

Les mécanismes impliqués dans la transmission du SPDO entre le dictionnaire d'objets de CN et les SN (décrits par des flèches bleues dans la Figure 15) ne s'inscrivent pas dans le domaine d'application de la présente partie et peuvent être spécifiques au fournisseur.

NOTE 2 Du point de vue des services et du protocole de communication de Type 13, il n'est pas nécessaire de faire la distinction entre la communication de SPDO et d'autres données. Il n'y aucun traitement particulier des données de sécurité par des mécanismes de Type 13. Ceci correspond à l'approche "canal noir" qui est appliquée.

NOTE 3 Aucune connaissance concernant le trajet de communication de Type 13 n'est nécessaire aux SN impliqués dans la communication.

6.3.2.5 Transport de SSDO

La couche de communication non sécuritaire doit assurer le transport des données SSDO entre les SN. Les services définis dans la CEI 61158-5-13 doivent être utilisés.

Le principe de base du transport de SSDO est illustré en Figure 16.

NOTE 1 Les données utiles SPDO sont utilisées pour les communications de données spontanées, telles que la configuration et le paramétrage.

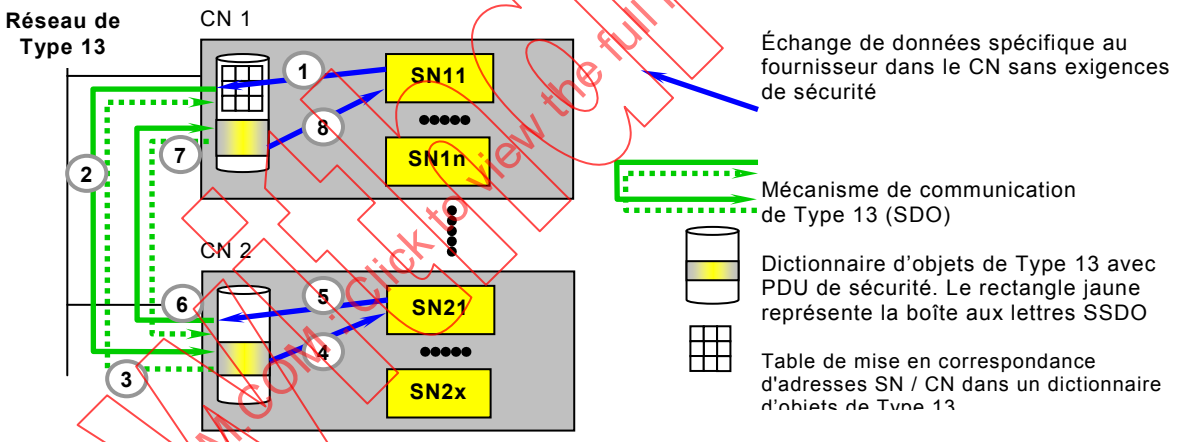


Figure 16 – Transport de SSDO

Les données SSDO doivent être transmises comme des PDU de sécurité dans des PDU de couche d'application de Type 13 (APDU). Les APDU Asyn2 avec SDO de type pduBody conformes à la CEI 61158-6-13 doivent être utilisées.

Un SSDO est généré de manière spontanée par un SN (le SN d'origine) et il est transmis à précisément un SN cible. Pour transporter un SSDO de son origine à sa destination, il doit être utilisé des mécanismes de Type 13.

L'échange de SSDO entre SN est réalisé au moyen de services et de protocoles de Type 13 associés à un mécanisme de messagerie (les chiffres dans les parenthèses font référence à la Figure 16):

(1) Le SN d'origine génère une demande SSDO à un SN de destination spécifique et construit une SPDU correspondante,

(2) le CN desservant le SN d'origine crée un accès en écriture de SDO vers la boîte aux lettres du CN desservant le SN de destination; la SPDU de l'étape (1) est écrite dans le dictionnaire d'objets du CN cible. Ceci nécessite une table de correspondance entre les adresses SN et CN dans le CN desservant le SN d'origine afin d'obtenir l'ID du nœud de Type 13 du CN desservant le SN de destination SN (CN cible); il incombe à la configuration système de Type 13 de fournir la table de correspondance des adresses,

la boîte aux lettres doit être une entrée de dictionnaire d'objets de Type 13 du domaine type de données qu'il convient de définir pour pouvoir intégrer une SPDU,

(3) l'accès en écriture SDO est confirmé par le protocole SDO et la connexion est maintenue ouverte,

(4) le CN de destination redirige les données (SPDU) dans la boîte aux lettres vers le SN de destination et pour cela, le CN de destination doit lire les informations d'adresse dans la SPDU,

(5) le SN de destination génère une réponse SSDO et construit une SPDU correspondante,

(6) le CN de destination crée un accès en écriture SDO vers la boîte aux lettres du CN d'origine et écrit la SPDU de réponse dans le dictionnaire d'objets du CN d'origine,

(7) l'accès en écriture de SDO est confirmé, la connexion est fermée, et

(8) le CN d'origine redirige les données (SPDU) dans la boîte aux lettres du SN de destination et pour cela le CN d'origine doit lire les informations d'adresse contenues dans la SPDU.

Les mécanismes impliqués dans la transmission du SPDO entre le dictionnaire d'objets de CN et les SN (décrits par des flèches bleues dans la Figure 16) ne s'inscrivent pas dans le domaine d'application de la présente partie et peuvent être spécifiques au fournisseur.

NOTE 2 Du point de vue des services et du protocole de communication de Type 13, il n'est pas nécessaire de faire la distinction entre la communication de SSDO et d'autres données. Il n'y aucun traitement particulier des données de sécurité par des mécanismes de Type 13. Ceci correspond à l'approche "canal noir" qui est appliquée.

NOTE 3 Aucune connaissance concernant le trajet de communication de Type 13 n'est nécessaire aux SN impliqués dans la communication.

6.3.2.6 Représentation des données de diagnostic

La couche de communication non sécuritaire doit prendre en charge des services fournissant des données de diagnostic spécifiques au FSCP 13/1 conformément à l'Article 7.

Le principe de base de la représentation des données de diagnostic est illustré en Figure 17.

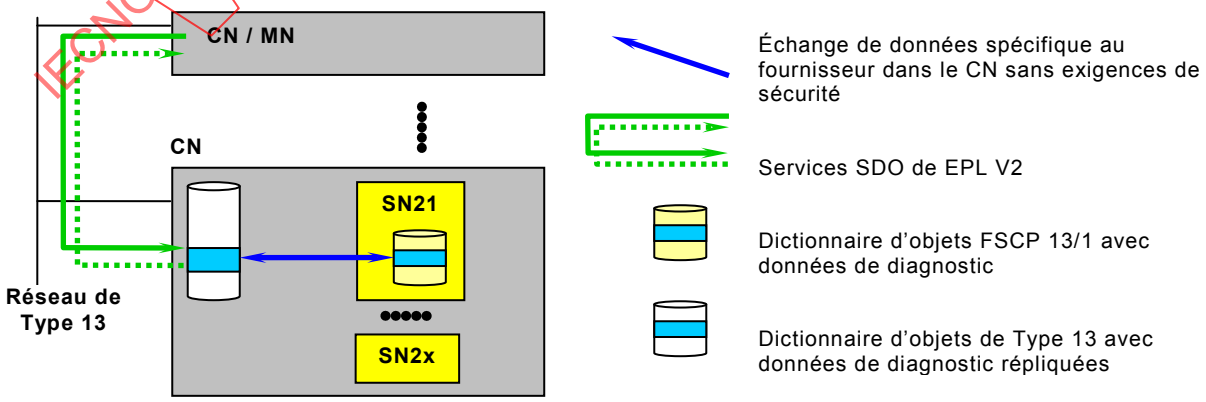


Figure 17 – Représentation des données de diagnostic

Les données de diagnostic dans un SN sont stockées dans le SOD (voir l'Article 7).

NOTE En général, ces données de diagnostic sont demandées à des nœuds particuliers (par exemple une unité de commande centralisée, une interface homme machine, des équipements de service) pour traiter les données de diagnostic et en rendre compte à l'utilisateur final.

Le CN doit répliquer les données de diagnostic du SOD dans le dictionnaire d'objets de Type 13 du CN et on peut alors y accéder au moyen de services SDO de Type 13.

6.3.3 Protection et séparation des domaines

Conformément au 6.1.3.3.2 et au 6.1.3.3.3, la couche de communication doit prendre en charge des fonctions de sécurité spécifiques afin de protéger le domaine contre toute attaque de l'extérieur et éviter les interférences entre des domaines séparés.

7 Protocole de couche de communication de sécurité

7.1 Format de PDU de sécurité

7.1.1 Généralités

Une PDU de sécurité est transférée dans la partie « données » d'une APDU de Type 13. Une APDU de Type 13 peut contenir de zéro à plusieurs PDU de sécurité, chacune reflétant une relation producteur / consommateur entre SN (voir la Figure 18). Les données sécuritaires et non sécuritaires peuvent être mélangées dans une APDU de Type 13.

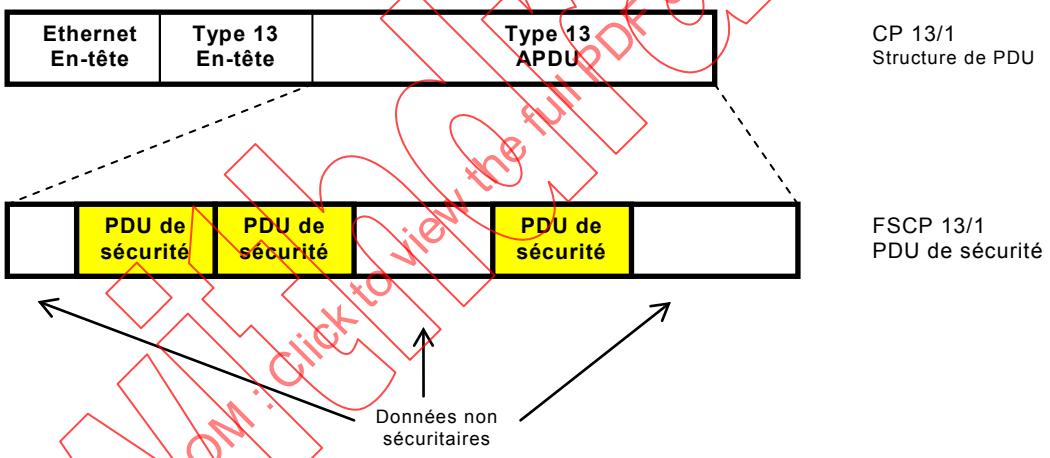


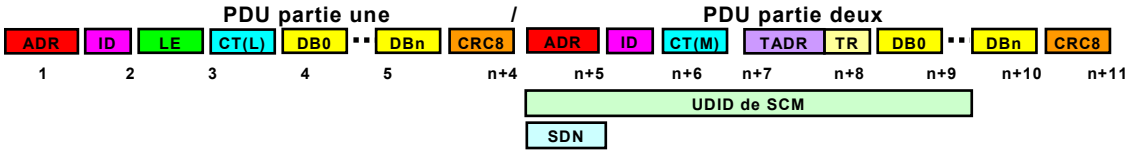
Figure 18 – PDU de sécurité dans une PDU CP 13/1

La structure de la PDU de sécurité doit être telle qu'illustrée en Figure 19 et en Figure 20. La PDU de sécurité doit être constituée de deux sous-parties et pouvoir transporter jusqu'à 254 octets de données utiles. La probabilité d'erreurs et la distance de Hamming, eu égard à la longueur de la PDU de sécurité, sont discutées en A.3 et en A.4.

En fonction de la longueur des données utiles, deux types de PDU de sécurité doivent être utilisées:

— 1 à 8 octets de données utiles: il doit être utilisé un format de PDU comme illustré en Figure 19 (avec un CRC sur 8 bits);

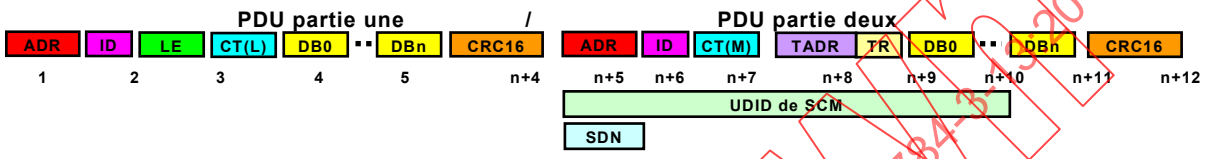
— 9 à 254 octets de données utiles: il doit être utilisé un format de PDU comme illustré en Figure 20 (avec un CRC sur 16 bits).



Légende

n nombre d'octets de charge utile

Figure 19 – PDU de sécurité pour n = 0 à 8 octets de données utiles



Légende

n nombre d'octets de charge utile

Figure 20 – PDU de sécurité pour n = 9 à 254 octets de données utiles

L'application de la présente partie, conformément aux exigences définies dans la CEI 61784-3:2010, 5.6.1 (calcul du taux d'erreurs résiduelles) entraîne une limitation des octets de charge utile à une plage de 9 à 25 octets.

NOTE 1 La CEI 61784-3:2010 précise en 5.6.1 "à moins qu'il ne puisse être démontré une meilleure probabilité d'erreurs, on doit utiliser une valeur de 10^{-2}". Pour les calculs du taux d'erreurs résiduelles, dans le cas où la probabilité d'erreur est de 10^{-2}, voir A.3.

NOTE 2 Les conditions préalables à remplir pour justifier un taux d'erreurs sur les bits de 10^{-3} et les résultats des calculs de taux d'erreurs résiduelles dans ce cas sont présentés en A.4.

Le Tableau 4 décrit la PDU de sécurité. Elle doit être constituée de deux parties appelées partie une de la PDU et partie deux de la PDU. Les données de la partie deux de la PDU de télégrammes SPDO et SSDO (et non de télégrammes SNMT) sont en outre codées avec un UDID du SCM en utilisant une opération logique XOR - OU exclusif (voir 7.1.10).

Tableau 4 – Format de PDU

Décalage d'octet [a]	Décalage binaire							
	7	6	5	4	3	2	1	0
0	ADR (Bit 0-7), voir 7.1.2							
1	ID, voir 7.1.3						ADR (8,9)	
2	LE, voir 7.1.4							
3	CT (Bit 0-7) , voir 7.1.5							
4 -- n+3	DB 0 à DB n, voir 7.1.6							
n+4 -- n+4+o	CRC-8 / CRC-16, voir 7.1.7							
n+5+o	ADR (Bit 0-7) XOR SDN (Bit 0-7)							
n+6+o	ID						n+6+o	
n+7+o	CT (Bit 8-15)							
n+8+o	TADR (Bit 0-7), voir 7.1.8							
n+9+o	TR, voir 7.1.9						n+9+o	

Décalage d'octet [a]	Décalage binaire							
	7	6	5	4	3	2	1	0
n+10+o -- 2n+9+o	DB 0 -- DB n							
2n+10+o -- 2n+10+2o	CRC-8 / CRC-16							
NOTE Les lignes en gris indiquent les données de la partie deux de la PDU.								
[a] Par rapport au début de la PDU de sécurité, n doit être le nombre d'octets de données utiles dans le champ DB 0 à DB n, où 0 ≤ n ≤ 254. o doit être le décalage de correction du CRC si 0 ≤ n ≤ 8, dans ce cas o = 0 (CRC 8); si 9 ≤ n ≤ 254, dans ce cas o = 1 (CRC 16).								

Les champs de la PDU de sécurité sont en outre définis dans les paragraphes suivants.

7.1.2 Champ d'adresse (ADR)

Le champ ADR définit l'adresse de Sécurité (SADR) du nœud de sécurité (SN). Le champ doit être une valeur de 10 bits. Le champ ADR doit accepter une plage de valeurs de 1 à 1 023 (0 est réservé et ne doit pas être utilisé).

Le champ ADR dans la partie deux de la PDU doit également être utilisé pour transporter les informations du numéro de domaine de sécurité (SDN) en utilisant une opération logique XOR - OU exclusif (voir 7.1.10).

Le champ ADR doit contenir la SADR du SN d'où la PDU est envoyée (adresse d'origine) ou à laquelle la PDU est destinée (adresse de destination). L'adresse d'origine doit être la SADR du SN qui envoie la PDU; l'adresse de destination doit être la SADR du SN auquel le télégramme est envoyé.

En fonction du contenu des champs ADR des deux parties de la PDU, la couche de communication non sécuritaire est capable de transporter les données au nœud de destination désigné sans analyse supplémentaire du champ ID ou d'autres attributs de PDU.

7.1.3 Champ d'identification de PDU (ID)

Le champ ID doit identifier le type de PDU. Le codage particulier du champ ID est donné dans le Tableau 5. Il est admis d'utiliser des codes réservés dans de futures éditions de la présente partie. Actuellement, ils ne doivent pas être utilisés et doivent être ignorés s'ils sont reçus. Pour identification du type de PDU, on doit utiliser les bits 5, 6 et 7. Chaque type de PDU dispose de types de télégrammes différents qui doivent être spécifiés par les bits 2, 3 et 4 du champ ID.

Tableau 5 – Champ d'identification de PDU (ID)

Décalage binaire								Type de PDU
7	6	5	4	3	2	1	0	
0	0	0	0	0	X [a]	bit 9 du champ d'adresse	bit 8 du champ d'adresse	Non autorisé (PDU illicite)
0	x	x	x	x	x			réservé
1	0	0	x	x	x			réservé
1	0	1	x	x	x			SNMT
1	1	0	x	x	x			SPDO
1	1	1	x	x	x			SSDO
[a] Un x dans ce tableau indique le type de télégramme pour un type de PDU spécifique.								

Le Tableau 6 donne toutes les combinaisons de champs ID valides.

