IEC 62061

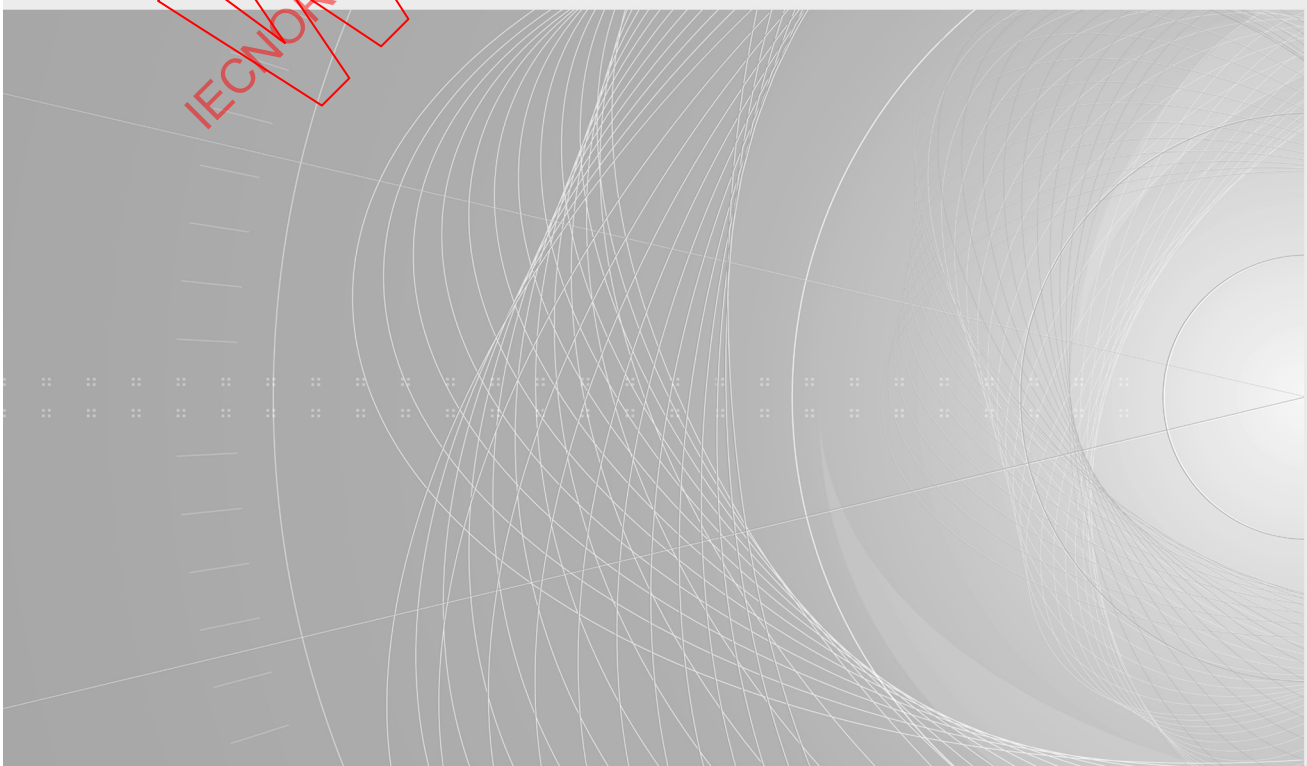
Edition 1.0 2005-01

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Safety of machinery – Functional safety of safety-related electrical, electronic and programmable electronic control systems

Sécurité des machines – Sécurité fonctionnelle des systèmes de commande électriques, électroniques et électroniques programmables relatifs à la sécurité



IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: 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.

- Catalogue of IEC publications: www.iec.ch/searchpub
The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.
- IEC Just Published: www.iec.ch/online_news/justpub
Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.
- Electropedia: www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.
- Customer Service Centre: www.iec.ch/webstore/custserv
If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:
Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00

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.

- Catalogue des publications de la CEI: www.iec.ch/searchpub/cur_fut-f.htm
Le Catalogue en-ligne de la CEI vous permet d'effectuer des recherches en utilisant différents critères (numéro de référence, texte, comité d'études,…). Il donne aussi des informations sur les projets et les publications retirées ou remplacées.
- Just Published CEI: www.iec.ch/online_news/justpub
Restez informé sur les nouvelles publications de la CEI. Just Published détaille deux fois par mois les nouvelles publications parues. Disponible en-ligne et aussi par email.
- Electropedia: www.electropedia.org
Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 20 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International en ligne.
- Service Clients: www.iec.ch/webstore/custserv/custserv_entry-f.htm
Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions, visitez le FAQ du Service clients ou contactez-nous:
Email: csc@iec.ch
Tél.: +41 22 919 02 11
Fax: +41 22 919 03 00

IEC 62061

Edition 1.0 2005-01

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Safety of machinery – Functional safety of safety-related electrical, electronic and programmable electronic control systems

Sécurité des machines – Sécurité fonctionnelle des systèmes de commande électriques, électroniques et électroniques programmables relatifs à la sécurité

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX XD

ICS 13.110; 25.040.99; 29.020

ISBN 2-8318-7818-7

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

SAFETY OF MACHINERY – FUNCTIONAL SAFETY OF SAFETY-RELATED ELECTRICAL, ELECTRONIC AND PROGRAMMABLE ELECTRONIC CONTROL SYSTEMS

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 provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62061 has been prepared by IEC technical committee 44: Safety of machinery – Electrotechnical aspects.

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

FDIS	Report on voting
44/460/FDIS	44/470/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.