Tableau 6 – Combinaisons de champs ID utilisées

Décalage binaire								Type de télégramme
7	6	5	4	3	2	1	0	
1	0	1	0	0	0	bit 9 du champ d'adresse	bit 8 du champ d'adresse	SNMT_Request_UDID
1	0	1	0	0	1			SNMT_Response_UDID
1	0	1	0	1	0			SNMT_Assign_SADR
1	0	1	0	1	1			SNMT_SADR_assigned
1	0	1	1	0	0			SNMT_Service_Request
1	0	1	1	0	1			SNMT_Service_Response
1	0	1	1	1	1			SNMT_SN_reset_guarding_SCM
1	1	0	0	0	0			SPDO_Data_Only
1	1	0	0	1	0			SPDO_Data_with_Time_Request
1	1	0	1	0	0			SPDO_Data_with_Time_Response
1	1	1	0	0	0			SSDO_Service_Request
1	1	1	0	0	1			SSDO_Service_Response

Les PDU SSDO et SNMT doivent être divisées en télégrammes de demande et de réponse, comme présenté dans le Tableau 7. Le bit 2 du champ ID doit être utilisé pour identifier le télégramme: demande ou réponse. Les bits 0 et 1 de l'octet où se trouve le champ ID sont utilisés par le champ d'adresse (voir Tableau 4).

Tableau 7 – Identifiant de demande / réponse

Décalage binaire								Description
7	6	5	4	3	2	1	0	
x	x	x	x	x	0			Demande SSDO / SNMT
x	x	x	x	x	1			Réponse SSDO / SNMT

7.1.4 Champ de longueur (LE)

Le champ LE indique le nombre d'octets de données utiles dans la PDU. La plage de valeurs valides doit être de 0 à 254. La valeur de LE doit déterminer le type de CRC utilisé pour la PDU comme spécifié dans le Tableau 8.

Tableau 8 – Type de CRC en fonction de LE

Valeur de LE	En-tête 2
0 à 8	CRC-8
9 à 254	CRC-16
255	réservé

7.1.5 Champ de Temps consécutifs (CT)

Le champ Temps consécutifs (CT) doit être divisé en une partie LSB envoyée dans la partie une de la PDU et une partie MSB envoyée dans la partie deux de la PDU. Les deux octets constituent le jeu de données CT dans une plage de 0 à 65 535. Dans le cas d'un SPDO, lorsque la PDU est envoyée, le champ CT doit être utilisé pour la valeur du temporisateur interne du SN (voir 7.2); dans le cas d'un SSDO, le champ CT décrit le numéro de demande d'accès SOD (voir 7.3). La base de temps du champ CT doit être définie dans un objet SOD 1200h de sous-index 03h (voir 7.5.4.11).

7.1.6 Champ de données utiles (DB0 à DBn)

Ce champ doit contenir les données utiles. Des données utiles d'une longueur de 254 octets doivent être prises en charge.

7.1.7 Champ de contrôle de redondance cyclique (CRC-8 / CRC-16)

Le champ CRC doit contenir la valeur du contrôle de redondance cyclique pour la partie de PDU à laquelle il appartient. Le type (CRC-8 ou CRC-16) dépend de la valeur de LE.

NOTE 1 La probabilité d'erreurs résiduelles ainsi que la distance de Hamming prévue sont décrites en A.3 et en A.4.

Le polynôme qui doit être utilisé pour CRC-8 est le suivant: $X^8+X^5+X^3+X^2+X+1$

Le polynôme qui doit être utilisé pour CRC-16 est le suivant: $X^{16}+X^{14}+X^{12}+X^{11}+X^8+X^5+X^4+X^2+1$

Les règles suivantes doivent s'appliquer à la génération du CRC:

1) La valeur de la graine de départ est 0 pour le CRC-8 et pour le CRC-16;
2) Le bit de poids fort du premier octet est décalé en premier;
3) Le CRC doit être calculé sur tous les champs de la partie de PDU, sauf le CRC;
   i) Partie une de la PDU: ADR, ID, LE, CT(L), DB0 à DBn;
   ii) Partie deux de la PDU: ADR, ID, CT(H), TADR, TR, DB0 à DBn.

Le CRC-8 doit satisfaire aux règles de codage d'un type de données UNSIGNED8 (NONSIGNE8).

Le CRC-16 doit satisfaire aux règles de codage d'un type de données UNSIGNED16 (NONSIGNE16).

NOTE 2 L'utilisation du CRC8 est justifiée uniquement si des conditions préalables particulières concernant le taux d'erreurs sur les bits sont satisfaites (voir A.4).

7.1.8 Champ d'adresse de demande de temps (TADR)

Le champ TADR spécifie:

— l'adresse de sécurité (SADR) d'un SN qui répond à la Demande de synchronisation temporelle dans le cas de SPDO (voir 7.2), ou

— la SADR (d'origine ou de destination) dans le cas de SSDO (voir 7.3) et de SNMT (voir 7.4).

7.1.9 Champ de numéro distinctif de demande de temps (TR)

Le champ TR doit être un nombre dans une plage de 0 à 63, qui doit être écrit en miroir par le dispositif afin de distinguer ce message d'autres télégrammes de demande de temps.

7.1.10 UDID de codage SCM (UDID de SCM)

L'UDID de 6 octets de SCM doit être codé par une opération OU exclusif au niveau du bit avec les six premiers octets des données utiles de la partie deux de la PDU des télégrammes SPDO et SSDO. Les télégrammes SNMT ne doivent pas être codés de cette manière.

L'UDID du SCM est connu pour chaque dispositif (voir 7.4.3.12); le SN doit vérifier chaque télégramme SPDO et SSDO reçu afin de s'assurer que l'UDID correct a été utilisé pour le codage des télégrammes.

NOTE Il est ainsi possible de détecter les éventuelles discordances de séparation de domaines (voir 6.1.3.3.3).

7.2 Objets de données de processus de sécurité (SPDO)

7.2.1 Généralités

Les objets de données de processus de sécurité (SPDO) doivent être utilisés pour la transmission de données de processus critiques en matière de sécurité, ainsi que pour la transmission de données de demande/réponse de synchronisation temporelle provenant d'un producteur SPDO et destinées aux consommateurs SPDO. La réception d'un SPDO par un consommateur SPDO doit être déterminée par l'adresse du producteur SPDO.

7.2.2 Types de télégrammes SPDO

Le champ ID (bit 2 à 4) doit identifier le type de télégramme SDPO. Trois télégrammes types doivent être définis:

— télégramme de données uniquement
— données avec télégramme de demande de temps, et
— données avec télégramme de réponse de temps

Le codage particulier du champ ID est donné dans le Tableau 9.

Tableau 9 – Types de télégrammes SPDO (champ ID, bits 2, 3 et 4)

Numéro de bit du champ d'identification								Type de PDU
7	6	5	4	3	2	1	0	
1	1	0	0	0	0	bit 8,9 du champ d'adresse		SPDO_Data_Only
1	1	0	x	x	1			Réservé
1	1	0	0	1	0			SPDO_Data_with_Time_Request
1	1	0	1	0	0			SPDO_Data_with_Time_Response
1	1	0	1	1	0			Réservé

7.2.3 Télégramme de données uniquement

Le télégramme de données uniquement (voir la Figure 21) doit être utilisé pour transmettre des données de processus d'un SN (producteur SPDO) sans une demande ou une réponse de synchronisation temporelle. Les données peuvent être consommées par tout autre SN (consommateur de SPDO).

En même temps que les données (Data), le producteur SPDO doit fournir son adresse SN (ADR) ainsi que le nombre d'octets de données utiles transmis (LE) et la valeur courante de son temporisateur interne (CT).

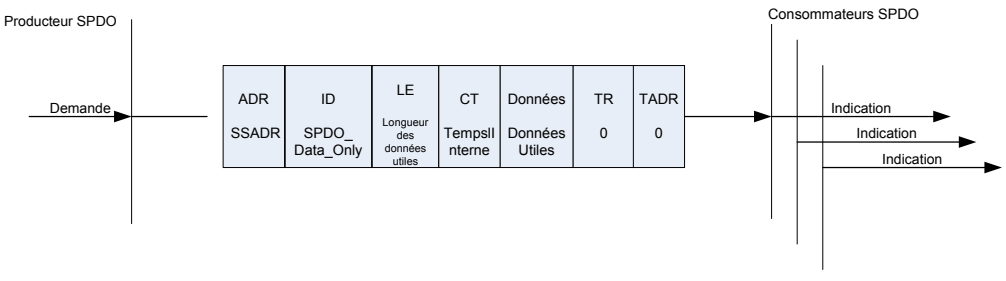


Figure 21 – Télégramme SPDO_Data_Only

Le Tableau 10 définit les champs d'un télégramme SPDO_Data_Only.

Tableau 10 – Champs du télégramme SPDO_Data_Only

Champ	Information	Contenu / Valeur
ADR	Adresse du producteur de SPDO	SSADR
ID	SPDO_Data_Only	$110000xx_b$
LE	Nombre d'octets de données utiles	0 à 254
CT	Valeur de temps interne du producteur de SPDO	Temps
Données	Données utiles	Données
TR	Non utilisé	0
TADR	Non utilisé	0

7.2.4 Données avec télégramme de demande de temps

Les données avec télégramme de demande de temps (voir la Figure 22) doivent être utilisées pour transmettre des données de processus d'un SN (producteur SPDO) avec une demande de synchronisation temporelle à un autre SN indiqué par TADR. Les données peuvent être consommées par tout autre SN (consommateur de SPDO).

En même temps que les données (Data), le producteur SPDO fournit son adresse SN (ADR), le type de télégramme (ID) ainsi que le nombre d'octets de données utiles (LE) transmis, la valeur courante de son temporisateur interne (CT), une valeur de compteur indiquant le nombre de demandes de temps (TR) et l'adresse du SN auquel il est demandé de fournir son temps interne courant (TADR).

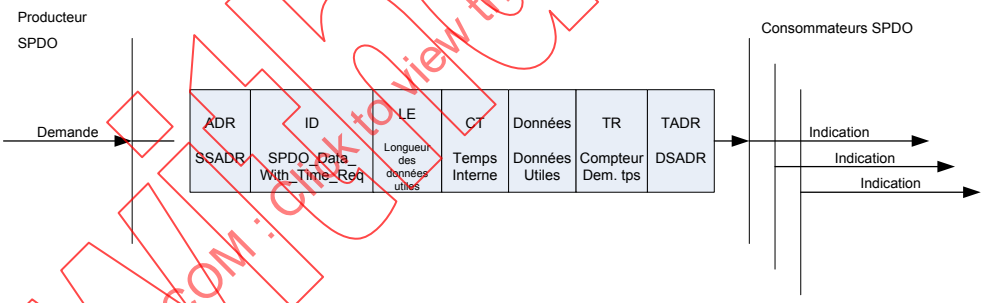


Figure 22 – Télégramme SPDO_Data_with_Time_Request

Le Tableau 11 définit les champs d'un télégramme SPDO_Data_with_Time_Request.

Tableau 11 – Champs de télégramme SPDO_Data_with_Time_Request

Champ	Information	Contenu / Valeur
ADR	Adresse du producteur de SPDO	SSADR
ID	SPDO_Data_with_Time_Request	110010xxb
LE	Nombre d'octets de données utiles	0 à 254
CT	Valeur de temps interne de SN	Temps
Données	Données utiles	Données
TR	Compteur de demande de temps interne	0 à 63
TADR	Adresse du SN dont les informations temporelles sont demandées	DSADR

7.2.5 Données avec télégramme de réponse de temps

Les données avec télégramme de réponse de temps (voir la Figure 23) doivent être utilisées pour transmettre des données de processus d'un SN (producteur SPDO) en même temps que la valeur du temporisateur interne du nœud concerné. Les données peuvent être consommées par tout autre SN (consommateur de SPDO).

En même temps que les données (Data), le producteur SPDO fournit son adresse SN (ADR), le type de télégramme (ID) ainsi que le nombre d'octets de données utiles (LE) transmis, la valeur courante de son temporisateur interne (CT), la valeur du compteur indiquant le nombre de demandes de temps (TR) et l'adresse du SN qui a émis la demande (TADR).

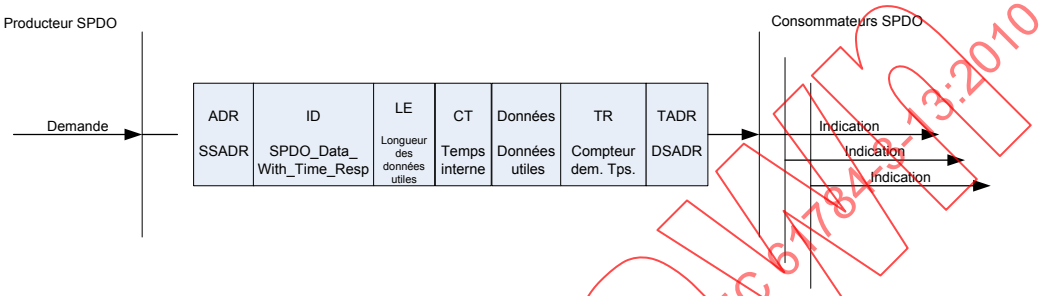


Figure 23 – Télégramme SPDO_Data_with_Time_Response

Le Tableau 12 définit les champs d'un télégramme SPDO_Data_with_Time_Response.

Tableau 12 – Champs de télégramme SPDO_Data_with_Time_Response

Champ	Information	Contenu / Valeur
ADR	Adresse du producteur de SPDO	SSADR
ID	SPDO_Data_with_Time_Response	110100xxb
LE	Nombre d'octets de données utiles	0 à 254
CT	Valeur de temps interne du producteur de SPDO	Temps
Données	Données utiles	Données
TR	Valeur du compteur de demande de temps pour demande correspondante	0 à 63
TADR	Adresse du SN qui a émis la demande d'informations temporelles	DSADR

7.3 Objet de données de service de sécurité (SSDO)

7.3.1 Généralités

Les objets de données de service de sécurité (SSDO) doivent être utilisés pour l'accès au dictionnaire d'objets (SOD) d'un serveur SSDO par un client SSDO. Les télégrammes SSDO doivent être identifiés par les bits 7, 6 et 5 (111b) de l'ID. Les bits 3 et 4 de l'ID doivent identifier un accès SOD (00b). Le bit 2 doit identifier les télégrammes de demande (0b) et de réponse (1b).

7.3.2 Types de télégrammes SSDO

Le champ ID identifie le télégramme. Les types de télégrammes suivants sont spécifiés:

— Demande de service SSDO

— Réponse de service SSDO

Le codage particulier du champ ID est donné dans le Tableau 13.

Tableau 13 – Types de télégrammes SSDO (champ ID, bits 2, 3 et 4)

Numéro de bit du champ d'identification								Type de télégramme
7	6	5	4	3	2	1	0	
1	1	1	0	0	0			SSDO_Service_Request
1	1	1	0	0	1			SSDO_Service_Response

Le télégramme de demande de service SSDO (SSDO_Service_Request) doit être utilisé par un client SSDO pour accéder au SOD d'un serveur SSDO. Avec une demande SSDO_Service_Request, un client SSDO doit fournir l'adresse SN du serveur (ADR), la longueur des données utiles (LE), le numéro de la demande d'accès SANo (champ CT). Il doit également indiquer dans le premier octet de données de l'"Octet Commande" (SACmd, voir le Tableau 14) le type d'accès (lecture/écriture), le type de transfert (accéléré/segmenté), le bit de bascule et "initialiser" ou dernier segment du transfert. L'index de l'entrée du dictionnaire d'objets à laquelle il est accédé doit être spécifié dans les octets de données 1 et 2; le sous-index de l'entrée à laquelle il est accédé doit être spécifié dans l'octet de données 3. L'adresse SN du client SSDO doit être fournie dans TADR.

Le télégramme de réponse de service SSDO (SSDO_Service_Response) doit être utilisé par un serveur SSDO pour répondre à une demande d'un client SSDO. Le numéro de demande d'accès SOD (SANo) doit être unique pour chaque connexion et être incrémenté de un (1) à chaque nouveau télégramme de demande d'accès SOD. Le télégramme de réponse doit contenir la valeur SANo de la demande de service correspondante.

Le nombre maximal de données utiles doit être de 250 octets.

Le codage des bits de la Commande d'accès SOD (SACmd) est spécifié dans le Tableau 14.

Tableau 14 – Codage binaire de la Commande d'accès SOD (SACmd)

Numéro de bit	Nom	Valeur	Signification
0	Type d'accès	0	Accès au SOD en lecture
		1	Accès au SOD en écriture
1	Réservé 1	0	Réservé
2	Abandonner le transfert	0	Transmission réussie
		1	Abandonner le transfert
3	Segmentation	0	Accès accéléré au SOD
		1	Accès segmenté au SOD
4	Bascule	0	Ce bit doit changer pour chaque segment suivant et ne doit pas changer de valeur s'il s'agit d'une répétition de la transmission. Le bit de bascule du premier segment doit être mis à 0. Le bit de bascule doit être identique pour les télégrammes de demande et de réponse.
		1	
5	Lancer le transfert	0	Ne pas lancer le transfert
		1	Lancer le transfert
6	Transfert de segment de fin	0	Il reste des segments à transférer
		1	Il ne reste plus de segments à transférer
7	Réservé 2	0	Réservé

7.3.3 Services et protocoles SSDO

Les services SSDO suivants doivent être définis:

- Téléchargement aval SSDO, divisé en:
  - Lancer téléchargement aval SSDO
  - Segmenter téléchargement aval SSDO
- Téléchargement amont SSDO, divisé en:
  - Lancer Téléchargement amont SSDO
  - Segmenter téléchargement amont SSDO
- Abandonner SSDO.

La Figure 24 définit les protocoles de téléchargement aval, segmenté et accéléré de SSDO.

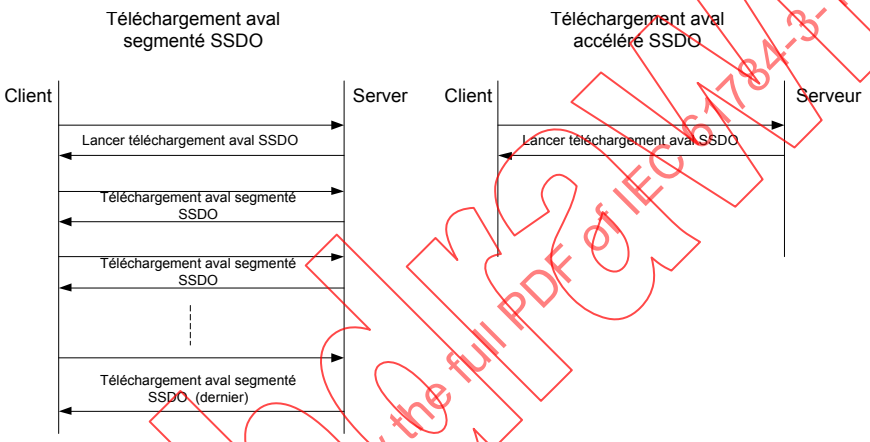


Figure 24 – Protocoles de téléchargement aval de SSDO

La Figure 25 définit les protocoles de téléchargement amont, segmentés et accélérés de SSDO.

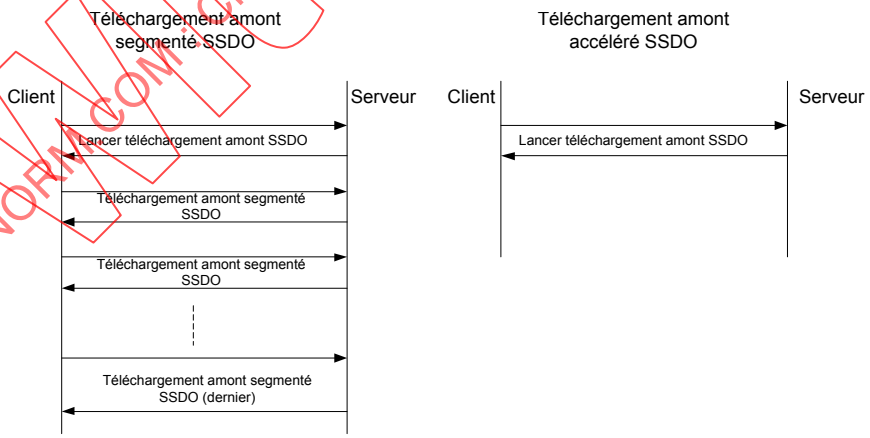


Figure 25 – Protocoles de téléchargement amont de SSDO

7.3.4 Lancer Téléchargement aval de SSDO

La Figure 26 décrit le protocole de lancement du téléchargement aval.

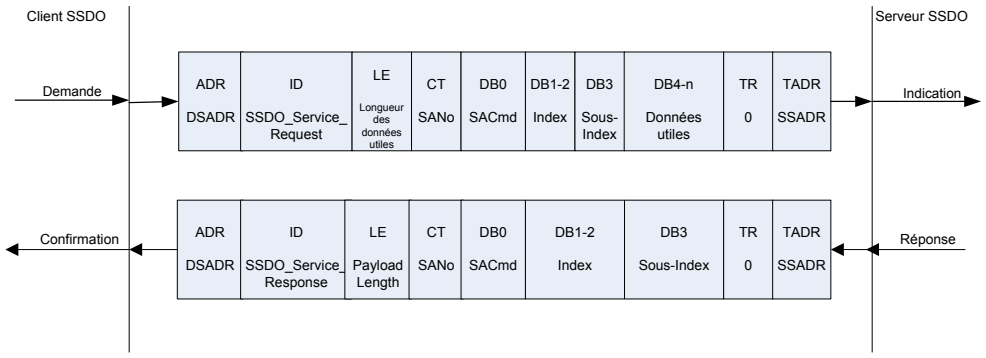


Figure 26 – Protocole de lancement de téléchargement aval de SSDO

Le Tableau 15 définit les champs d'un télégramme SSDO_Service_Request Lancer téléchargement aval.

Tableau 15 – Champs de télégramme de SSDO_Service_Request Lancer Téléchargement aval

Champ	Information	Contenu / Valeur
ADR	Adresse du serveur SSDO auquel il faut accéder	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Longueur des données utiles	4 à 254
CT	Numéro de demande d'accès au SOD (SANo)	1 à 65 535, 0 non autorisé
DB0	Commande d'accès au SOD (SACmd) accéléré ou segmenté	00100001b ou 00101001b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
DB4 – DBn	Données utiles (en cas de téléchargement aval segmenté, DB4 à DB7 contient la taille des données utiles du bloc. Si la taille des données est 0, elle n'est pas indiquée.)	Données utiles
TR	Non utilisé	0
TADR	Adresse de client SSDO	SSADR

Le Tableau 16 définit les champs d'un télégramme SSDO_Service_Response Lancer téléchargement aval.

Tableau 16 – Champs de télégramme de SSDO_Service_Response Lancer Téléchargement aval

Champ	Information	Contenu / Valeur
ADR	Adresse de client SSDO auquel il faut répondre	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Longueur des données utiles	4
CT	Echo du numéro de demande d'accès au SOD	1 à 65 535, 0 non autorisé
DB0	Commande d'accès au SOD (SACmd) accéléré ou segmenté	00100001b ou 00101001b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
TR	Non utilisé	0
TADR	Adresse du serveur SSDO	SSADR

7.3.5 Téléchargement aval segmenté de SSDO

La Figure 27 définit le protocole de lancement de téléchargement aval segmenté.

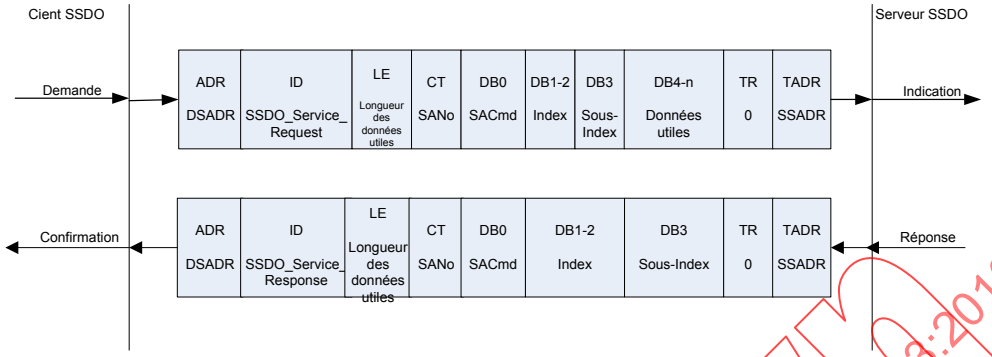


Figure 27 – Protocole de téléchargement aval segmenté de SSDO

Le Tableau 17 définit les champs d'un télégramme de SSDO_Service_Request de téléchargement aval segmenté.

Tableau 17 – Champs de télégramme SSDO_Service_Request de téléchargement aval segmenté

Champ	Information	Contenu / Valeur
ADR	Adresse du serveur SSDO auquel il faut accéder	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Longueur des données utiles	4 à 254
CT	Numéro de demande d'accès au SOD (SANo)	1 à 65 535, 0 non autorisé
DB0	Segment de milieu ou de fin de Commande d'accès au SOD (SACmd)	000x1001b ou 010x1001b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
DB4 – DBn	Données utiles	Données utiles
TR	Non utilisé	0
TADR	Adresse de client SSDO	SSADR

Le Tableau 18 définit les champs d'un télégramme SSDO_Service_Response de téléchargement aval segmenté.

Tableau 18 – Champs de télégramme SSDO_Service_Response de téléchargement aval segmenté

Champ	Information	Contenu / Valeur
ADR	Adresse de client SSDO auquel il faut répondre	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Longueur des données utiles	4
CT	Écho du numéro de demande d'accès au SOD	1 à 65 535, 0 non autorisé
DB0	Segment de milieu ou de fin de Commande d'accès au SOD (SACmd)	000x1001b ou 010x1001b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255

Champ	Information	Contenu / Valeur
TR	Non utilisé	0
TADR	Adresse du serveur SSDO	SSADR

7.3.6 Lancer Téléchargement amont de SSDO

La Figure 28 décrit le protocole Lancer téléchargement amont. Le télégramme Lancer téléchargement amont doit toujours être en accéléré. Sur la base de la réponse, le client SSDO doit poursuivre avec un transfert accéléré ou segmenté.

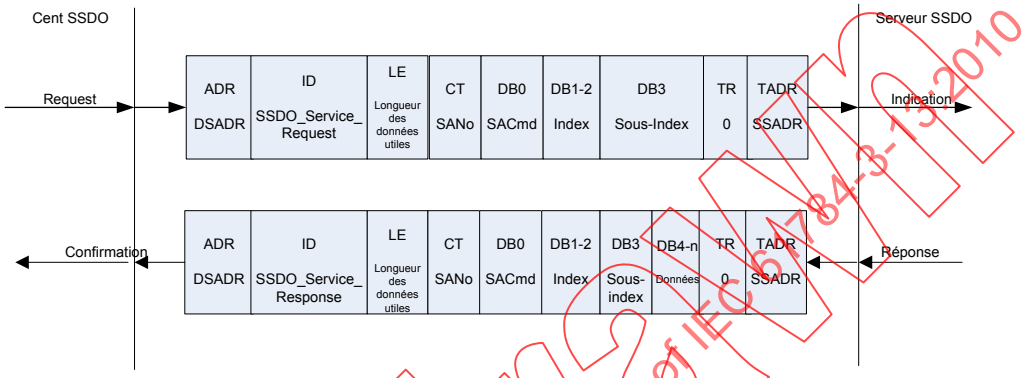


Figure 28 – Protocole de lancement de téléchargement amont de SSDO

Le Tableau 19 définit les champs d'un télégramme SSDO_Service_Request Lancer téléchargement amont.

Tableau 19 – Champs de télégramme de SSDO_Service_Request Lancer Téléchargement amont

Champ	Information	Contenu / Valeur
ADR	Adresse du serveur SSDO auquel il faut accéder	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Longueur des données utiles	4
CT	Numéro de demande d'accès au SOD (SANo)	1 à 65 535, 0 non autorisé
DB0	Commande d'accès au SOD (SACmd) accéléré	00100000b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
TR	Non utilisé	0
TADR	Adresse de client SSDO	SSADR

Le Tableau 20 définit les champs d'un télégramme SSDO_Service_Response Lancer téléchargement amont.

Tableau 20 – Champs de télégramme de SSDO_Service_Response Lancer Téléchargement amont

Champ	Information	Contenu / Valeur
ADR	Adresse de client SSDO auquel il faut répondre	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Longueur des données utiles	4 à 254

Champ	Information	Contenu / Valeur
CT	Echo du numéro de demande d'accès au SOD	1 à 65 535, 0 non autorisé
DB0	Commande d'accès au SOD (SACmd) accéléré ou segmenté	00100000b ou 00101000b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
DB4 – DBn	Données utiles (en cas de téléchargement amont segmenté, DB4 à DB7 contient la taille des données utiles du bloc. Si la taille des données est 0, elle n'est pas indiquée.)	Données utiles
TR	Non utilisé	0
TADR	Adresse du serveur SSDO	SSADR

7.3.7 Téléchargement amont segmenté de SSDO

La Figure 29 décrit le protocole de Téléchargement amont segmenté. Le client SSDO demande toujours un segment de milieu. Sur la base de la réponse, le client SSDO doit décider si un segment de milieu ou le segment de fin a été envoyé.

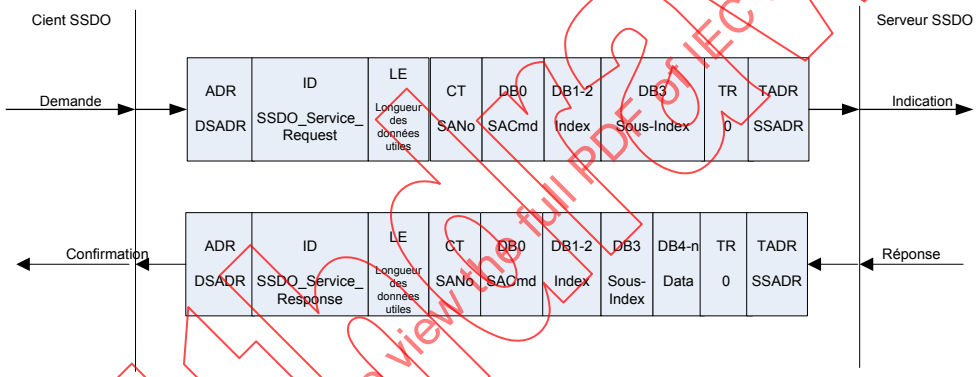


Figure 29 – Protocole de téléchargement amont segmenté de SSDO

Le Tableau 21 définit les champs d'un télégramme SSDO_Service_Request de téléchargement amont segmenté.

Tableau 21 – Champs de télégramme de SSDO_Service_Request de téléchargement amont segmenté

Champ	Information	Contenu / Valeur
ADR	Adresse du serveur SSDO auquel il faut accéder	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Longueur des données utiles	4
CT	Numéro de demande d'accès au SOD (SANo)	1 à 65 535, 0 non autorisé
DB0	Segment de milieu de commande d'accès au SOD (SACmd)	000x1000b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
TR	Non utilisé	0
TADR	Adresse de client SSDO	SSADR

Le Tableau 22 définit les champs d'un télégramme SSDO_Service_Response de téléchargement amont segmenté.

Tableau 22 – Champs de télégramme de SSDO_Service_Response de téléchargement amont segmenté

Champ	Information	Contenu / Valeur
ADR	Adresse de client SSDO auquel il faut répondre	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Longueur des données utiles	4 à 254
CT	Écho du numéro de demande d'accès au SOD	1 à 65 535, 0 non autorisé
DB0	Segment de milieu ou de fin de Commande d'accès au SOD (SACmd)	000x1000b ou 010x1000b
DB1, DB2	Index de l'entrée SOD à laquelle il faut accéder	0 à 65 535
DB3	Sous-index de l'entrée SOD à laquelle il faut accéder	0 à 255
DB4 -- DBn	Données utiles	Données utiles
TR	Non utilisé	0
TADR	Adresse du serveur SSDO	SSADR

7.3.8 Abandonner SSDO

La Figure 30 décrit le protocole d'abandon de SSDO.

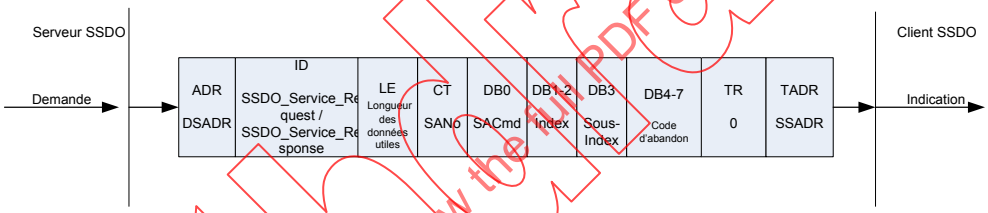


Figure 30 – Protocole Abandonner SSDO

Le Tableau 23 définit les champs d'un télégramme SSDO_Service_Request abandonné.

Tableau 23 – Champs de télégramme de SSDO_Service_Request de téléchargement amont segmenté

Champ	Information	Contenu / Valeur
ADR	Adresse de client SSDO à informer	DSADR
ID	SSDO_Service_Request	111000xxb
LE	Longueur des données utiles	8
CT	Numéro de demande d'accès au SOD (SANo)	1 à 65 535, 0 non autorisé
DB0	Commande d'accès au SOD (SACmd)	00000100b
DB1, DB2	Index d'entrée SOD	0 à 65 535
DB3	Sous-index d'entrée SOD	0 à 255
DB4 -- DB7	Code d'abandon, voir le Tableau 25	Code d'abandon
TR	Non utilisé	0
TADR	Adresse du serveur SSDO qui a initié l'abandon	SSADR

Le Tableau 24 définit les champs d'un télégramme SSDO_Service_Response abandonner.

Tableau 24 – Champs de télégramme SSDO_Service_Response de téléchargement amont segmenté

Champ	Information	Contenu / Valeur
ADR	Adresse de client SSDO à informer	DSADR
ID	SSDO_Service_Response	111001xxb
LE	Longueur des données utiles	8
CT	Numéro de demande d'accès au SOD (SANo)	1 à 65 535, 0 non autorisé
DB0	Commande d'accès au SOD (SACmd)	00000100b
DB1, DB2	Index d'entrée SOD	0 à 65 535
DB3	Sous-index d'entrée SOD	0 à 255
DB4 -- DB7	Code d'abandon, voir le Tableau 25	Code d'abandon
TR	Non utilisé	0
TADR	Adresse du serveur SSDO qui a initié l'abandon	SSADR

Les données utiles d'un télégramme Abandonner SSDO doivent contenir un code d'erreur comme spécifié dans le Tableau 25. Le code d'abandon doit être une valeur UNSIGNED32 (NONSIGNÉ32). Les codes d'abandon non indiqués dans le Tableau 25 doivent être réservés.

Tableau 25 – Codes d'abandon SSDO

Code d'abandon [a]	Description
05030000h	Réservé
05040000h	Expiration du délai de protocole SSDO
05040001h	ID de commande client/serveur non valide ou inconnue
05040002h	Taille de bloc non valide
05040003h	Numéro de séquence non valide
05040004h	Réservé
05040005h	Mémoire insuffisante
06010000h	Accès vers un objet non pris en charge
06010001h	Tentative de lecture d'un objet en écriture seule
06010002h	Tentative d'écriture d'un objet en lecture seule
06020000h	L'objet n'existe pas dans le dictionnaire d'objets
06040041h	L'objet ne peut pas être mis en correspondance avec le SPDO
06040042h	Le nombre et la longueur des objets à mettre en correspondance dépasseraient la longueur du SPDO
06040043h	Incompatibilité générale du paramètre
06040047h	Incompatibilité interne générale dans le dispositif
06060000h	Échec de l'accès dû à une erreur matérielle
06070010h	Le type de données ne correspond pas, la longueur du paramètre de service ne correspond pas
06070012h	Le type de données ne correspond pas, la longueur du paramètre de service est trop importante
06070013h	Le type de données ne correspond pas, la longueur du paramètre de service est trop faible
06090011h	Le sous-index n'existe pas
06090030h	Plage de valeurs du paramètre dépassée (uniquement pour l'accès en écriture)
06090031h	Valeur trop élevée du paramètre écrit
06090032h	Valeur trop élevée du paramètre écrit
06090036h	Valeur trop élevée du paramètre écrit
08000000h	Valeur trop élevée du paramètre écrit
08000020h	Les données ne peuvent pas être transmises ou enregistrées dans l'application
08000021h	Les données ne peuvent pas être transmises ou enregistrées dans l'application à cause du contrôle local
08000022h	Les données ne peuvent pas être transmises ou enregistrées dans l'application à cause de l'état actuel du dispositif
08000023h	Les données ne peuvent pas être transmises ou enregistrées dans l'application parce que l'application est occupée
[a] Comme indiqué par le suffixe "h", le code d'abandon est donné en notation hexadécimale.	