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

The committee has decided that the contents of this publication will remain unchanged until the maintenance result 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.

The contents of the corrigenda of July 2005 and April 2008 have been included in this copy.

INTRODUCTION

As a result of automation, demand for increased production and reduced operator physical effort, Safety-Related Electrical Control Systems (referred to as SRECS) of machines play an increasing role in the achievement of overall machine safety. Furthermore, the SRECS themselves increasingly employ complex electronic technology.

Previously, in the absence of standards, there has been a reluctance to accept SRECS in safety-related functions for significant machine hazards because of uncertainty regarding the performance of such technology.

This International Standard is intended for use by machinery designers, control system manufacturers and integrators, and others involved in the specification, design and validation of a SRECS. It sets out an approach and provides requirements to achieve the necessary performance.

This standard is machine sector specific within the framework of IEC 61508. It is intended to facilitate the specification of the performance of safety-related electrical control systems in relation to the significant hazards (see 3.8 of ISO 12100-1) of machines.

This standard provides a machine sector specific framework for functional safety of a SRECS of machines. It only covers those aspects of the safety lifecycle that are related to safety requirements allocation through to safety validation. Requirements are provided for information for safe use of SRECS of machines that can also be relevant to later phases of the life of a SRECS.

There are many situations on machines where SRECS are employed as part of safety measures that have been provided to achieve risk reduction. A typical case is the use of an interlocking guard that, when it is opened to allow access to the danger zone, signals the electrical control system to stop hazardous machine operation. Also in automation, the electrical control system that is used to achieve correct operation of the machine process often contributes to safety by mitigating risks associated with hazards arising directly from control system failures. This standard gives a methodology and requirements to

- assign the required safety integrity level for each safety-related control function to be implemented by SRECS;
- enable the design of the SRECS appropriate to the assigned safety-related control function(s);
- integrate safety-related subsystems designed in accordance with ISO 13849 ;
- validate the SRECS.

This standard is intended to be used within the framework of systematic risk reduction described in ISO 12100-1 and in conjunction with risk assessment according to the principles described in ISO 14121 (EN 1050). A suggested methodology for safety integrity level (SIL) assignment is given in informative Annex A.

Measures are given to co-ordinate the performance of the SRECS with the intended risk reduction taking into account the probabilities and consequences of random or systematic faults within the electrical control system.

Figure 1 shows the relationship of this standard to other relevant standards.

Table 1 gives recommendations on the recommended application of this standard and the revision of ISO 13849-1.