7.4 Gestion du réseau de sécurité (SNMT)

7.4.1 Généralités

La gestion du réseau de sécurité (SNMT) repose sur les nœuds et suit une structure logique maître/esclave. Le Maître SNMT doit être mis en œuvre par un SCM ou un outil de configuration. Les services SNMT suivants doivent être fournis:

- demande d'UDID,
- attribution d'adresse de nœud,
- services étendus, qui doivent être désignés par le champ DB0 et comporter les services suivants:
  - contrôle de l'état de communication,
  - sauvegarde du nœud,
  - attribution d'adresse de nœud supplémentaire.

7.4.2 Types de télégramme SNMT

Le champ ID (bit 2 à 4) doit identifier le type de télégramme SNMT. Le Tableau 26 définit les types de télégrammes SNMT:

Tableau 26 – Types de télégrammes SNMT (champ ID, bits 2, 3 et 4)

Numéro de bit du champ d'identification								Type de télégramme	Service
7	6	5	4	3	2	1	0		
1	0	1	0	0	0			SNMT_Request_UDID	Demande d'UDID
1	0	1	0	0	1			SNMT_Response_UDID	
1	0	1	0	1	0			SNMT_Assign_SADR	Attribution d'adresse de nœud
1	0	1	0	1	1			SNMT_SADR_assigned	
1	0	1	1	0	0			SNMT_Service_Request	Services étendus
1	0	1	1	0	1			SNMT_Service_Response	
1	0	1	1	1	1			SNMT_SN_reset_guarding_SCM	Réinitialisation de la temporisation de sauvegarde

7.4.3 Services et protocoles SNMT

7.4.3.1 Demande / réponse d'UDID

La demande/réponse d'UDID doit permettre à un Maître SNMT de récupérer l'UDID d'un Esclave SNMT spécifique. Pour cela, le Maître SNMT doit envoyer la demande UDID à un Esclave SNMT spécifique. L'Esclave SNMT doit répondre, même si la valeur dans l'adresse ADR ne correspond pas à la SADR interne des SN.

NOTE 1 Ceci peut arriver s'il n'y a pas eu d'attribution préalable de SADR au SN.

Le champ TADR de la réponse doit contenir la SADR de la réponse UDID.

NOTE 2 Si l'Esclave SNMT répond par sa propre (probablement invalide) SADR, le Maître SNMT peut ne pas pouvoir faire correspondre la réponse avec une demande donnée.

Une erreur d'exécution de ce protocole doit être indiquée par un UDID 00-00-00-00-00-00 dans la réponse.

La Figure 31 décrit le protocole de demande UDID.

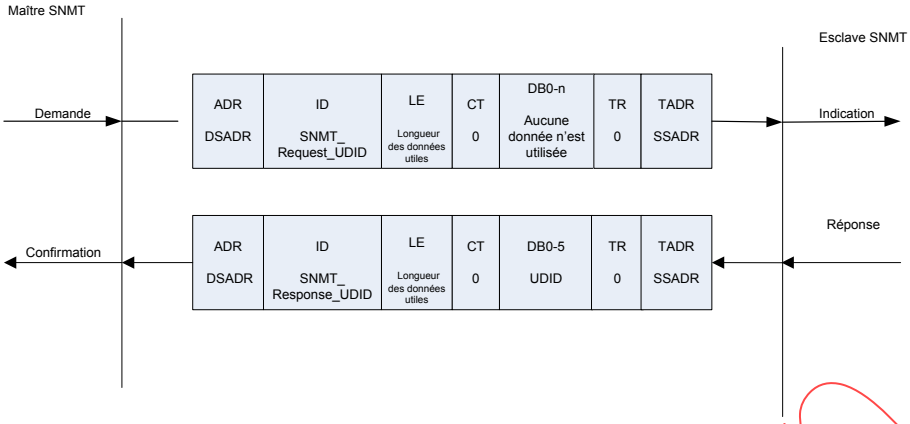


Figure 31 – Protocole de demande / réponse d'UDID

Le Tableau 27 définit les champs d'un télégramme SNMT_Request_UDID.

Tableau 27 – Champs d'un télégramme SNMT_Request_UDID

Champ	Information	Contenu / Valeur
ADR	Adresse d'Esclave SNMT de destination	DSADR
ID	SNMT_Request_UDID	101000xxb
LE	Longueur des données utiles	0
CT	Non utilisé	0
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 28 définit les champs d'un télégramme SNMT_Response_UDID.

Tableau 28 – Champs d'un télégramme SNMT_Response_UDID

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	SNMT_Response_UDID	101001xxb
LE	Longueur des données utiles	6
CT	Non utilisé	0
DB0 -- DB5	UDID du SN	UDID
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

7.4.3.2 Attribution de SADR

L'attribution d'une SADR doit être utilisée par un Maître SNMT pour établir la SADR d'un Esclave SNMT spécifique. Le protocole doit en outre attribuer le numéro de domaine de sécurité (SDN) à l'Esclave SNMT. Le SDN doit être établi au sein du champ d'adresse de la partie deux de la PDU en utilisant une opération logique XOR - OU exclusif (voir 7.1). Un SN doit envoyer une réponse unique uniquement si l'UDID reçu correspond à son UDID propre.

Si le(la) SDN/SADR demandé(e) par le Maître SNMT ne peut pas être attribué(e), l'Esclave SNMT doit répondre au moyen de son(sa) SDN/SADR courant(e). Le Maître SNMT doit alors

comparer le(la) SDN/SADR de la réponse et l'UDID avec le(la)(l') SDN/SADR/UDID établi(e) pour vérifier que l'attribution est réussie.

NOTE Un échec d'attribution peut avoir lieu si le(la) SDN/SADR des nœuds est codé(e) en dur au moyen d'un commutateur et qu'il(elle) ne correspond pas à la valeur SCM.

Une erreur d'exécution de ce service doit être indiquée par un UDID 00-00-00-00-00-00 dans la réponse.

La Figure 32 décrit le protocole d'attribution d'une SADR.

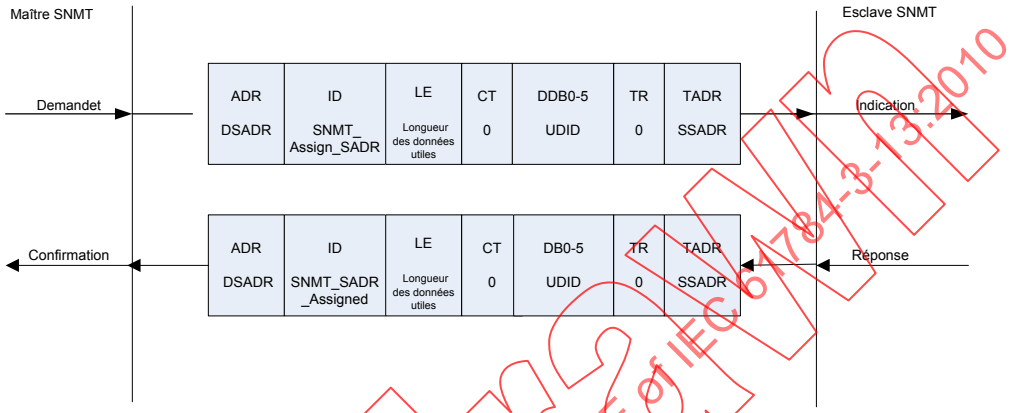


Figure 32 – Protocole d'attribution d'une SADR

Le Tableau 29 définit les champs d'un télégramme SNMT_Assign_SADR.

Tableau 29 – Champs d'un télégramme SNMT_Assign_SADR

Champ	Information	Contenu / Valeur
ADR	Adresse de l'Esclave SNMT à attribuer	DSADR
ID	SNMT_Assign_SADR	101010xxb
LE	Longueur des données utiles	6
CT	Non utilisé	0
DB0 -- DB5	UDID du SN	UDID
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 30 définit les champs d'un télégramme SNMT_SADR_Assigned.

Tableau 30 – Champs d'un télégramme SNMT_SADR_Assigned

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	SNMT_SADR_ Assigned	101011xxb
LE	Longueur des données utiles	6
CT	Non utilisé	0
DB0 -- DB5	UDID du SN	UDID
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

7.4.3.3 Réinitialiser intervalle de sauvegarde de nœud

Le service Réinitialiser intervalle de sauvegarde de nœud doit être utilisé par le SN pour déclencher une sauvegarde du nœud sur le SCM. Le SCM doit protéger tous les SN, qu'il y ait ou non correspondance entre la SADR ou SDN du télégramme et la SADR et le SDN du SCM.

Le télégramme doit toujours être envoyé avec la SADR et le SDN à 1.

NOTE Ceci est dû au fait que le télégramme est envoyé avant attribution d'une SADR ou d'un SDN au nœud.

La Figure 33 décrit le protocole de service Réinitialiser intervalle de sauvegarde de nœud.

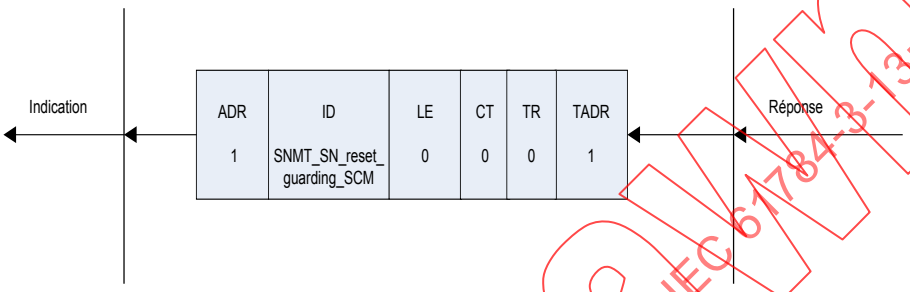


Figure 33 – Protocole du service Réinitialiser intervalle de sauvegarde de nœud

Le Tableau 31 définit les champs d'un télégramme SNMT_SN_reset_guarding_SCM.

Tableau 31 – Champs d'un télégramme SNMT_SN_reset_guarding_SCM

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	1
ID	SNMT_SN_reset_guarding_SCM	$101111xx_b$
LE	Longueur des données utiles	0
CT	Non utilisé	0
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	1

7.4.3.4 Services étendus de SNMT

Les services étendus SNMT sont spécifiés par le champ ID et le champ DB0 de la PDU FSPC 13/1 (voir le Tableau 32 et le Tableau 33).

Tableau 32 – Types de télégrammes de demande SNMT

Numéro de bit de DB0								Type de télégramme SNMT
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	SNMT_SN_set_to_PRE_OP
0	0	0	0	0	0	1	0	SNMT_SN_set_to_OP
0	0	0	0	0	1	0	0	SNMT_SCM_set_to_STOP
0	0	0	0	0	1	1	0	SNMT_SCM_set_to_OP
0	0	0	0	1	0	0	0	SNMT_SCM_guard_SN
0	0	0	0	1	0	1	0	SNMT_assign_additional_SADR
0	0	0	0	1	1	0	0	SNMT_SN_ACK
0	0	0	0	1	1	1	0	SNMT_assign_UDID_SCM

Tableau 33 – Types de télégramme de réponse SNMT

Numéro de bit de DB0								Type de télégramme SNMT
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	1	SNMT_SN_status_PRE_OP
0	0	0	0	0	0	1	1	SNMT_SN_status_OP
0	0	0	0	0	1	0	1	SNMT_assigned_additional_SADR
0	0	0	0	0	1	1	1	SNMT_SN_FAIL
0	0	0	0	1	0	0	1	SNMT_SN_busy
0	0	0	0	1	1	1	1	SNMT_assigned_UDID_SCM

7.4.3.5 SN mis à l'état pré-opérationnel

Le service SN mis à l'état pré-opérationnel doit être utilisé pour faire passer un Esclave SNMT de l'état opérationnel à l'état pré-opérationnel de communication.

La Figure 34 décrit le protocole SN mis à l'état pré-opérationnel.

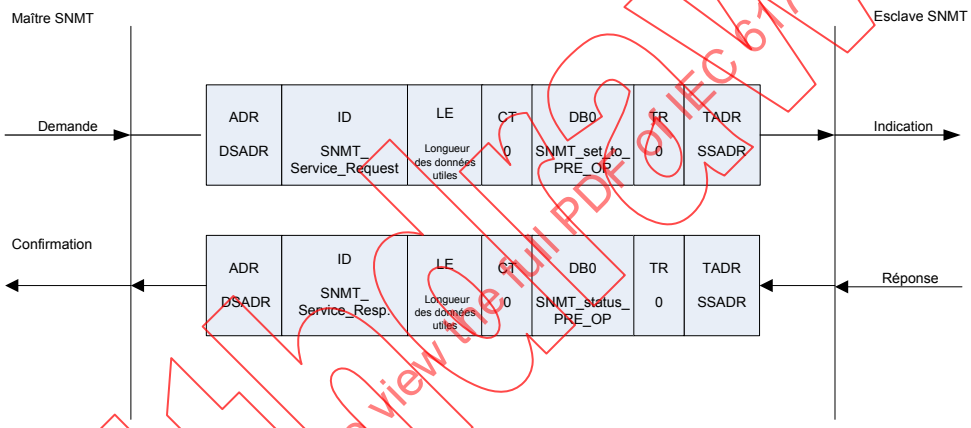


Figure 34 – Protocole SN mis à l'état pré-opérationnel

Le Tableau 34 définit les champs d'un télégramme SNMT_SN_set_to_PRE_OP.

Tableau 34 – Champs d'un télégramme SNMT_SN_set_to_PRE_OP

Champ	Information	Contenu / Valeur
ADR	Adresse de l'Esclave SNMT auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SN_set_to_PRE_OP	00000000b
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 35 définit les champs d'un télégramme SNMT_SN_status_PRE_OP.

Tableau 35 – Champs d'un télégramme SNMT_SN_status_PRE_OP

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR

Champ	Information	Contenu / Valeur
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SN_status_PRE_OP	00000001b
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

7.4.3.6 SN mis à l'état opérationnel

Le service SN mis à l'état opérationnel doit être utilisé pour faire passer un Esclave SNMT de l'état pré-opérationnel à l'état opérationnel de communication. La réponse doit être soit l'information que le SN est passé à l'état opérationnel, soit une information d'erreur utilisée par le SCM pour rendre compte à son application.

La Figure 35 décrit le protocole SN mis à l'état opérationnel.

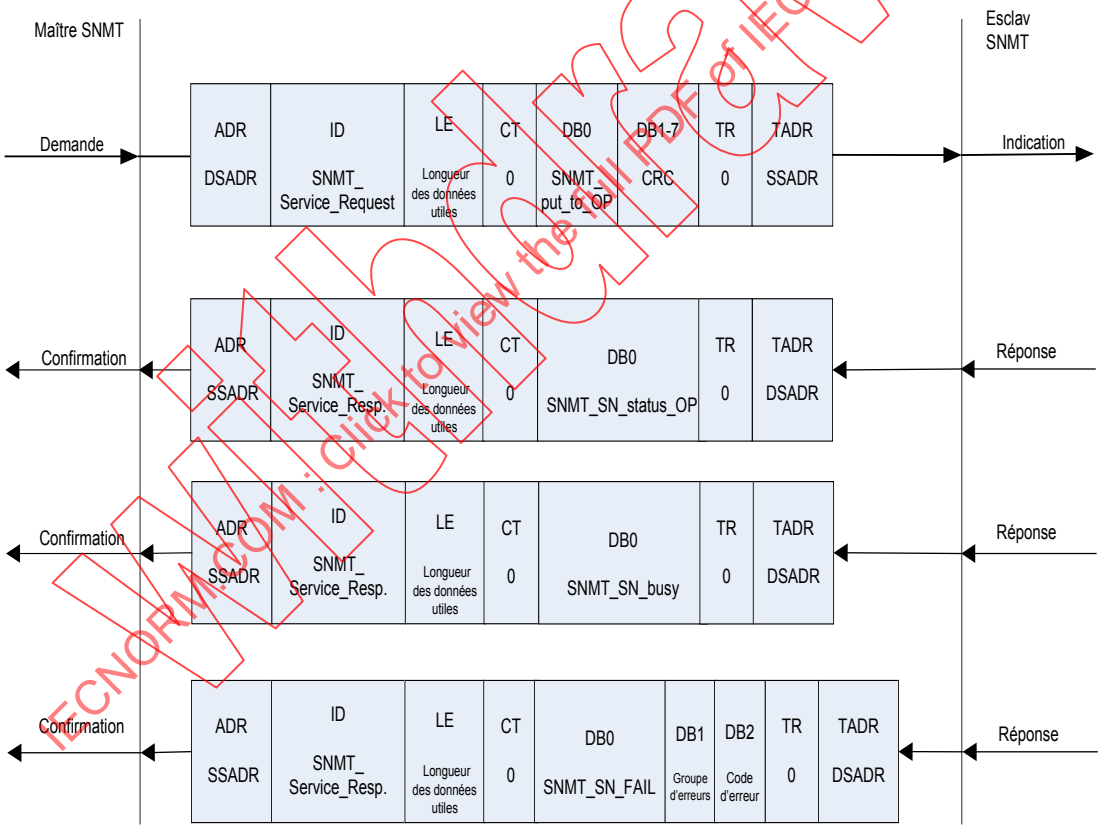


Figure 35 – Protocole SN mis à l'état opérationnel

Le Tableau 36 définit les champs d'un télégramme SNMT_SN_set_to_OP.

Tableau 36 – Champs d'un télégramme SNMT_SN_set_to_OP

Champ	Information	Contenu / Valeur
ADR	Adresse d'Esclave SNMT auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb

Champ	Information	Contenu / Valeur
LE	Longueur des données utiles	7
CT	Non utilisé	0
DB0	SNMT_SN_set_to_OP	00000010b
DB1 -- DB2	Somme de contrôle du paramètre	Voir en 7.5.4.7
DB3 -- DB6	Horodatage du paramètre	Voir en 7.5.4.7
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 37 définit les champs d'un télégramme SNMT_SN_status_OP.

Tableau 37 – Champs d'un télégramme SNMT_SN_status_OP

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SN_status_OP	00000011b
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

Le Tableau 38 définit les champs d'un télégramme SNMT_SN_busy.

Tableau 38 – Champs d'un télégramme SNMT_SN_busy

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SN_busy[a]	00001001b
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

[a] L'application doit répondre à un télégramme SNMT_SN_busy (par exemple pendant le calcul du CRC ou lors de la sauvegarde des paramètres).

Le Tableau 39 définit les champs d'un télégramme SNMT_SN_FAIL.

Tableau 39 – Champs d'un télégramme SNMT_SN_FAIL

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	3
CT	Non utilisé	0
DB0	SNMT_SN_FAIL	00000111b

Champ	Information	Contenu / Valeur
DB1	Groupe d'erreurs (voir le Tableau 40)	0 à 255
DB2	Code d'erreur (voir le Tableau 41)	0 à 255
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

Le Tableau 40 définit les valeurs du groupe d'erreurs SNMT_SN_FAIL.

Tableau 40 – Valeurs du groupe d'erreurs SNMT_SN_FAIL

Valeur	Groupe d'erreurs
0	Dispositif
1	Application
2	Paramètre
3	Spécifique au fournisseur
4	Pile FSCP 13/1
5 -- 255	Réservé pour un futur usage

Le Tableau 41 définit les valeurs du code d'erreur SNMT_SN_FAIL.

Tableau 41 – Valeurs du code d'erreur SNMT_SN_FAIL

Valeur	Groupe d'erreurs
0	Par défaut
1 -- 255	Spécifique au fournisseur

7.4.3.7 Acquittement SN de SNMT

Le service Acquitter SN doit être utilisé par le SCM pour acquitter une erreur indiquée (voir 7.4.3.6) au niveau du SN.

La Figure 36 décrit le protocole Acquitter SN.

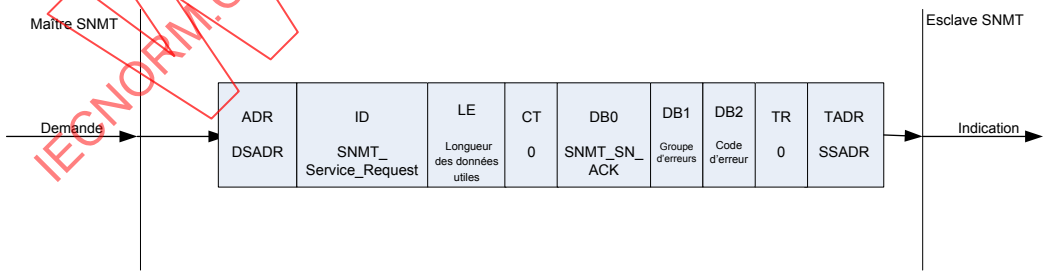


Figure 36 – Protocole Acquitter SN

Le Tableau 42 définit les champs d'un télégramme SNMT_SN_ACK.

Tableau 42 – Champs d'un télégramme SNMT_SN_ACK

Champ	Information	Contenu / Valeur
ADR	Adresse d'Esclave SNMT auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb

Champ	Information	Contenu / Valeur
LE	Longueur des données utiles	3
CT	Non utilisé	0
DB0	SNMT_SN_ACK	00001100b
DB1	Groupe d'erreurs (voir le Tableau 40)	0 à 255
DB2	Code d'erreur (voir le Tableau 41)	0 à 255
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

7.4.3.8 SCM mis à l'état arrêté

Le service SCM mis à l'état arrêté doit être utilisé pour mettre hors-tension un SCM (de l'état Opérationnel à l'état Arrêté) pour configurer le système au moyen d'un outil de configuration externe.

La Figure 37 décrit le protocole SCM mis à l'état arrêté.

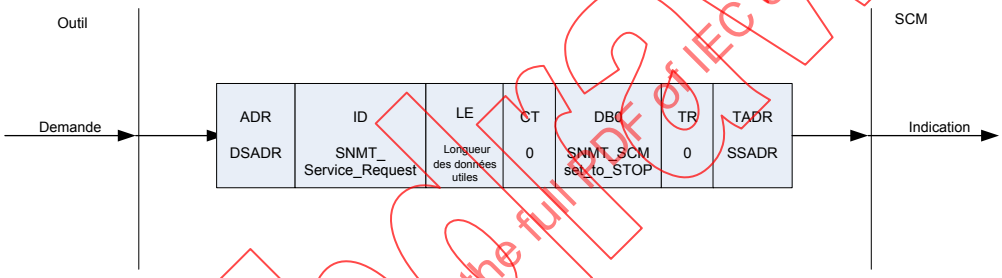


Figure 37 – Protocole SN mis à l'état arrêté

Le Tableau 43 définit les champs d'un télégramme SNMT_SCM_set_to_STOP.

Tableau 43 – Champs d'un télégramme SNMT_SCM_set_to_STOP

Champ	Information	Contenu / Valeur
ADR	Adresse du SCM auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SCM_set_to_STOP	00000100b
TR	Non utilisé	0
TADR	Adresse de l'OUTIL	SSADR

7.4.3.9 SCM mis à l'état opérationnel

Le service SCM mis à l'état opérationnel doit être utilisé pour faire passer le SCM de l'état Arrêté à l'état Opérationnel. Ce service peut être utilisé après achèvement de la configuration au moyen d'un outil externe.

La Figure 38 décrit le protocole SCM mis à l'état opérationnel.

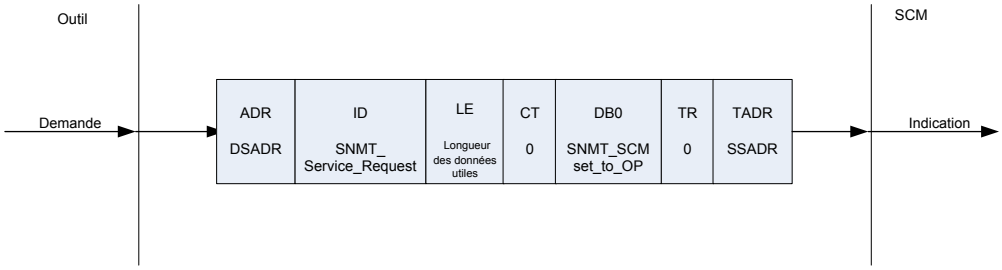


Figure 38 – Protocole SCM mis à l'état opérationnel

Le Tableau 44 définit les champs d'un télégramme SNMT_SCM_set_to_OP.

Tableau 44 – Champs d'un télégramme SNMT_SCM_set_to_OP

Champ	Information	Contenu / Valeur
ADR	Adresse du SCM auquel il faut accéder	DSADR
ID	SNMT_Service_Request	101100xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SCM_set_to_OP	00000110b
TR	Non utilisé	0
TADR	Adresse de l'OUTIL	SSADR

7.4.3.10 Sauvegarde du nœud

Le service de sauvegarde du nœud doit être utilisé pour protéger un SN dans l'état de communication Opérationnel. Le SN doit répondre en indiquant son état. Si un SN ne reçoit pas un télégramme SNMT_SCM_guard_SN au cours de sa durée de vie (voir 7.5.4.5), il doit repasser à l'état pré-opérationnel. La signalisation d'une perte de demande ou de réponse de l'application doit être spécifique au fournisseur.

La Figure 39 décrit le protocole de sauvegarde du nœud.

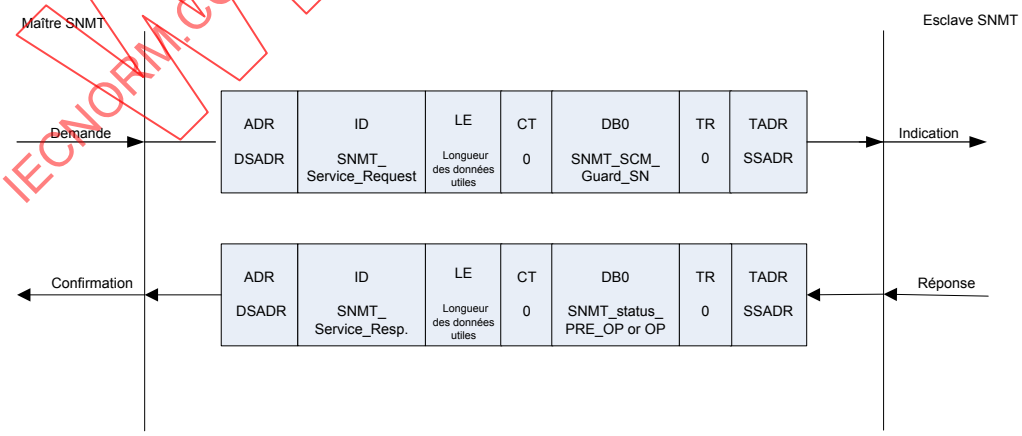


Figure 39 – Protocole de sauvegarde du nœud

Le Tableau 45 définit les champs d'un télégramme SNMT_SCM_guard_SN.

Tableau 45 – Champs d'un télégramme SNMT_SCM_guard_SN

Champ	Information	Contenu / Valeur
ADR	Adresse d'Esclave SNMT auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SCM_guard_SN	00001000b
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 46 définit les champs de télégrammes SNMT_SN_status_PRE_OP / SNMT_SN_status_OP.

Tableau 46 – Champs de télégrammes SNMT_SN_status_OP/SNMT_SN_status_OP

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	1
CT	Non utilisé	0
DB0	SNMT_SN_status_PRE_OP ou SNMT_SN_status_OP	00000001b ou 00000011b
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

7.4.3.11 Attribution de SADR supplémentaire

Le service Attribution de SADR supplémentaire doit être utilisé pour attribuer une adresse SADR supplémentaire à un Esclave SNMT. Ce service doit uniquement être utilisé si un SN ne prend pas en charge plusieurs TxSPDO (voir 7.5.4.16).

Si l'Esclave SNMT ne prend pas en charge plusieurs TxSPDO, il doit répondre par une valeur 0 de SADR.

La Figure 40 décrit le protocole d'attribution d'une SADR supplémentaire.

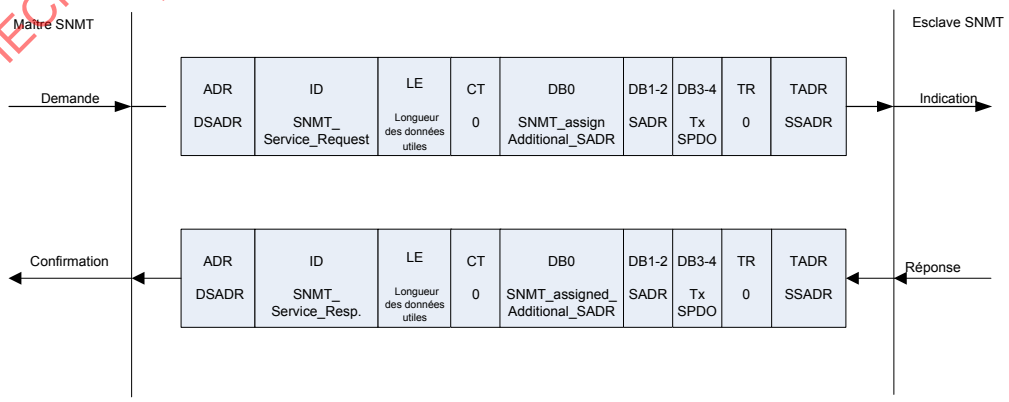


Figure 40 – Protocole d'attribution d'une SADR supplémentaire

Le Tableau 47 définit les champs d'un télégramme SNMT_assign_additional_SADR.

Tableau 47 – Champs d'un télégramme SNMT_assign_additional_SADR

Champ	Information	Contenu / Valeur
ADR	Adresse d'Esclave SNMT auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb
LE	Longueur des données utiles	5
CT	Non utilisé	0
DB0	SNMT_assign_additional_SADR	00001010b
DB1,DB2	SADR supplémentaire à attribuer	SADR
DB3,DB4	TxSPDO auquel la SADR supplémentaire est attribuée	2 -- 1 023
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 48 définit les champs d'un télégramme SNMT_assigned_additional_SADR.

Tableau 48 – Champs d'un télégramme SNMT_assigned_additional_SADR

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	5
CT	Non utilisé	0
DB0	SNMT_assigned_additional_SADR	00000101b
DB1, DB2	SADR qui a été attribuée	SADR
DB3, DB4	TxSPDO auquel la SADR a été attribuée	2 -- 1 023
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

7.4.3.12 Attribution d'UDID de SCM

Le service Attribution d'UDID de SCM doit être utilisé pour attribuer l'UDID du SCM à l'Esclave SNMT.

Une erreur d'exécution de ce service doit être indiquée par une valeur d'UDID 00-00-00-00-00-00 dans la réponse.

La Figure 41 décrit le protocole d'attribution d'UDID de SCM.

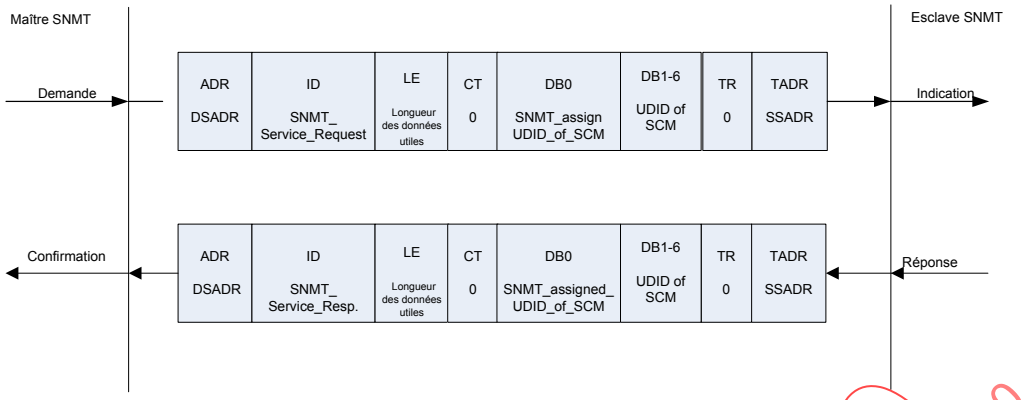


Figure 41 – Protocole d'Attribution d'UDID de SCM

Le Tableau 49 définit les champs d'un télégramme SNMT_assign_UDID_of_SCM.

Tableau 49 – Champs d'un télégramme SNMT_assign_UDID_of_SCM

Champ	Information	Contenu / Valeur
ADR	Adresse d'Esclave SNMT auquel il faut accéder	DSADR
ID	Demande de service SNMT	101100xxb
LE	Longueur des données utiles	7
CT	Non utilisé	0
DB0	SNMT_assign_UDID_of_SCM	00001110b
DB1 -- DB6	UDID du SCM	UDID
TR	Non utilisé	0
TADR	Adresse de Maître SNMT	SSADR

Le Tableau 50 définit les champs d'un télégramme SNMT_assigned_UDID_of_SCM.

Tableau 50 – Champs d'un télégramme SNMT_assigned_UDID_of_SCM

Champ	Information	Contenu / Valeur
ADR	Adresse de Maître SNMT	DSADR
ID	Réponse de service SNMT	101101xxb
LE	Longueur des données utiles	7
CT	Non utilisé	0
DB0	SNMT_assigned_UDID_of_SCM	00001111b
DB1 -- DB6	UDID du SCM	UDID
TR	Non utilisé	0
TADR	Adresse d'Esclave SNMT	SSADR

7.5 Dictionnaire d'objets de sécurité (SOD)

7.5.1 Généralités

Les sous-classes suivantes définissent la structure et les entrées d'un dictionnaire d'objets de sécurité; ceci doit être commun à tous les dispositifs.

7.5.2 Définition d'une entrée de dictionnaire d'objets

7.5.2.1 Généralités

Une entrée de dictionnaire d'objets doit être définie par les éléments suivants:

— index,
— type d'objet
— nom,
— type de données,
— catégorie,
— accès,
— plage de valeurs,
— valeur par défaut,
— correspondance SPDO.

7.5.2.2 Index

L'index doit décrire la position des objets dans le dictionnaire d'objets. Il agit comme une sorte d'adresse pour référencer le champ de données requis. Un index doit être déclaré en valeur hexadécimale.

Un objet peut être subdivisé en sous-index. Le sous-index doit être utilisé pour référencer des champs de données dans un objet composite tel qu'une matrice ou un enregistrement. Le sous-index n'est pas spécifié dans les présentes.

7.5.2.3 Type d'objet

Le type d'objet doit indiquer quelle sorte d'objet se trouve à cet index particulier dans le dictionnaire d'objets. Les types d'objets doivent être tels que spécifié dans le Tableau 51.