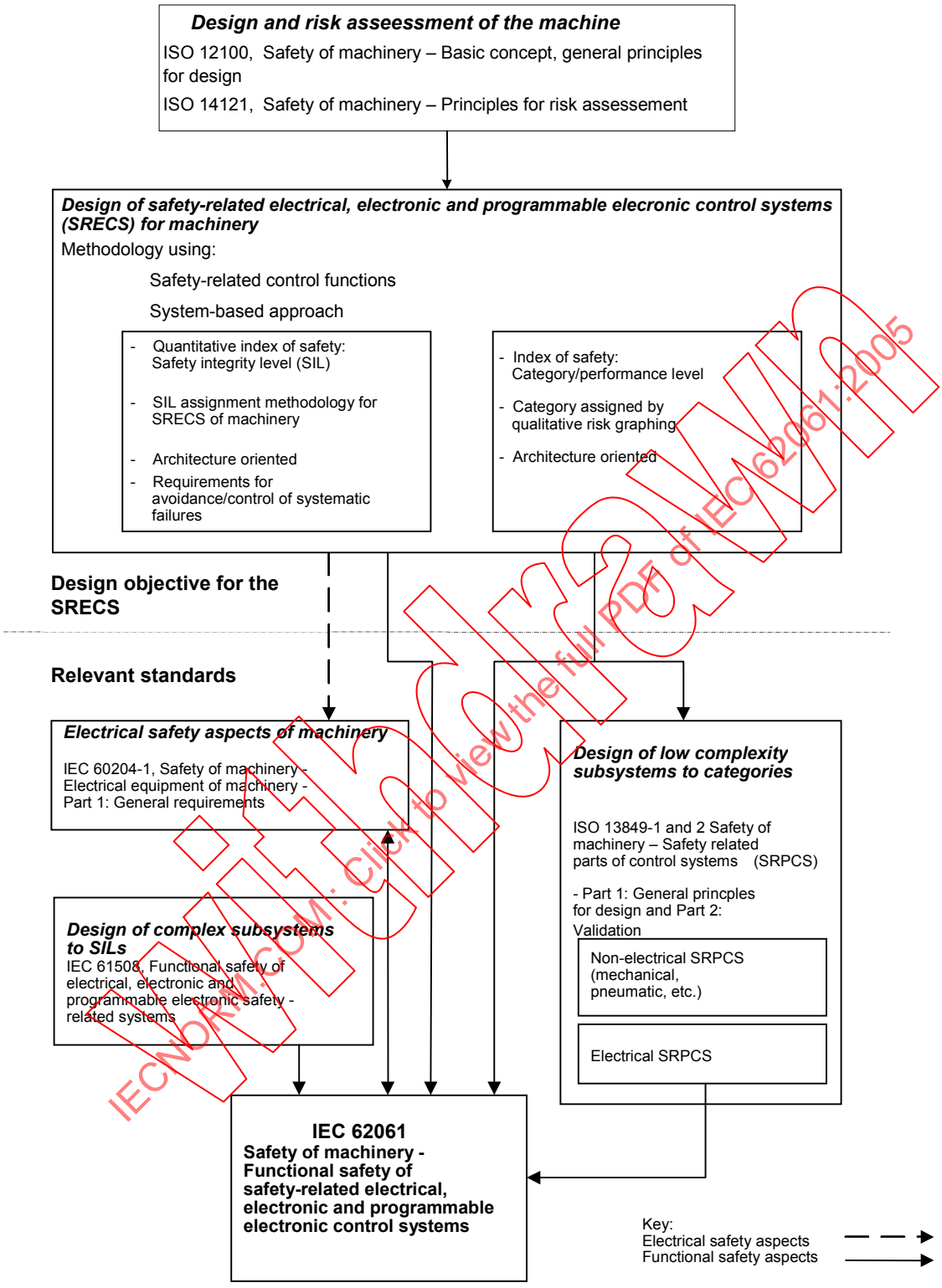


Figure 1 – Relationship of IEC 62061 to other relevant standards

Information on the recommended application of IEC 62061 and ISO 13849-1 (under revision)

IEC 62061 and ISO 13849-1 (under revision) specify requirements for the design and implementation of safety-related control systems of machinery. The use of either of these standards, in accordance with their scopes, can be presumed to fulfil the relevant essential safety requirements. Table 1 summarises the scopes of IEC 62061 and ISO 13849-1(under revision).

NOTE ISO 13849-1 is currently under preparation by ISO TC 199 and CEN TC 114.

Table 1 – Recommended application of IEC 62061 and ISO 13849-1(under revision)

	Technology implementing the safety-related control function(s)	ISO 13849-1 (under revision)	IEC 62061
A	Non electrical, e.g. hydraulics	X	Not covered
B	Electromechanical, e.g. relays, or non complex electronics	Restricted to designated architectures (see Note 1) and up to PL=e	All architectures and up to SIL 3
C	Complex electronics, e.g. programmable	Restricted to designated architectures (see Note 1) and up to PL=d	All architectures and up to SIL 3
D	A combined with B	Restricted to designated architectures (see Note 1) and up to PL=e	X see Note 3
E	C combined with B	Restricted to designated architectures (see Note 1) and up to PL=d	All architectures and up to SIL 3
F	C combined with A, or C combined with A and B	X see Note 2	X see Note 3

"X" indicates that this item is dealt with by the standard shown in the column heading.

NOTE 1 Designated architectures are defined in Annex B of EN ISO 13849-1(rev.) to give a simplified approach for quantification of performance level.

NOTE 2 For complex electronics: Use of designated architectures according to EN ISO 13849-1(rev.) up to PL=d or any architecture according to IEC 62061.

NOTE 3 For non-electrical technology use parts according to EN ISO 13849-1(rev.) as subsystems.

SAFETY OF MACHINERY – FUNCTIONAL SAFETY OF SAFETY-RELATED ELECTRICAL, ELECTRONIC AND PROGRAMMABLE ELECTRONIC CONTROL SYSTEMS

1 Scope

This International Standard specifies requirements and makes recommendations for the design, integration and validation of safety-related electrical, electronic and programmable electronic control systems (SRECS) for machines (see Notes 1 and 2). It is applicable to control systems used, either singly or in combination, to carry out safety-related control functions on machines that are not portable by hand while working, including a group of machines working together in a co-ordinated manner.

NOTE 1 In this standard, the term “electrical control systems” is used to stand for ”Electrical, Electronic and Programmable Electronic (E/E/PE) control systems” and “SRECS” is used to stand for “safety-related electrical, electronic and programmable electronic control systems”.

NOTE 2 In this standard, it is presumed that the design of complex programmable electronic subsystems or subsystem elements conforms to the relevant requirements of IEC 61508. This standard provides a methodology for the use, rather than development, of such subsystems and subsystem elements as part of a SRECS.

This standard is an application standard and is not intended to limit or inhibit technological advancement. It does not cover all the requirements (e.g. guarding, non-electrical interlocking or non-electrical control) that are needed or required by other standards or regulations in order to safeguard persons from hazards. Each type of machine has unique requirements to be satisfied to provide adequate safety.

This standard:

– is concerned only with functional safety requirements intended to reduce the risk of injury or damage to the health of persons in the immediate vicinity of the machine and those directly involved in the use of the machine;

– is restricted to risks arising directly from the hazards of the machine itself or from a group of machines working together in a co-ordinated manner;

NOTE 3 Requirements to mitigate risks arising from other hazards are provided in relevant sector standards. For example, where a machine(s) is part of a process activity, the machine electrical control system functional safety requirements should, in addition, satisfy other requirements (e.g. IEC 61511) insofar as safety of the process is concerned.

– does not specify requirements for the performance of non-electrical (e.g. hydraulic, pneumatic) control elements for machines;

NOTE 4 Although the requirements of this standard are specific to electrical control systems, the framework and methodology specified can be applicable to safety-related parts of control systems employing other technologies.

– does not cover electrical hazards arising from the electrical control equipment itself (e.g. electric shock – see IEC 60204–1).

The objectives of specific Clauses in IEC 62061 are as given in Table 2.

Table 2 – Overview and objectives of IEC 62061

Clause	Objective
4: Management of functional safety	To specify the management and technical activities which are necessary for the achievement of the required functional safety of the SRECS.
5: Requirements for the specification of safety-related control functions	To set out the procedures to specify the requirements for safety-related control functions. These requirements are expressed in terms of functional requirements specification, and safety integrity requirements specification.
6: Design and integration of the safety-related electrical control system	To specify the selection criteria and/or the design and implementation methods of the SRECS to meet the functional safety requirements. This includes: selection of the system architecture, selection of the safety-related hardware and software, design of hardware and software, verification that the designed hardware and software meets the functional safety requirements.
7: Information for use of the machine	To specify requirements for the information for use of the SRECS, which has to be supplied with the machine. This includes: provision of the user manual and procedures, provision of the maintenance manual and procedures.
8: Validation of the safety-related electrical control system	To specify the requirements for the validation process to be applied to the SRECS. This includes inspection and testing of the SRECS to ensure that it achieves the requirements stated in the safety requirements specification.
9: Modification of the safety-related electrical control system	To specify the requirements for the modification procedure that has to be applied when modifying the SRECS. This includes: modifications to any SRECS are properly planned and verified prior to making the change; the safety requirements specification of the SRECS is satisfied after any modifications have taken place.

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

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

IEC 61310 (all parts), *Safety of machinery – Indication, marking and actuation*

IEC 61508-2, *Functional safety of electrical/electronic/ programmable electronic safety-related systems – Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems*

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

ISO 12100-1:2003, *Safety of machinery – Basic concepts, general principles for design – Part 1: Basic terminology, methodology*

ISO 12100-2:2003, *Safety of machinery – Basic concepts, general principles for design – Part 2: Technical principles*

ISO 13849-1:1999, *Safety of machinery – Safety related parts of control systems – Part 1: General principles for design*

ISO 13849-2:2003, *Safety of machinery – Safety-related parts of control systems – Part 2: Validation*

ISO 14121, *Safety of machinery – Principles of risk assessment*

3 Terms, definitions and abbreviations

3.1 Alphabetical list of definitions

Term	Definition number
application software	3.2.46
architectural constraint	3.2.36
architecture	3.2.35
common cause failure	3.2.43
complex component	3.2.8
control function	3.2.14
dangerous failure	3.2.40
demand	3.2.25
diagnostic coverage	3.2.38
electrical control system	3.2.3
embedded software	3.2.47
failure	3.2.39
fault	3.2.30
fault tolerance	3.2.31
full variability language (FVL)	3.2.48
function block	3.2.32
function block element	3.2.33

functional safety	3.2.9
hardware safety integrity	3.2.20
hazard (from machinery)	3.2.10
hazardous situation	3.2.11
high demand or continuous mode	3.2.27
limited variability language (LVL)	3.2.49
low complexity component	3.2.7
low demand mode	3.2.26
machine control system	3.2.2
machinery (machine)	3.2.1
mean time to failure (MTTF)	3.2.34
probability of dangerous failure per hour (PFH_D)	3.2.28
proof test	3.2.37
protective measure	3.2.12
random hardware failure	3.2.44
risk	3.2.13
safe failure	3.2.41
safe failure fraction	3.2.42
safety function	3.2.15
safety integrity	3.2.19
safety integrity level (SIL)	3.2.23
safety-related control function (SRCF)	3.2.16
safety-related electrical control system (SRECS)	3.2.4
safety-related software	3.2.50
SIL claim limit	3.2.24
software safety integrity	3.2.21
SRECS diagnostic function	3.2.17
SRECS fault reaction function	3.2.18
subsystem	3.2.5
subsystem element	3.2.6
systematic failure	3.2.45
systematic safety integrity	3.2.22
target failure value	3.2.29
validation	3.2.52
verification	3.2.51

3.2 Terms and definitions

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

3.2.1
machinery
assembly of linked parts or components, at least one of which moves, with the appropriate machine actuators, control and power circuits, joined together for a specific application, in particular for the processing, treatment, moving or packaging of a material.

The terms “machinery” and “machine” also cover an assembly of machines which, in order to achieve the same end, are arranged and controlled so that they function as an integral whole.

[ISO 12100-1:2003, 3.1]

3.2.2
machine control system
system which responds to an input from, for example, the process, other machine elements, an operator, external control equipment, and generates an output(s) causing the machine to behave in the intended manner

3.2.3
electrical control system
all the electrical, electronic and programmable electronic parts of the machine control system used to provide, for example, operational control, monitoring, interlocking, communications, protection and safety-related control functions

NOTE Safety-related control functions can be performed by an electrical control system that is either integral to or independent of those parts of a machine’s control system that perform non-safety-related functions.

3.2.4
Safety-Related Electrical Control System
SRECS
electrical control system of a machine whose failure can result in an immediate increase of the risk(s)

NOTE A SRECS includes all parts of an electrical control system whose failure may result in a reduction or loss of functional safety and this can comprise both electrical power circuits and control circuits.

3.2.5
subsystem
entity of the top-level architectural design of the SRECS where a failure of any subsystem will result in a failure of a safety-related control function

NOTE 1 A complete subsystem can be made up from a number of identifiable and separate subsystem elements, which when put together implement the function blocks allocated to the subsystem.

NOTE 2 This definition is a limitation of the general definition of IEC 61508-4: `set of elements which interact according to a design, where an element of a system can be another system, called a subsystem, which may include hardware, software and human interaction.

NOTE 3 This differs from common language where “subsystem” may mean any sub-divided part of an entity, the term “subsystem” is used in this standard within a strongly defined hierarchy of terminology: “subsystem” is the first level subdivision of a system. The parts resulting from further subdivision of a subsystem are called “subsystem elements”.

3.2.6
subsystem element
part of a subsystem, comprising a single component or any group of components

3.2.7
low complexity component
component in which

– the failure modes are well-defined; and

– the behaviour under fault conditions can be completely defined

[IEC 61508-4, 3.4.4 modified]

NOTE 1 Behaviour of the low complexity component under fault conditions may be determined by analytical and/or test methods.