Tableau 51 – Définition des types d'objet

Type d'objet	Commentaires	Code
NULL (nul)	Une entrée de dictionnaire sans champs de données	0
DOMAIN (DOMAINE)	Grande quantité variable de données, par exemple le code d'un programme exécutable	2
DEFTYPE	Représente une définition de type de données de base (par exemple BOOLEAN, UNSIGNED16, REAL32)	5
DEFSTRUCT	Définit un type d'enregistrement	6
VAR	Une valeur unique (par exemple UNSIGNED8, BOOLENA, REAL32, INTEGER16, VISIBLE STRING - CHAÎNE VISIBLE)	7
ARRAY (Matrice)	Un objet à champs de données multiples où chaque champ de données est une simple variable du MÊME type de données de base, par exemple une matrice de UNSIGNED16, etc. Le sous-index 0 est du type UNSIGNED8 et par conséquent ne fait pas partie des données de la MATRICE.	8
RECORD (ENREGISTREMENT)	Un objet à champ de données multiples dont les champs de données peuvent être toutes combinaisons de variables simples. Le sous-index 0 est du type UNSIGNED8 et par conséquent ne fait pas partie des données d'ENREGISTREMENT	9

7.5.2.4 Nom

Le nom doit fournir une description textuelle simple de la fonction de cet objet particulier. Le nom doit être conforme à la CEI 61131-3 et doit être constitué:

— du préfixe du domaine, indiquant l'association de l'objet à un domaine fonctionnel et utilisant jusqu'à 4 caractères en majuscules suivis d'un trait bas,

— d'un nom (verbal) constitué de mots, chacun doit commencer par une majuscule suivie par des minuscules ou des chiffres, sans traits bas ou espaces,

— du suffixe de type de données, indiquant le type de données de l'objet (trait bas suivi de 5 caractères en majuscules ou de chiffres).

La longueur totale du nom ne doit pas dépasser 32 caractères.

7.5.2.5 Type de Données

Le type de données doit fournir des informations sur le type de données de l'objet. Il comprend les types de données de base prédéfinis suivants: BOOLEEN (BOOLEAN), nombre à virgule flottante, entier UNSIGNED (NON SIGNÉ), entier signé, chaîne visible d'octets et DOMAINE. Il comprend également les types de données composés prédéfinis et peut aussi inclure des types spécifiques au fabricant ou au dispositif.

Les éléments suivants ne doivent pas être définis:

— les enregistrements d'enregistrements,

— les matrices d'enregistrements, ou

— des enregistrements avec des matrices comme champs de cet enregistrement.

Dans le cas où un objet est une matrice ou un enregistrement, le sous-index doit être utilisé pour référencer un champ de données de type de données de base au sein de l'objet.

Le type de données doit être un type de données de base pour les objets de type ARRAY (MATRICE) ou un type de données composé pour les types d'objets RECORD (ENREGISTREMENT).

7.5.2.6 Catégorie

La catégorie définit si l'objet est obligatoire (M) ou facultatif (O). Un objet obligatoire doit être mis en œuvre dans un dispositif. Un objet facultatif peut être mis en œuvre dans un dispositif. La prise en charge de certains objets ou fonctions peut cependant nécessiter la mise en œuvre d'objets liés. Dans ce cas, les relations sont décrites dans la spécification détaillée de l'objet.

La catégorie peut être conditionnelle (Cond) si la catégorie M/O d'un objet dépend de la mise en œuvre d'un autre objet.

La catégorie doit faire référence à un objet de type de données composé dans son ensemble et non à un sous-index particulier. Un objet obligatoire peut être constitué de sous-index obligatoires et facultatifs. Ceci signifie que l'objet doit être pris en charge mais que certains de ses sous-index sont facultatifs. Il est également admis de définir des objets facultatifs avec des sous-index obligatoires. Ceci signifie que ces sous-index doivent être pris en charge si l'objet est mis en œuvre.

7.5.2.7 Accès

L'accès définit les droits d'accès à un objet particulier. Le point de vue est de l'extérieur vers l'intérieur du dispositif.

Les droits d'accès sont spécifiés dans le Tableau 52.

Tableau 52 – Attributs d'accès pour des objets de données

Droit d'accès	Description
rw	accès en lecture et en écriture
wo	accès en écriture uniquement
RO	accès en lecture seule
Const	accès en lecture seule, valeur constante

Les objets facultatifs doivent être répertoriés dans le dictionnaire d'objets avec l'attribut rw; ils peuvent être utilisés en lecture seule. Les exceptions sont définies dans la spécification détaillée de l'objet.

7.5.2.8 Plage de valeurs

Cette entrée doit indiquer la plage de valeurs de l'objet correspondant. Elle peut comprendre une ou plusieurs valeurs distinctes ou des plages de valeurs. Si l'élément indiqué est un identifiant de type de données, la plage de valeurs complète du type de données mentionné doit être autorisée.

7.5.2.9 Valeur par défaut

Cette entrée doit indiquer la valeur d'initialisation qui doit être attribuée par la mise en œuvre du profil. Pour les objets qui peuvent être mis en correspondance, cette valeur doit également représenter la valeur de l'état de sécurité.

7.5.2.10 Mise en correspondance SPDO

Cette entrée indique si une entrée peut être mise en correspondance avec un message SPDO.

Les valeurs de mise en correspondance SPDO doivent être telles que spécifiées dans le Tableau 53.

Tableau 53 – Attributs de mise en correspondance SPDO pour des objets de données

Valeur	Description
Opt.	L'objet peut être mis en correspondance avec un SPDO
Non	L'objet ne doit pas être mis en correspondance avec un SPDO

7.5.2.11 Exemple de définition d'objets de type de données de base

Une définition complète d'objet de type de données de base (Type d'objet = VAR ou DOMAIN) est donnée dans le Tableau 54.

Tableau 54 – Exemple de définition d'objets de type de données de base

Élément	Valeur
Index	1234 h
Nom	SSDO_VarDummyParameter_S16
Type d'objet	VAR
Type de données	INTEGER16 (Entier16)
Plage de valeurs	-15 000 à 10 000
Valeur par défaut	0

Élément	Valeur
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Opt.

7.5.2.12 Exemple de définition d'objets de type de données composé

Les définitions d'objets de type de données composé (Object Type = ARRAY ou RECORD) sont données sous forme réduite, car Accès, Plage de valeurs, Valeur par défaut et Mise en correspondance SPDO doivent être définis individuellement pour le sous-index.

Le Tableau 55 donne un exemple de définition d'objet de type de données composé.

Tableau 55 – Exemple de définition d'objets de type de données composé

Élément	Valeur
Index	2345 h
Nom	SSDO_ArrayDummyParameter_AU16
Type d'objet	ARRAY (Matrice)
Type de données	UNSIGNED16 (NONSIGNE16)
Catégorie	M

7.5.2.13 Définition des sous-index

7.5.2.13.1 Généralités

Les types d'objets composés (objets ARRAY, RECORD) doivent être constitués d'un nombre d'éléments de données allant jusqu'à 256. Chaque élément de données doit être identifié par un sous-index de type UNSIGNED8.

Sous-index doit être interprété comme spécifié dans le Tableau 56.

Tableau 56 – Interprétation des sous-index

Sous-index	Nom
00h	NumberOfEntries
01h à FDh	Données spécifiques à l'objet
FEh	ChecksumOfObject
FFh	StructureOfObject

7.5.2.13.2 Sous-index 00h – NumberOfEntries (nombre d'entrées)

NumberOfEntries doit décrire le sous-index disponible le plus élevé, non compris FEh et FFh. Cette entrée doit être codée en UNSIGNED8, quel que soit le type de l'objet. Si l'objet existe, NumberOfEntries doit être obligatoire. Le type de données et la catégorie ne doivent pas être représentés dans les descriptions NumberOfEntries.

Dans la présente partie, NumberOfEntries est spécifié sous la forme indiquée dans le Tableau 57. Les éléments décrits par <...> doivent être établis en fonction du contexte spécifique.

Tableau 57 – Spécification du sous-index NumberOfEntries

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries
Plage de valeurs	<valeurs valides, par exemple 1 à 15>
Valeur par défaut	<Valeur par défaut, par exemple 15>
Accès	<Accès, par exemple rw>
Mise en correspondance SPDO	Non

Pour les matrices (ARRAY), NumberOfEntries doit exprimer le nombre d'éléments occupés dans une liste (l'accès doit être rw). Pour les enregistrements (RECORD), NumberOfEntries doit être constant (l'accès doit être ro ou Const).

7.5.2.13.3 Sous-index 01h - FDh – Données spécifiques à l'objet

Les sous-index compris entre 01h et FDh doivent contenir les données utiles de l'objet. L'index accessible le plus élevé doit être donné par NumberOfEntries.

Dans la présente partie, les données spécifiques à l'objet pour des objets de type RECORD sont spécifiées sous la forme indiquée dans le Tableau 58. Les éléments décrits par <...> doivent être établis en fonction du contexte spécifique.

Tableau 58 – Spécification du sous-index d'objet de type RECORD

Élément	Valeur
Sous-index	<sous-index en notation hexadécimale, par exemple 01h>
Nom	<nom>
Type d'objet	
Type de données	<type de données, par exemple UNSIGNED8>
Plage de valeurs	<valeurs valides, par exemple 1 à 255>
Valeur par défaut	<Valeur par défaut, par exemple 1>
Catégorie	<Obligatoire (M) ou facultatif (O), par exemple M>
Accès	<Accès, par exemple rw>
Mise en correspondance SPDO	<mise en correspondance possible, par exemple Opt>

Le nom doit être constitué d'un texte descriptif suivi d'un suffixe de type de données. Le paragraphe 7.5.2.4 doit s'appliquer à la seule exception que le préfixe du domaine est omis.

Si une entrée Nom est définie dans une définition de type de données, le sous-index Nom doit être égal à cette entrée.

La catégorie doit renvoyer au sous-index correspondant uniquement. Obligatoire (M) doit indiquer que le sous-index doit être appliqué si l'objet est utilisé. Un sous-index obligatoire ne doit pas forcer l'objet parent à être obligatoire.

Dans la présente partie, les données spécifiques à l'objet des objets de type ARRAY sont spécifiées sous la forme indiquée dans le Tableau 59. Les éléments décrits par <...> doivent être établis en fonction du contexte spécifique.

Tableau 59 – Spécification du sous-index d'objet de type ARRAY

Élément	Valeur
Sous-index	<sous-index en notation hexadécimale, par exemple 01h>
Nom	<nom>
Plage de valeurs	<type de données, par exemple UNSIGNED16>
Valeur par défaut	<valeur par défaut, par exemple 0>
Catégorie	<Obligatoire (M) ou facultatif (O), par exemple M>
Accès	<Accès, par exemple rw>
Mise en correspondance SPDO	<mise en correspondance possible, par exemple Opt>

Le nom doit être un texte descriptif. Le paragraphe 7.5.2.4 doit s'appliquer à la seule exception que le préfixe du domaine et le suffixe du type de données sont omis. Si une entrée Nom est définie dans une définition de type de données, le sous-index Nom doit être égal à cette entrée.

La catégorie doit renvoyer au sous-index correspondant uniquement. Obligatoire (M) doit indiquer que le sous-index doit être appliqué si l'objet est utilisé. Un sous-index obligatoire ne doit pas forcer l'objet parent à être obligatoire.

NOTE En dépit de l'équivalence fonctionnelle de tous les sous-index dans une matrice, il peut être défini plusieurs entrées de sous-index pour un objet particulier afin de montrer les différences des options Catégorie, Accès et / ou Mise en correspondance SPDO des sous-index.

7.5.2.13.4 Sous-index FEh – ChecksumOfObject (somme de contrôle de l'objet)

Le sous-index FEh doit contenir la somme de contrôle des données dans l'objet parent. Quel que soit le type d'objet parent, il doit s'agir d'un CRC-8 et il doit être codé comme UNSIGNED8. Pour construire le CRC-8, toutes les données qui peuvent être écrites à partir du sous-index 00h-FDh doivent être utilisées (les données en lecture seule ne doivent pas être prises en compte car elles ne peuvent pas être modifiées par le SCM). Le polynôme de calcul du CRC-8 doit être le même que celui utilisé pour calculer le CRC-8 des parties une et deux de la PDU de sécurité (voir 7.1.7).

Les objets qui ne sont pas pertinents pour le calcul de la somme de contrôle du SOD doivent avoir leur sous-index FEh mis à zéro.

EXEMPLE L'index 1001h n'est pas significatif pour la somme de contrôle SOD car il ne contient aucun paramètre configurable par le SCM.

7.5.2.13.5 Sous-index FFh – StructureOfObject (structure d'objet)

Le sous-index FFh peut représenter la structure de l'objet en fournissant le type de données et le type d'objet. Quel que soit le type d'objet, il est codé UNSIGNED32 (NONSIGNÉ32).

Le codage de StructureOfObject est spécifié dans le Tableau 60.

Tableau 60 – Codage de StructureOfObject

	MSB [a]			LSB [b]
Numéro de bit	31 à 24	23 à 16	15 à 8	7 à 0
Valeur	00h	Type de données (se reporter au Tableau 61) (Octet de poids fort))	(Octet de poids faible)	Type d'objet (se reporter au Tableau 51))
[a] Octet de poids fort. [b] Octet de poids faible.				

NOTE La prise en charge du sous-index FFh est facultative. Si elle est utilisée dans l'ensemble du dictionnaire d'objets et si la structure des types de données composés est également fournie, il est possible de télécharger l'ensemble de la structure du dictionnaire d'objets.

Si StructureOfEntry n'est pas pris en charge, le sous-index FFh doit être réservé.

StructureOfObject n'est pas présenté dans les définitions d'objets de la présente spécification.

7.5.3 Spécification de l'entrée type de données

7.5.3.1 Généralités

Les types de données de base doivent être placés dans le dictionnaire d'objets à des fins de définition uniquement. Cependant, les index de plage 0001h à 0007h peuvent être mis en correspondance afin de définir l'espace approprié du RxSPDO (mise en correspondance factice) comme n'étant pas utilisé par le dispositif (sans importance). Les index 0009h à 000Bh et 000Fh ne doivent pas être mis en correspondance avec un SPDO.

La plage d'index 0001h à 04FFh doit être utilisée pour des entrées du dictionnaire d'objets spécifiant le type de données.

Le Tableau 61 définit les types de données utilisés dans le SOD.

Tableau 61 – Types de données de dictionnaire d'objets

Index	Objet	Nom
0001h	DEFTYPE	BOOLEAN
0002h	DEFTYPE	INTEGER8
0003h	DEFTYPE	INTEGER16
0004h	DEFTYPE	INTEGER32
0005h	DEFTYPE	UNSIGNED8
0006h	DEFTYPE	UNSIGNED16
0007h	DEFTYPE	UNSIGNED32
0008h	DEFTYPE	REAL32
0009h	DEFTYPE	VISIBLE_STRING
000Ah	DEFTYPE	OCTET_STRING
000Bh	DEFTYPE	UNICODE_STRING
000Ch	réservé	
000Dh	réservé	
000Eh	réservé	
000Fh	DEFTYPE	DOMAIN
0010h	DEFTYPE	INTEGER24
0011h	DEFTYPE	REAL64
0012h	DEFTYPE	INTEGER40
0013h	DEFTYPE	INTEGER48
0014h	DEFTYPE	INTEGER56
0015h	DEFTYPE	INTEGER64
0016h	DEFTYPE	UNSIGNED24
0017h	réservé	

Index	Objet	Nom
0018h	DEFTYPE	UNSIGNED40
0019h	DEFTYPE	UNSIGNED48
001Ah	DEFTYPE	UNSIGNED56
001Bh	DEFTYPE	UNSIGNED64
001Ch -- 001Fh	réservé	
0020h	DEFSTRUCT	SSDO_LifeGuardRecord_TYPE
0021h	DEFSTRUCT	SINF_DevVendorInfoRecord_TYPE
0022h	DEFSTRUCT	SCOM_CommonCommParamRecord_TYPE
0023h	DEFSTRUCT	SSDO_CommParamRecord_TYPE
0024h	DEFSTRUCT	SNMT_CommParamRecord_TYPE
0025h	DEFSTRUCT	SPDO_RxCommParamRecord_TYPE
0026h	DEFSTRUCT	SPDO_TxCommParamRecord_TYPE
0027h	DEFSTRUCT	SSCM_SADRDVIListRecord_TYPE
0028h	DEFSTRUCT	SSCM_UDIDSADRListRecord_TYPE
0029h	DEFSTRUCT	SSCM_SpecificParametersRecord_TYPE
002Ah -- 003Fh	réservé	

7.5.3.2 Types de données de base

Pour les types de données de base (Type d'objet = DEFTYPE) voir 5.5.2.

Un dispositif peut, de manière facultative, fournir la longueur des types de données de base codée comme UNSIGNED32 pour accès en lecture de l'index qui fait référence au type de données.

EXEMPLE L'index 0007h (UNSIGNED32) contient la valeur 20h=32d car le type de données UNSIGNED32 est codé au moyen d'une séquence binaire de 32 bits. Si la longueur est variable (par exemple 000Fh = Domain), l'entrée contient 0h.

7.5.3.3 Types de données composés

Pour les types de données composés pris en charge, un dispositif peut, en option, fournir la structure de ce type de données pour l'accès en lecture à l'index de type de données correspondant. Le sous-index 0 donne ensuite le nombre d'entrées à cet index, sans compter les sous-index 0, 254 et 255; le sous-index suivant contient le type de données conformément au Tableau 61, codé comme UNSIGNED16.

L'entrée à l'index 0021h décrivant la structure de l'objet identité SINF_DevVendorInfoRecord_TYPE est spécifié dans le Tableau 62 et dans le Tableau 63.

Tableau 62 – Description du type de données composé 0021h

Élément	Valeur
Index	0021 h
Nom	SINF_DevVendorInfoRecord_TYPE
Type d'objet	DEFSTRUCT

Tableau 63 – Description des sous-index composés 0021h

Sous-index	Nom du composant	Valeur	Type de Données
00h	NumberOfEntries	06h	UNSIGNED16
01h	VendorId_U32	07h	UNSIGNED32
02h	ProductCode_U32	07h	UNSIGNED32
03h	RevisionNo_U32	07h	UNSIGNED32
04h	SerialNo_U32	07h	UNSIGNED32
05h	FirmWareChecksum_U32	07h	UNSIGNED32

Sous-index	Nom du composant	Valeur	Type de Données
06h	ParameterChecksum_U16	06h	UNSIGNED16
07h	ParameterTimestamp_U32	07h	UNSIGNED32

On doit faire la distinction entre les types de données spécifiques au fabricant qui sont normalisés (simples) et ceux qui sont composés en tentant de lire le sous-index 1h: Pour un type de données composé, le dispositif doit renvoyer une valeur et le sous-index 0h contient le numéro du sous-index suivant. Pour un type de données normalisé, le dispositif doit abandonner l'accès SOD car aucun sous-index 1h n'est disponible.