NOTE 2 A subsystem or subsystem element comprising one or more limit switches, operating, possibly via interposing electro-mechanical relays, one or more contactors to de-energise an electric motor is an example of a low complexity component.

3.2.8
complex component
component in which
– the failure modes are not well-defined; or
– the behaviour under fault conditions cannot be completely defined

3.2.9
functional safety
part of the safety of the machine and the machine control system which depends on the correct functioning of the SRECS, other technology safety-related systems and external risk reduction facilities

[IEC 61508-4, 3.1.9 modified]

NOTE 1 This standard only considers the functional safety that depends on the correct functioning of the SRECS in machinery applications.

NOTE 2 ISO/IEC Guide 51 defines safety as freedom from unacceptable risk.

3.2.10
hazard (from machinery)
potential source of physical injury or damage to health

[ISO 12100-1: 2003, 3.6 modified]

NOTE The term hazard can be qualified in order to define its origin or the nature of the expected harm (e.g. electric shock hazard, crushing hazard, cutting hazard, toxic hazard, fire hazard).

3.2.11
hazardous situation
circumstance in which a person is exposed to a hazard(s)

[ISO 12100-1:2003, 3.9 modified]

3.2.12
protective measure
measure intended to achieve risk reduction

[ISO 12100-1:2003, 3.18 modified]

3.2.13
risk
combination of the probability of occurrence of harm and the severity of that harm

ISO 12100-1:2003, 3.11]

3.2.14
control function
function that evaluates input information or signals and produces output information or activities

3.2.15
safety function
function of a machine whose failure can result in an immediate increase of the risk(s)

[ISO 12100-1:2003, 3.28]

NOTE This definition differs from the definitions in IEC 61508-4 and ISO 13849-1.

3.2.16
Safety-Related Control Function
SRCF
control function implemented by a SRECS with a specified integrity level that is intended to maintain the safe condition of the machine or prevent an immediate increase of the risk(s)

3.2.17
SRECS diagnostic function
function intended to detect faults in the SRECS and produce a specified output information or activity when a fault is detected

NOTE This function is intended to detect faults that could lead to a dangerous failure of a SRCF and initiate a specified fault reaction function.

3.2.18
SRECS fault reaction function
function that is initiated when a fault within a SRECS is detected by the SRECS diagnostic function

3.2.19
safety integrity
probability of a SRECS or its subsystem satisfactorily performing the required safety-related control functions under all stated conditions

[IEC 61508-4, 3.5.2 modified]

NOTE 1 The higher the level of safety integrity of the item, the lower the probability that the item will fail to carry out the required safety-related control function.

NOTE 2 Safety integrity comprises hardware safety integrity (see 3.2.20) and systematic safety integrity (see 3.2.22).

3.2.20
hardware safety integrity
part of the safety integrity of a SRECS or its subsystems comprising requirements for both the probability of dangerous random hardware failures and architectural constraints

[IEC 61508-4, 3.5.5 modified]

3.2.21
software safety integrity
part of the systematic safety integrity of a SRECS or its subsystems related to the capability of software in a programmable electronic system performing its safety-related control functions under all stated conditions during a stated period of time

[IEC 61508-4, 3.5.3 modified]

NOTE Software safety integrity cannot usually be quantified precisely.

3.2.22
systematic safety integrity
part of the safety integrity of a SRECS or its subsystems relating to its resistance to systematic failures (see 3.2.45) in a dangerous mode.

[IEC 61508-4, 3.5.4 modified]

NOTE 1 Systematic safety integrity cannot usually be quantified precisely.

NOTE 2 Requirements for systematic safety integrity apply to both hardware and software aspects of a SRECS or its subsystems.

3.2.23
Safety Integrity Level
SIL
discrete level (one out of a possible three) for specifying the safety integrity requirements of the safety-related control functions to be allocated to the SRECS, where safety integrity level three has the highest level of safety integrity and safety integrity level one has the lowest

[IEC 61508-4, 3.5.6 modified]

NOTE SIL 4 is not considered in this standard, as it is not relevant to the risk reduction requirements normally associated with machinery. For requirements applicable to SIL 4, see IEC 61508-1 and IEC 61508-2.

3.2.24
SIL Claim Limit (for a subsystem)
SILCL
maximum SIL that can be claimed for a SRECS subsystem in relation to architectural constraints and systematic safety integrity

3.2.25
demand
event that causes the SRECS to perform its SRCF

3.2.26
low demand mode
mode of operation in which the frequency of demands on a SRECS is no greater than one per year and no greater than twice the proof-test frequency

NOTE Equipment that is only designed in accordance with requirements for the low demand mode of operation described in IEC 61508-1 and IEC 61508-2 can be unsuitable for use as part of a SRECS in this standard. Low demand mode of operation is not considered to be relevant for SRECS applications at machinery.

3.2.27
high demand or continuous mode
mode of operation in which the frequency of demands on a SRECS is greater than one per year or greater than twice the proof-test frequency

[IEC 61508-4, 3.5.12 modified]

NOTE 1 Low demand mode of operation is not considered to be relevant for SRECS applications at machinery. Therefore, in this standard SRECS are only considered to operate in the high demand or continuous mode.

NOTE 2 Demand mode means that a safety-related control function is only performed on request (demand) in order to transfer the machine into a specified state. The SRECS does not influence the machine until there is a demand on the safety-related control function.

NOTE 3 Continuous mode means that a safety-related control function is performed perpetually (continuously), i.e. the SRECS is continuously controlling the machine and a (dangerous) failure of its function can result in a hazard.