7.5.4 Description des objets

7.5.4.1 Généralités

La présente sous-classe ainsi que la suivante décrivent les entrées du dictionnaire d'objets normalisées pour un SN et un SCM. La mise en œuvre de ces objets peut être différente de la description donnée dans la spécification.

EXEMPLE La SADR principale pourrait être un paramètre en lecture seule si elle était établie au moyen d'un commutateur DIP.

Les objets normalisés FSCP 13/1 sont spécifiés du Tableau 64 au Tableau 72.

Tableau 64 – Objets normalisés

Index	Type d'objet	Nom	Type de données	Acc	M/O
1001h	VAR	Registre d'erreurs	UNSIGNED32	RO	M
1002h	VAR	Registre d'états du fabricant	SINF_DevVendorInfoRecord_TYPE (21h)	RO	O
1003h	ARRAY	Champ d'erreurs prédéfini	OCTET_STRING	rw	O
100Ch	RECORD	Sauvegarde	DOMAIN	-	M
100Dh	VAR	Temps de rafraîchissement pour la réinitialisation de sauvegarde	SSCM_SpecificParametersRecord_TYPE (29h)	rw	M
1018h	RECORD	Informations de fournisseur de dispositifs	UNSIGNED32	-	M
1019h	VAR	Id unique de dispositif	SINF_DevVendorInfoRecord_TYPE (21h)	RO	M
101Ah	DOMAIN	Téléchargement aval de paramètres	OCTET_STRING	rw	O
101Bh	RECORD	Paramètres spécifiques au SCM	DOMAIN	-	O

Tableau 65 – Objets de communication communs

Index	Type d'objet	Nom	Type de Données	Acc	M/O
1200 h	RECORD	Paramètres de communication communs	SCOM_CommonCommParamRecord_TYPE (22h)	-	M
1201 h	RECORD	Paramètres de communication SSDO	SSDO_CommParamRecord_TYPE (23h)	-	C[a]
1202 h	RECORD	Paramètres de communication SNMT	SNMT_CommParamRecord_TYPE (24h)	-	C
[a] Les objets doivent être uniquement disponibles si le nœud peut être un client SSDO.					

Tableau 66 – Objets de communication de SPDO de réception

Index	Type d'objet	Nom	Type de Données	Acc	M/O
1400h	RECORD	Paramètres de communication du 1er RxSPDO	SPDO_RxCommParamRecord_TYPE (25h)	rw	M[a]
1401 h	RECORD	Paramètres de communication du 2nd RxSPDO	SPDO_RxCommParamRecord_TYPE (25h)	rw	M[a]
:::	:::	:::	:::	:::	:::
17FEh	RECORD	Paramètres de communication du 1 023ème RxSPDO	SPDO_RxCommParamRecord_TYPE (25h)	rw	M[a]
[a] Si un dispositif prend en charge des SPDO, les entrées de paramètres de communication SPDO et de mise en correspondance des SPDO correspondants dans le dictionnaire d'objets sont obligatoires.					

Tableau 67 – Objets de mise en correspondance de SPDO de réception

Index	Type d'objet	Nom	Type de Données	Acc	M/O
1800h	ARRAY	Paramètres de mise en correspondance du 1er RxSPDO	UNSIGNED32	rw	M[a]
1801h	ARRAY	Paramètres de mise en correspondance du 2nd RxSPDO	UNSIGNED32	rw	M[a]
:::	:::	:::	:::	:::	:::
1BFEh	ARRAY	Paramètres de mise en correspondance du 1 023ème RxSPDO	UNSIGNED32	rw	M[a]
[a] Si un dispositif prend en charge des SPDO, les entrées de paramètres de communication SPDO et de mise en correspondance des SPDO correspondants dans le dictionnaire d'objets sont obligatoires.					

Tableau 68 – Objets de communication de SPDO d'émission

Index	Type d'objet	Nom	Type de Données	Acc	M/O
1C00h	VAR	Paramètres de communication du 1er TxSPDO	SPDO_TxCommParamRecord_TYPE (26h)	rw	M[a]
1C01h	VAR	Paramètres de communication du 2nd TxSPDO	SPDO_TxCommParamRecord_TYPE (26h)	rw	M[a]
:::	:::	:::	:::	:::	:::
1FFEh	VAR	Paramètres de communication du 1 023ème TxSPDO	SPDO_TxCommParamRecord_TYPE (26h)	rw	M[a]
[a] Si un dispositif prend en charge des SPDO, les entrées de paramètres de communication SPDO et de mise en correspondance des SPDO correspondants dans le dictionnaire d'objets sont obligatoires.					

Tableau 69 – Objets de mise en correspondance de SPDO d'émission

Index	Type d'objet	Nom	Type de Données	Acc	M/O
C000h	ARRAY	Paramètres de mise en	UNSIGNED32	rw	M[a]

Index	Type d'objet	Nom	Type de Données	Acc	M/O
		correspondance du 1er TxSPDO			
C001h	ARRAY	Paramètres de mise en correspondance du 2nd TxSPDO	UNSIGNED32	rw	M[a]
:::	:::	:::	:::	:::	:::
C3FEh	ARRAY	Paramètres de mise en correspondance du 1 023ème TxSPDO	UNSIGNED32	rw	M[a]
[a] Si un dispositif prend en charge des SPDO, les entrées de paramètres de communication SPDO et de mise en correspondance des SPDO correspondants dans le dictionnaire d'objets sont obligatoires.					

Tableau 70 – Liste de DVI - SADR

Index	Type d'objet	Nom	Type de Données	Acc	M/O
C400h	RECORD	1ère liste de DVI - SADR	SSCM_SADRDVIListRecord_TYPE (27h)	rw	C[a]
C401h	RECORD	2nde liste de DVI - SADR	SSCM_SADRDVIListRecord_TYPE (27h)	rw	C[a]
:::	:::	:::	:::	:::	:::
C7FEh	RECORD	1 023ème liste de DVI - SADR	SSCM_SADRDVIListRecord_TYPE (27h)	rw	C[a]
[a] Elles ne sont nécessaires que si le dispositif a une fonctionnalité de SCM.					

Tableau 71 – Liste de SADR supplémentaires

Index	Type d'objet	Nom	Type de Données	Acc	M/O
C801h	ARRAY	1ère SADR	UNSIGNED16	rw	C[a]
C802h	ARRAY	2nde SADR	UNSIGNED16	rw	C[a]
:::	:::	:::	:::	:::	:::
CBFFh	ARRAY	1 023ème SADR	UNSIGNED16	rw	C[a]
[a] Elles ne sont nécessaires que si le dispositif a une fonctionnalité de SCM.					

Tableau 72 – Liste d'UDID - SADR

Index	Type d'objet	Nom	Type de Données	Acc	M/O
CC01h	RECORD	1ère liste d'UDID - SADR	SSCM_UDIDSADRListRecord_TYPE (28h)	rw	C[a]
CC02h	RECORD	2nde liste d'UDID - SADR	SSCM_UDIDSADRListRecord_TYPE (28h)	rw	C[a]
:::	:::	:::	:::	:::	:::
CFFFh	RECORD	1 023ème liste d'UDID - SADR	SSCM_UDIDSADRListRecord_TYPE (28h)	rw	C[a]
[a] Elles ne sont nécessaires que si le dispositif a une fonctionnalité de SCM.					

7.5.4.2 Objet 1001h: Registre d'erreurs

Cet objet doit être un registre d'erreurs pour le dispositif. Le dispositif peut mettre en correspondance des erreurs internes dans cet octet. Si la valeur de l'objet est zéro, le dispositif doit être dans un état correct, toutes les autres valeurs doivent signaler que le dispositif est dans un état d'erreur.

L'objet 1001h: registre d'erreurs, est spécifié dans le Tableau 73.

Tableau 73 – Objet 1001h: Registre d'erreurs

Élément	Valeur
Index	1001 h
Nom	SERR_GeneralError_U8
Type d'objet	VAR
Type de données	UNSIGNED8
Plage de valeurs	UNSIGNED8
Valeur par défaut	0
Catégorie	M
Accès	RO
Mise en correspondance SPDO	Oui
Pertinent pour la somme de contrôle SOD	Non

L'interprétation des valeurs pour l'objet 1001h: registre d'erreurs, est spécifiée dans le Tableau 74.

Tableau 74 – Interprétation des valeurs de l'objet 1001h: registre d'erreurs

Bit	M/O	Description
0	M	erreur générique
1	O	courant
2	O	tension
3	O	température
4	O	erreur de communication (dépassement, état d'erreur)
5	O	spécifique au profil de dispositif
6	O	Réservé (toujours à 0)
7	O	spécifique au fabricant

7.5.4.3 Objet 1002h: Registre d'états du fabricant

Cet objet doit être un registre d'états destiné au fabricant. Dans la présente partie, seule la taille et l'emplacement de cet objet doivent être définis.

L'objet 1002h: Registre d'états du fabricant, est spécifié dans le Tableau 75.

Tableau 75 – Objet 1002h: Registre d'états du fabricant

Élément	Valeur
Index	1002h
Nom	SERR_ManufacturerStatus_U32

Élément	Valeur
Type d'objet	VAR
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	O
Accès	ro
Mise en correspondance SPDO	oui
Pertinent pour la somme de contrôle SOD	Non

7.5.4.4 Objet 1003h: Champ d'erreurs prédéfini

L'objet à l'index 1003h doit conserver les erreurs qui ont eu lieu sur le dispositif. Chaque nouvelle erreur doit être stockée au sous-index 01h et les erreurs plus anciennes doivent être déplacées vers le sous-index immédiatement supérieur. De ce fait, il fournit un historique des erreurs. Le sous-index 0 doit fournir le nombre d'erreurs dans la liste. La mise à zéro du sous-index 0 doit être utilisée pour effacer la liste.

L'objet 1003h Champ d'erreurs prédéfini est spécifié du Tableau 76 au Tableau 79.

Tableau 76 – Objet 1003h: Champ d'erreurs prédéfini

Élément	Valeur
Index	1003h
Nom	SERR_ErrorHistory_AU32
Type d'objet	ARRAY
Type de données	UNSIGNED32
Catégorie	O
Pertinent pour la somme de contrôle SOD	Non

Tableau 77 – Objet 1003h sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	1 à 253
Valeur par défaut	1
Accès	rw
Mise en correspondance SPDO	Non
[a] Nombre d'erreurs.	

Tableau 78 – Objet 1003h sous-index 01h

Élément	Valeur
Sous-index	01 h
Nom	Error_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M

Élément	Valeur
Accès	RO
Mise en correspondance SPDO	Non
[a] Erreur dans la liste historique.	

Tableau 79 – Objet 1003h sous-index 02h à FDh

Élément	Valeur
Sous-index	02h -- FDh
Nom	Error_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	O
Accès	ro
Mise en correspondance SPDO	Non
[a] Erreur dans la liste historique.	

7.5.4.5 Objet 100Ch: Sauvegarde

Cet objet définit le protocole de sauvegarde. L'intervalle de sauvegarde doit être l'intervalle de temps au cours duquel il convient que le SN reçoive un signal de durée de vie du SCM. Le facteur de durée de vie multiplié par l'intervalle de sauvegarde donne la durée de vie du nœud. L'intervalle de temps pour l'envoi du signal de durée de vie aux SN doit être calculé en divisant le GuardTime_U32 (Intervalle de sauvegarde U32) par le nombre de nœuds. L'intervalle de sauvegarde pour un SN dans le SCM commence après réception de la dernière réponse de sauvegarde du SN.

L'objet 100Ch: Sauvegarde est spécifié du Tableau 80 au Tableau 83.

Tableau 80 – Objet 100Ch: Sauvegarde

Élément	Valeur
Index	100Ch
Nom	SSDO_LifeGuard_REC
Type d'objet	RECORD
Type de données	SSDO_LifeGuardRecord_TYPE
Catégorie	M
Pertinent pour la somme de contrôle SOD	oui

Tableau 81 – Objet 100Ch sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	2
Valeur par défaut	2
Accès	ro
Mise en correspondance SPDO	No

Élément	Valeur
[a] Nombre d'entrées dans cet index.	

Tableau 82 – Objet 100Ch sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	GuardTime_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	10 000 000
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Intervalle de sécurité pour le protocole de durée de vie. La base de temps de ce paramètre est définie par CommonCommParam_REC. ConsecutiveTimeBase_I8. La plage des valeurs peut être limitée à une valeur inférieure à 232. Cette limite est spécifique au fournisseur et/ou due à des raisons de mise en œuvre.	

Tableau 83 – Objet 100Ch Sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	LifeTimeFactor_U8
Type de données	UNSIGNED8
Plage de valeurs	1 – 255
Valeur par défaut	2
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Nombre d'expirations de GuardTime_U32 avant que le nœud ne repasse de l'état opérationnel à l'état pré-opérationnel.	

7.5.4.6 Objet 100Dh: Intervalle de rafraîchissement de la sécurité de réinitialisation

Cet objet contient l'intervalle de rafraîchissement de la réinitialisation de sauvegarde (voir 7.7.6.3).

L'objet 100Dh: Intervalle de rafraîchissement de la sécurité de réinitialisation est spécifié dans le Tableau 84.

Tableau 84 – Objet 100Dh: Intervalle de rafraîchissement de la sécurité de réinitialisation

Élément	Valeur
Index	100Dh
Nom	SNMT_ResetGuarding_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	100 000

Élément	Valeur
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	No
[a] Le sous-index donne le temps de rafraîchissement pour le service SNMT_SN_reset_guarding_SCM. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

7.5.4.7 Objet 1018h: Informations de fournisseur de dispositif

Cet objet doit contenir les informations de dispositif spécifiques au fournisseur. Il doit également définir des dispositifs compatibles. Tout dispositif ayant un ID fournisseur identique, un code produit, un numéro de révision majeure et un numéro de révision mineure supérieurs ou égaux à ceux d'un autre dispositif, doit être compatible. Le fabricant d'un dispositif doit vérifier et garantir la compatibilité des modifications de révisions mineures pour un dispositif sécuritaire.

L'objet 1018h: Informations de fournisseur de dispositif est spécifié du Tableau 85 au Tableau 93.

Tableau 85 – Objet 1018h: Informations de fournisseur de dispositif

Élément	Valeur
Index	1018h
Nom	SINF_DevVendorInfo_REC
Type d'objet	RECORD
Type de données	SINF_DevVendorInfoRecord_TYPE
Catégorie	M
Pertinent pour la somme de contrôle SOD	Non

Tableau 86 – Objet 1018h sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	7
Valeur par défaut	7
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 87 – Objet 1018h sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	VendorID_32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0

Élément	Valeur
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] ID fournisseur selon la Liste de Fournisseurs EPSG.	

Tableau 88 – Objet 1018h sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	ProductCode_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Le code produit spécifique au fabricant identifie une version de dispositif spécifique.	

Tableau 89 – Objet 1018h sous-index 03h

Élément	Valeur
Sous-index	03h
Nom	RevisionNumber_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Voir Tableau 94.	

Tableau 90 – Objet 1018h sous-index 04h

Élément	Valeur
Sous-index	04h
Nom	SerialNumber_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Numéro de série du dispositif.	

Tableau 91 – Objet 1018h sous-index 05h

Élément	Valeur
Sous-index	05h
Nom	FirmWareChecksum_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Somme de contrôle utilisée pour vérifier l'intégrité du microprogramme enregistré dans le dispositif. Le calcul est spécifique au fournisseur.	

Tableau 92 – Objet 1018h sous-index 06h

Élément	Valeur
Sous-index	06h
Nom	ParameterChecksum_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	UNSIGNED16
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Cette somme de contrôle est un CRC-16 (qui utilise le même polynôme que pour les parties une et deux de la PDU de sécurité, voir 7.1.7) construit à partir des sommes de contrôle des objets disponibles (sous-index FEh). La somme de contrôle doit être vérifiée dans chaque dispositif au démarrage. Les sommes de contrôle ayant des index marqués comme non pertinents pour la somme de contrôle du SOD doivent être omises. Si ce paramètre est lu et si le SN est à l'état pré-opérationnel, la somme de contrôle doit être recalculée avant génération de la réponse.	

Tableau 93 – Objet 1018h sous-index 07h

Élément	Valeur
Sous-index	07h
Nom	ParameterTimestamp_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] L'horodatage de ce paramètre est généré par l'outil de configuration et le SCM le vérifie au démarrage dans chaque dispositif (voir 7.7.12.2).	

Le numéro de révision spécifique au fabricant doit être constitué d'un numéro de révision majeure et d'un numéro de révision mineure. Le numéro de révision majeure doit identifier le comportement de dispositif spécifique. Si la fonctionnalité du dispositif est étendue, le numéro

de révision majeure doit être incrémenté. Le numéro de révision mineure doit identifier différentes versions ayant le même comportement de dispositif.

Le Tableau 94 définit le codage de RevisionNumber_U32.

Tableau 94 – Structure du numéro de révision

31	16	15	0
Numéro de révision majeure		Numéro de révision mineure	
MSB			LSB

7.5.4.8 Objet 1019h: Id unique de dispositif

L'objet d'index 1019h doit contenir l'identifiant unique de dispositif (UDID). L'UDID doit être constitué de l'ID du fournisseur MAC et de l'ID unique de dispositif du fournisseur.

NOTE Le SN vérifie l'unicité de l'UDID des SN connectés SN et de ce fait, l'attribution de l'UDID en cours de fabrication par le fournisseur n'est pas une étape de production critique du point de vue de la sécurité.

La composition de l'UDID doit être la suivante:

3 octets d'ID fournisseur de MAC + 3 octets de numéro unique de dispositif, spécifique au fournisseur (par exemple le numéro de série).

EXEMPLE ID fournisseur de MAC: 06065h et numéro de série: 471112h donne l'UDID: 00-60-65-47-11-12.

L'UDID 00-00-00-00-00-00 doit être réservé.

L'objet 1019h: ID unique de dispositif est spécifié dans le Tableau 95.

Tableau 95 – Objet 1019h: Id unique de dispositif

Élément	Valeur
Index	1019h
Nom	SINF_UniqueDeviceID_OSTR6 [a]
Type d'objet	OCTET_STRING6
Type de données	OCTET_STRING6
Plage de valeurs	
Valeur par défaut	0
Catégorie	M
Accès	Const
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	Non
[a] L'UDID doit être un identifiant globalement unique composé de 6 octets.	

7.5.4.9 Objet 101Ah: Téléchargement aval de paramètres

Cet objet doit être utilisé pour recevoir un paramètre spécifique à l'application configuré à partir du SCM.

L'objet 101Ah: téléchargement aval de paramètres est spécifié dans le Tableau 96.

Tableau 96 – Objet 101Ah: Téléchargement aval de paramètres

Élément	Valeur

Élément	Valeur
Index	101Ah
Nom	SSDO_ParameterDownload_DOM
Type de données	DOMAIN
Plage de valeurs	DOMAIN
Valeur par défaut	0
Catégorie	O
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	Non

7.5.4.10 Objet 101Bh: Paramètres du SCM

L'objet d'index $101B_h$ doit contenir les paramètres spécifiques au SCM.

L'objet 101Bh: Paramètres du SCM est spécifié dans le Tableau 97, le Tableau 98 et le Tableau 99.

Tableau 97 – Objet 101Bh: Paramètres du SCM

Élément	Valeur
Index	101Bh
Nom	SSCM_SpecificParameters_REC
Type d'objet	RECORD
Type de données	SSCM_SpecificParametersRecord_TYPE
Catégorie	O
Mise en correspondance SPDO	oui

Tableau 98 – Objet 101Bh sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	1
Valeur par défaut	1
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 99 – Objet 101Bh sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	ConfigurationMode_U8
Type de données	UNSIGNED8
Plage de valeurs	0 -- 1
Valeur par défaut	0

Élément	Valeur
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Mode de configuration du SCM selon 6.2.2, 0 = ACM, 1 = MCM.	

7.5.4.11 Objet 1200h: Paramètre de communication commun

Cet objet doit contenir les paramètres de communication communs pour le dispositif.

Le sous-index 3 doit contenir la base de temps consécutifs pour le module. Il n'est pas nécessaire qu'un dispositif prenne en charge toutes les bases de temps possibles. Au minimum une base de temps de 100 µs (type 2) doit être prise en charge

L'objet 1200h: Paramètre de communication commun est spécifié du Tableau 100 au Tableau 105.

Tableau 100 – Objet 1200h: Paramètre de communication commun

Élément	Valeur
Index	1200h
Nom	SCOM_CommonCommParam_REC
Type d'objet	RECORD
Type de données	SCOM_CommonCommParamRecord_TYPE
Catégorie	M
Pertinent pour la somme de contrôle SOD	oui

Tableau 101 – Objet 1200h sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	3
Valeur par défaut	3
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 102 – Objet 1200h sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	SDN_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	0 -- 1 023
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Numéro de domaine de sécurité du dispositif.	

Tableau 103 – Objet 1200h sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	SADR_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	0 -- 1 023
Valeur par défaut	0
Catégorie	M
Accès	ro
Mise en correspondance SPDO	Non
[a] Adresse de Sécurité du SCM au sein du SD.	

Tableau 104 – Objet 1200h sous-index 03h

Élément	Valeur
Sous-index	03h
Nom	ConsecutiveTimeBase_I8 [a]
Type de données	INTEGER8
Plage de valeurs	0 -- 3
Valeur par défaut	2
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Identification de la base de temps pour le champ CT, 0 = 1 µs, 1 = 10 µs, 2 = 100 µs, 3 = 1 000 µs.	

La base de temps du champ CT doit être calculée au moyen de l'Équation (1).

$$TimeBase = 1\,\mu s \times 10^{ConsecutiveTimeBase\_I8} \quad (1)$$

où

TimeBase est la base de temps du champ CT en µs;

ConsecutiveTimeBase_I8 est l'identifiant de la base de temps

Tableau 105 – Objet 1200h sous-index 04h

Élément	Valeur
Sous-index	04h
Nom	UDIDofSCM_ OSTR6 [a]
Type de données	OCTET_STRING6
Plage de valeurs	OCTET_STRING6
Valeur par défaut	UDID du propre SN
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] UDID du SCM.	

7.5.4.12 Objet 1201h: Paramètre de communication SSDO

Cet objet définit les paramètres de communication SSDO d'un client SSDO.

L'objet 1201h: Paramètre de communication SSDO est spécifié du Tableau 106 au Tableau 109.

Tableau 106 – Objet 1201h: Paramètre de communication SSDO

Élément	Valeur
Index	1201h
Nom	SSDO_CommParam_REC
Type d'objet	RECORD
Type de données	SSDO_CommParamRecord_TYPE
Catégorie	C
Pertinent pour la somme de contrôle SOD	oui

Tableau 107 – Objet 1201h sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	2
Valeur par défaut	2
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 108 – Objet 1201h sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	Timeout_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	1 -- 2^{32}

Élément	Valeur
Valeur par défaut	500 000
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Le sous-index donne le temps imparti pour un télégramme SSDO. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 109 – Objet 1201h sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	Retries_U8 [a]
Type de données	UNSIGNED8
Plage de valeurs	0 -- 255
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Ce sous-index indique le nombre de nouvelles tentatives pour un télégramme SSDO.	

7.5.4.13 Objet 1202h: Paramètre de communication SNMT

Cet objet définit les paramètres de communication SNMT d'un Maître SNMT.

L'objet 1202h: Paramètre de communication SNMT est spécifié du Tableau 110 au Tableau 113.

Tableau 110 – Objet 1202h: Paramètre de communication SNMT

Élément	Valeur
Index	1202h
Nom	SNMT_CommParam_REC
Type d'objet	RECORD
Type de données	SNMT_CommParamRecord_TYPE
Catégorie	C
Pertinent pour la somme de contrôle SOD	oui

Tableau 111 – Objet 1202h sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	2
Valeur par défaut	2
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 112 – Objet 1202h sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	Timeout_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	1 -- 2^{32}
Valeur par défaut	500 000
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Le sous-index fournit le temps imparti pour un télégramme SNMT. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 113 – Objet 1202h sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	Retries_U8 [a]
Type de données	UNSIGNED8
Plage de valeurs	0 -- 255
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Ce sous-index indique le nombre de nouvelles tentatives pour un télégramme SNMT.	

7.5.4.14 Objet 1400h à 17FEh: Paramètre de communication RxSPDO

Ces objets définissent de quelle SADR les données doivent être reçues.

Un producteur peut avoir plusieurs SADR (une pour chaque TxSPDO). Par conséquent un consommateur peut avoir plusieurs RxSPDO pour le même producteur physique.

L'objet 1400h à 17FEh: Paramètre de communication RxSPDO est spécifié du Tableau 114 au Tableau 127.

Tableau 114 – Objet 1400h à 17FEh: Paramètre de communication RxSPDO

Élément	Valeur
Index	1400h -- 17FEh
Nom	SPDO_RxCommParam_XXXh_REC
Type d'objet	RECORD
Type de données	SPDO_RxCommParamRecord_TYPE
Catégorie	O
Pertinent pour la somme de contrôle SOD	oui

Tableau 115 – Objet 1400h à 17FEh sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	8
Valeur par défaut	8
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 116 – Objet 1400h à 17FEh sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	SADR_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	0 -- 1 023
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Définit la SADR d'où les données RxSPDO doivent être reçues. La mise à zéro de SADR désactive le SPDO.	

Tableau 117 – Objet 1400h à 17FEh sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	SCT_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	1 -- 2^{32}
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Définit le temporisateur SCT pour les données provenant de ce RxSPDO (voir 7.7.1.2). La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 118 – Objet 1400h à 17FEh sous-index 03h

Élément	Valeur
Sous-index	03h
Nom	NumberOfConsecutiveTReq_U8 [a]
Type de données	UNSIGNED8
Plage de valeurs	1 -- 63
Valeur par défaut	1
Catégorie	M

Élément	Valeur
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Ce sous-index fournit le nombre de demandes de temps consécutifs.	

Tableau 119 – Objet 1400h à 17FEh sous-index 04h

Élément	Valeur
Sous-index	04h
Nom	TimeDelayTReq_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Ce sous-index fournit le délai entre deux ensembles de demandes de temps consécutifs. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 120 – Objet 1400h à 17FEh sous-index 05h

Élément	Valeur
Sous-index	05h
Nom	TimeDelaySync_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Ce sous-index fournit le délai entre une synchronisation temporelle réussie et la demande de temps suivante. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 121 – Objet 1400h à 17FEh sous-index 06h

Élément	Valeur
Sous-index	06h
Nom	MinTSyncPropagationDelay_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 65 535
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Le sous-index donne le délai de propagation minimal autorisé entre la demande de temps et la réponse de temps. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 122 – Objet 1400h à 17FEh sous-index 07h

Élément	Valeur
Sous-index	07h
Nom	MaxTSyncPropagationDelay_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 65 535
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Le sous-index donne le délai de propagation maximal autorisé entre la demande de temps et la réponse de temps. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 123 – Objet 1400h à 17FEh sous-index 08h

Élément	Valeur
Sous-index	08h
Nom	MinSPDOPropagationDelay_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 65 535
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Le sous-index donne le délai de propagation minimal autorisé entre émission et réception de la SPDO. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 124 – Objet 1400h à 17FEh sous-index 09h

Élément	Valeur
Sous-index	09h
Nom	MaxSPDOPropagationDelay_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 65 535
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Le sous-index donne le délai de propagation minimal autorisé entre émission et réception de la SPDO. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 125 – Objet 1400h à 17FEh sous-index 0Ah

Élément	Valeur
Sous-index	0Ah
Nom	BestCaseTResDelay_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	0 -- 65 535
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Le sous-index fournit le délai le plus favorable entre l'émission de la demande de temps et l'établissement de la réponse de temps. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 126 – Objet 1400h à 17FEh sous-index 0Bh

Élément	Valeur
Sous-index	0Bh
Nom	TimeRequestCycle_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	1 -- 2^{32}
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Le sous-index donne la durée de cycle de la demande de temps. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 127 – Objet 1400h à 17FEh sous-index 0Ch

Élément	Valeur
Sous-index	0Ch
Nom	TxSPDONo_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 1 023
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Contient le numéro TxSPDO avec lequel la demande de temps doit être envoyée.	

7.5.4.15 Objet 1800h à 1BFEh: Paramètre de mise en correspondance RxSPDO

Cet objet définit les paramètres de mise en correspondance RxSPDO. Les paramètres de mise en correspondance peuvent être prédéfinis par le fournisseur et en lecture seule. Une description détaillée de la mise en correspondance SPDO est fournie en 7.6.

L'objet 1800h à 1BFEh: Paramètre de mise en correspondance RxSPDO est spécifié du Tableau 128 au Tableau 131.

Tableau 128 – Objet 1800h à 1BFEh: Paramètre de communication RxSPDO

Élément	Valeur
Index	1800h -- 1BFEh
Nom	SPDO_RxMappParam_XXXh_AU32
Type d'objet	ARRAY
Type de données	UNSIGNED32
Catégorie	O
Pertinent pour la somme de contrôle SOD	oui

Tableau 129 – Objet 1800h à 1BFEh sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	0 -- 253
Valeur par défaut	0
Accès	rw / ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 130 – Objet 1800h à 1BFEh sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	SPDOMapping_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	rw / ro
Mise en correspondance SPDO	Non
[a] Le sous-index donne le paramètre de mise en correspondance pour ce TxSPDO.	

Tableau 131 – Objet 1800h à 1BFEh sous-index 02h à FDh

Élément	Valeur
Sous-index	02h -- FDh
Nom	SPDOMapping_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	O
Accès	rw / ro

Élément	Valeur
Mise en correspondance SPDO	Non
[a] Le sous-index donne le paramètre de mise en correspondance pour ce TxSPDO.	

7.5.4.16 Objet 1C00h à 1FFEh: Paramètre de communication TxSPDO

Chaque SN doit avoir au moins un TxSPDO, soit pour l'émission de données (producteur) soit pour la demande de temps (consommateur).

L'objet 1C00h à 1FFEh: Paramètre de communication TxSPDO est spécifié du Tableau 132 au Tableau 136.

Tableau 132 – Objet C00h à 1FFEh: Paramètre de communication TxSPDO

Élément	Valeur
Index	1C00h -- 1FFEh
Nom	SPDO_TxCommParam_XXXh_REC
Type d'objet	RECORD
Type de données	SPDO_TxCommParamRecord_TYPE
Catégorie	M/O
Pertinent pour la somme de contrôle SOD	oui

Tableau 133 – Objet 1C00h à 1FFEh sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	3
Valeur par défaut	3
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 134 – Objet 1C00h à 1FFEh sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	SADR_XXXh_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	0 -- 1 023
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Définit la SADR qui est utilisée lors de la diffusion des données du TxSPDO correspondant dans le réseau. La mise à zéro de SADR désactive le SPDO.	

Tableau 135 – Objet 1C00h à 1FFEh sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	RefreshPrescale_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 32 767
Valeur par défaut	1
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Le sous-index fournit le temps de rafraîchissement du producteur. La base de temps de ce paramètre est définie par CommonCommParam_REC.ConsecutiveTimeBase_I8.	

Tableau 136 – Objet 1C00h à 1FFEh sous-index 03h

Élément	Valeur
Sous-index	03h
Nom	NumberOfTRes_U8 [a]
Type de données	UNSIGNED8
Plage de valeurs	UNSIGNED8
Valeur par défaut	0
Catégorie	M/C
Accès	rw
Mise en correspondance SPDO	Non
[a] Définit le nombre de TRes consécutives pour une TReq reçue. Si le paramètre est à zéro, le TxSPDO peut uniquement être utilisé pour des télégrammes TReq (pour des dispositifs qui sont uniquement des consommateurs).	

7.5.4.17 Objet C000h à C3FEh: Paramètre de mise en correspondance TxSPDO

Cet objet définit les paramètres de mise en correspondance TxSPDO. Les paramètres de mise en correspondance peuvent être prédéfinis par le fournisseur et en lecture seule. Une description détaillée de la mise en correspondance SPDO est fournie en 7.6.

L'objet C000h à C3FEh: Paramètre de mise en correspondance TxSPDO est spécifié du Tableau 137 au Tableau 140.

Tableau 137 – Objet C000h à C3FEh: Paramètre de mise en correspondance TxSPDO

Élément	Valeur
Index	C000h -- C3FEh
Nom	SPDO_TxMappParam_XXXh_AU32
Type d'objet	ARRAY
Type de données	UNSIGNED32
Catégorie	M/O
Pertinent pour la somme de contrôle SOD	oui

Tableau 138 – Objet C000h à C3FEh sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	0 -- 253
Valeur par défaut	0
Accès	rw / ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 139 – Objet C000h à C3FEh sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	SPDOMapping_U32 [a]
Type d'objet	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	rw/ro
Mise en correspondance SPDO	Non
Pertinent pour la somme de contrôle SOD	
[a] Le sous-index donne le paramètre de mise en correspondance pour ce TxSPDO.	

Tableau 140 – Objet C000h à C3FEh sous-index 02h à FDh

Élément	Valeur
Sous-index	02h -- FDh
Nom	SPDOMapping_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	O
Accès	rw/ro
Mise en correspondance SPDO	Non
[a] Le sous-index donne le paramètre de mise en correspondance pour ce TxSPDO.	

7.5.4.18 Objet C400h à C7FEh: Liste de DVI - SADR

Cet objet doit être mis en œuvre si le SN a une fonctionnalité SCM. Il doit contenir la liste de tous les SN au sein du SD. La liste doit contenir les paramètres de chaque SN qui sont nécessaires à la vérification du SN lors du démarrage. Le nombre maximal de SN pris en charge doit être spécifique au fournisseur. Par conséquent, les deux premiers index (C400h, C401h) doivent être obligatoires (un SN pour le SCM et au moins un SN supplémentaire) et tous les autres doivent être facultatifs.

Objet C400h à C7FEh: Liste de DVI - SADR est spécifié du Tableau 141 au Tableau 153.

Tableau 141 – Objet C400h à C7FEh: Liste de DVI - SADR

Élément	Valeur
Index	C400h -- C7FEh
Nom	SSCM_SADRDVIList_XXXh_REC
Type d'objet	RECORD
Type de données	SSCM_SADRDVIListRecord_TYPE
Catégorie	C
Pertinent pour la somme de contrôle SOD	Non

Tableau 142 – Objet C000h à C3FEh sous-index 00h

Élément	Valeur
Sous-index	00h
Nom	NumberOfEntries [a]
Plage de valeurs	09h -- 0Bh
Valeur par défaut	9
Accès	ro
Mise en correspondance SPDO	Non
[a] Nombre d'entrées dans cet index.	

Tableau 143 – Objet C000h à C3FEh sous-index 01h

Élément	Valeur
Sous-index	01h
Nom	SADR_U16 [a]
Type de données	UNSIGNED16
Plage de valeurs	1 -- 1 023
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Contient la SADR prévue du SN correspondant.	

Tableau 144 – Objet C000h à C3FEh sous-index 02h

Élément	Valeur
Sous-index	02h
Nom	VendorId_U32 [a]
Type de données	UNSIGNED32
Plage de valeurs	UNSIGNED32
Valeur par défaut	0
Catégorie	M
Accès	rw
Mise en correspondance SPDO	Non
[a] Contient l'ID fournisseur du SN correspondant.	