3.2.28
Probability of dangerous Failure per Hour
PFH_D
average probability of dangerous failure within 1 h

NOTE PFH_D should not be confused with probability of failure on demand (PFD).

3.2.29
target failure value
intended PFH_D to be achieved to meet a specific safety integrity requirement(s)

NOTE Target failure value is specified in terms of the probability of dangerous failure per hour.

[IEC 61508-4, 3.5.13 modified]

3.2.30
fault
abnormal condition that may cause a reduction in or loss of, the capability of a SRECS, a subsystem, or a subsystem element to perform a required function

[IEC 61508-4, 3.6.1 modified]

3.2.31
fault tolerance
ability of a SRECS, a subsystem, or subsystem element to continue to perform a required function in the presence of faults or failures

[IEC 61508-4, 3.6.3 modified]

3.2.32
function block
smallest element of a SRCF whose failure can result in a failure of the SRCF

NOTE 1 In this standard, a SRCF (F) may be seen as a logical AND of the function blocks (FB), i.e. F = FB_1 AND FB_2 AND FB_n.

NOTE 2 This definition of a function block differs from those used in IEC 61131-3 and other standards.

3.2.33
function block element
part of a function block

3.2.34
Mean Time To Failure
MTTF
expectation of the mean time to failure

[IEV 191-12-07, modified]

NOTE MTTF is normally expressed as an average value of expectation of the time to failure.

3.2.35
architecture
specific configuration of hardware and software elements in a SRECS

[IEC 61508-4, 3.3.5 modified]

3.2.36
architectural constraint
set of architectural requirements that limit the SIL that can be claimed for a subsystem

NOTE Requirements for architectural constraints are given in 6.7.6.

3.2.37
proof test
test that can detect faults and degradation in a SRECS and its subsystems so that, if necessary, the SRECS and its subsystems can be restored to an "as new" condition or as close as practical to this condition

[IEC 61508-4, 3.8.5 modified]

NOTE A proof test is intended to confirm that the SRECS is in a condition that assures the specified safety integrity.

3.2.38
diagnostic coverage
decrease in the probability of dangerous hardware failures resulting from the operation of the automatic diagnostic tests

[IEC 61508-4, 3.8.6 modified]

NOTE Diagnostic coverage (DC) can be calculated using the following equation:

$$DC = \Sigma \lambda_{DD} / \lambda_{Dtotal}$$

where λ_{DD} is the rate of detected dangerous hardware failures and λ_{Dtotal} is the rate of total dangerous hardware failures.

3.2.39
failure
termination of the ability of a SRECS, a subsystem, or a subsystem element to perform a required function

[IEC 61508-4, 3.6.4 modified and ISO 12100-1:2003, 3.32]

NOTE Failures are either random (in hardware) or systematic (in hardware or software).

3.2.40
dangerous failure
failure of a SRECS, a subsystem, or a subsystem element that has the potential to cause a hazard or non-functional state

[IEC 61508-4, 3.6.7 modified]

NOTE 1 Whether or not the potential is realised can depend on the channel architecture of the system; for example, in systems with multiple channels to improve safety, a dangerous hardware failure is less likely to lead to the overall dangerous or fail-to function state.

NOTE 2 In a subsystem with multiple channels, the probability of dangerous failure of the subsystem can be smaller than the dangerous failure rate of a channel that constitutes the subsystem. The probability of dangerous failure of a SRECS cannot be smaller than that of any subsystem constituting the SRECS. (This comes from the particular definition of "subsystem" in this standard.)

NOTE 3 A dangerous failure normally results in a failure or potential failure to perform the SRCF.

3.2.41
safe failure
failure of a SRECS, a subsystem of a SRECS, or a subsystem element of a SRECS that does not have the potential to cause a hazard

[IEC 61508-4, 3.6.8 modified]

3.2.42
Safe Failure Fraction
SFF
fraction of the overall failure rate of a subsystem that does not result in a dangerous failure

NOTE Safe Failure Fraction (*SFF*) can be calculated using the following equation:

$$(\Sigma\lambda_S + \Sigma\lambda_{DD}) \ / \ (\Sigma\lambda_S + \Sigma\lambda_D)$$

where

λ_S is the rate of safe failure,

$\Sigma\lambda_S + \Sigma\lambda_D$ is the overall failure rate,

λ_{DD} is the rate of dangerous failure which is detected by the diagnostic functions, and

λ_D is the rate of dangerous failure.

The diagnostic coverage (if any) of each subsystem in SRECS is taken into account in the calculation of the probability of random hardware failures. The safe failure fraction is taken into account when determining the architectural constraints on hardware safety integrity (see 6.7.7).

3.2.43
Common Cause Failure
CCF
failure, which is the result of one or more events, causing coincident failures of two or more separate channels in a multiple channel (redundant architecture) subsystem, leading to failure of a SRCF

[IEC 61508-4, 3.6.10 modified]

NOTE This definition differs from that given in ISO 12100-1 and IEV 191-04-23.

3.2.44
random hardware failure
failure occurring at a random time, which results from one or more of the possible degradation mechanisms in the hardware

[IEC 61508-4, 3.6.5]

3.2.45
systematic failure
failure related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation or other relevant factors

[IEC 61508-4, 3.6.6]

NOTE 1 Corrective maintenance without modification will usually not eliminate the failure cause.

NOTE 2 A systematic failure can be induced by simulating the failure cause.

NOTE 3 Examples of causes of systematic failures include human error in

- the safety requirements specification;
- the design, manufacture, installation and/or operation of the hardware;
- the design and/or implementation of the software.

3.2.46
application software
software specific to the application, that is implemented by the designer of the SRECS, generally containing logic sequences, limits and expressions that control the appropriate input, output, calculations, and decisions necessary to meet the SRECS functional requirements

3.2.47
embedded software
software, supplied by the manufacturer, that is part of the SRECS and that is not normally accessible for modification

NOTE Firmware and system software are examples of embedded software.

3.2.48
Full Variability Language
FVL
type of language that provides the capability to implement a wide variety of functions and applications

[IEC 61511-1, 3.2.81.1.3 modified]

NOTE 1 Typical example of systems using FVL are general-purpose computers.

NOTE 2 FVL is normally found in embedded software and is rarely used in application software.

NOTE 3 FVL examples include: Ada, C, Pascal, Instruction List, assembler languages, C++, Java, SQL.

3.2.49
Limited Variability Language
LVL
type of language that provides the capability to combine predefined, application specific, library functions to implement the safety requirements specifications

[IEC 61511-1, 3.2.81.1.2 modified]

NOTE 1 A LVL provides a close functional correspondence with the functions required to achieve the application.

NOTE 2 Typical examples of LVL are given in IEC 61131-3. They include ladder diagram, function block diagram and sequential function chart. Instruction lists and structured text are not considered to be LVL.

NOTE 3 Typical example of systems using LVL: Programmable Logic Controller (PLC) configured for machine control.

3.2.50
safety-related software
software that is used to implement safety-related control functions in a safety-related system

3.2.51
verification
confirmation by examination (e.g. tests, analysis) that the SRECS, its subsystems or subsystem elements meet the requirements set by the relevant specification

[IEC 61508-4, 3.8.1 modified and IEC 61511-1, 3.2.92 modified]

NOTE The verification results should provide documented objective evidence.

EXAMPLE: Verification activities include:

- reviews on outputs (documents from all phases) to ensure compliance with the objectives and requirements of the phase, taking into account the specific inputs to that phase;
- design reviews;
- tests performed on the designed products to ensure that they perform according to their specification;
- integration tests performed where different parts of a system are put together in a step-by-step manner and by the performance of environmental tests to ensure that all the parts work together in the specified manner.

3.2.52
validation
confirmation by examination (e.g. tests, analysis) that the SRECS meets the functional safety requirements of the specific application

[IEC 61508-4, 3.8.2 modified]

3.3 Abbreviations

The following abbreviations are used in this standard.

CCF	Common Cause Failure(s)
DC	Diagnostic Coverage
EMC	Electromagnetic Compatibility
FB	Function Block
FVL	Full Variability Language
I/O	Input/Output
LVL	Limited Variability Language
PFH_D	Probability of dangerous Failure per Hour
MTTF	Mean Time To Failure
MTTR	Mean Time To Restoration
P_{TE}	Probability of dangerous Transmission Error
SFF	Safe Failure Fraction
SIL	Safety Integrity Level
SILCL	Safety Integrity Level (SIL) Claim Limit (for subsystems)
S-R	Safety Related
SRECS	Safety-Related Electrical Control System
SRCF	Safety-Related Control Function
SRS	Safety Requirements Specification
SYS	System

4 Management of functional safety

4.1 Objective

This Clause specifies management and technical activities that are necessary for the achievement of the required functional safety of the SRECS.

4.2 Requirements

4.2.1 A functional safety plan shall be drawn up and documented for each SRECS design project, and shall be updated as necessary. The plan shall include procedures for control of the activities specified in Clauses 5 to 9.

NOTE 1 The content of the functional safety plan should depend upon the specific circumstances, which can include:

– size of project;
– degree of complexity;
– degree of novelty of design and technology;
– degree of standardization of design features;
– possible consequence(s) in the event of failure.

In particular the plan shall:

a) identify the relevant activities specified in Clauses 5 to 9.

b) describe the policy and strategy to fulfil the specified functional safety requirements.

c) describe the strategy to achieve functional safety for the application software, development, integration, verification and validation.

d) identify persons, departments or other units and resources that are responsible for carrying out and reviewing each of the activities specified in Clauses 5 to 9.

e) identify or establish the procedures and resources to record and maintain information relevant to the functional safety of a SRECS.

NOTE 2 The following should be considered:

- the results of the hazard identification and risk assessment;
- the equipment used for safety-related functions together with its safety requirements;
- the organization responsible for maintaining functional safety;
- the procedures necessary to achieve and maintain functional safety (including SRECS modifications).

f) describe the strategy for configuration management (see 9.3) taking into account relevant organizational issues, such as authorized persons and internal structures of the organization.

g) establish a verification plan that shall include:

– details of when the verification shall take place;
– details of the persons, departments or units who shall carry out the verification;
– the selection of verification strategies and techniques;
– the selection and utilization of test equipment;
– the selection of verification activities;
– acceptance criteria; and
– the means to be used for the evaluation of verification results.

h) establish a validation plan comprising:

- details of when the validation shall take place;
- identification of the relevant modes of operation of the machine (e.g. normal operation, setting);
- requirements against which the SRECS is to be validated;
- the technical strategy for validation, for example analytical methods or statistical tests;
- acceptance criteria; and
- actions to be taken in the event of failure to meet the acceptance criteria.

NOTE 3 The validation plan should indicate whether the SRECS and its subsystems are to be subject to routine testing, type testing and/or sample testing.

4.2.2 The functional safety plan shall be implemented to ensure prompt follow-up and satisfactory resolution of issues relevant to a SRECS arising from:

- activities specified in Clauses 5 to 9;
- verification activities; and
- validation activities.

5 Requirements for the specification of Safety-Related Control Functions (SRCFs)

5.1 Objective

This Clause sets out the procedures to specify the requirements of SRCF(s) to be implemented by the SRECS.

5.2 Specification of requirements for SRCFs

5.2.1 General

5.2.1.1 From the risk reduction strategy, as outlined in ISO 12100-1, ISO 12100-2, and ISO 14121, any need for safety functions will be determined.

5.2.1.2 Where safety functions are selected to be implemented (in whole or in part) by SRECS, then the associated SRCF(s) (see 3.2.16) shall be specified.

5.2.1.3 Specifications of each SRCF shall comprise:

- functional requirements specification (see 5.2.3);
- safety integrity requirements specification (see 5.2.4).

and these shall be documented in the safety requirements specification (SRS).

NOTE 1 Where non-electrical equipment contributes towards the performance of a safety function in combination with electrical means, the target failure value(s) applicable to the non-electrical equipment is not considered within this standard. Electrical means covers any and all devices or systems operating on electrical principles, including:

- electro-mechanical devices;
- non-programmable electronic devices;
- programmable electronic devices.

NOTE 2 The SRS needs to be subject to version control as part of the configuration management procedures (see 9.3).

5.2.1.4 The safety requirements specification shall be verified to ensure consistency and completeness for its intended use.

NOTE For example this may be achieved by inspection, analysis, check-lists. See also B.2.6 of IEC 61508-7.

5.2.2 Information to be available

The following information shall be used to produce both the functional requirements specification and safety integrity requirements specification of each SRCF:

– results of the risk assessment for the machine including all safety functions determined to be necessary for the risk reduction process for each specific hazard;
– machine operating characteristics, including:
  - modes of operation,
  - cycle time,
  - response time performance,
  - environmental conditions,
  - interaction of person(s) with the machine (e.g. repairing, setting, cleaning);
– all information relevant to the SRCFs which can have an influence on the SRECS design including, for example:
  - a description of the behaviour of the machine that a SRCF is intended to achieve or to prevent;
  - all interfaces between the SRCFs, and between SRCFs and any other function (either within or outside the machine);
  - required fault reaction functions of the SRCF.

NOTE Some of the information might not be available or sufficiently defined before starting the iterative design process of SRECS, so the SRECS safety requirements specifications can be required to be updated during the design process.

5.2.3 Functional requirements specification for SRCFs

5.2.3.1 The functional requirements specification for SRCFs shall describe details of each SRCF to be performed including, as applicable:

– the condition(s) (e.g. operating mode) of the machine in which the SRCF shall be active or disabled;
– the priority of those functions that can be simultaneously active and that can cause conflicting action;
– the frequency of operation of each SRCF;
– the required response time of each SRCF;
– the interface(s) of the SRCFs to other machine functions;
– the required response times (e.g. input and output devices);
– a description of each SRCF;
– a description of fault reaction function(s) and any constraints on, for example, re-starting or continued operation of the machine in cases where the initial fault reaction is to stop the machine;

– a description of the operating environment (e.g. temperature, humidity, dust, chemical substances, mechanical vibration and shock);

– tests and any associated facilities (e.g. test equipment, test access ports);

– rate of operating cycles, duty cycle, and/or utilisation category, for electromechanical devices intended for use in the SRCF.

5.2.3.2 In addition to the requirements of IEC 61000-6-2, when a SRECS is intended for use in an industrial environment, electromagnetic (EM) immunity levels are given in Annex E. SRECS intended for use in another EM environment (e.g. residential) should have immunity levels based on those specified in different EMC standards (e.g., for a residential environment, IEC 61000-6-1).

NOTE 1 When specifying EM immunity levels it is necessary to consider whether the levels used in different EMC standards cover cases which can occur in a SRECS application even with a low probability of occurrence.

NOTE 2 EM immunity performance criterion for functional safety of a SRECS is given in 6.4.3.

5.2.4 Safety integrity requirements specification for SRCFs

5.2.4.1 The safety integrity requirements for each SRCF shall be derived from the risk assessment to ensure the necessary risk reduction can be achieved. In this standard, a safety integrity requirement is expressed as a target failure value for the probability of dangerous failure per hour of each SRCF.

5.2.4.2 The safety integrity requirements for each SRCF shall be specified in terms of a SIL in accordance with Table 3 and documented. An example of a methodology is given in Annex A.

Table 3 – Safety integrity levels: target failure values for SRCFs

Safety integrity level	Probability of a dangerous Failure per Hour (PFH_D)
3	$\geq 10^{-8}$ to $< 10^{-7}$
2	$\geq 10^{-7}$ to $< 10^{-6}$
1	$\geq 10^{-6}$ to $< 10^{-5}$

NOTE Where the required safety integrity of a SRCF is less than SIL 1, as a minimum the requirements of category B of ISO 13849-1 should be met.

5.2.4.3 Where a product standard specifies a SIL for a SRCF then this shall take precedence over Annex A.

6 Design and integration of the safety-related electrical control system (SRECS)

6.1 Objective

This Clause specifies requirements for the selection or design of a SRECS to meet the functional and safety integrity requirements specified in the safety requirements specification (see 5.2).

6.2 General requirements

6.2.1 The SRECS shall be selected or designed to meet the safety requirements specification (see 5.2) and where relevant the software safety requirements specification (see 6.10) taking into account the appropriate requirements of this standard.

6.2.2 The selection or design of the SRECS (including the overall hardware and software architecture, sensors, actuators, programmable electronics, embedded software, application software, etc.) shall comply with either 6.5 or 6.6. Whichever method is used, the SRECS shall meet the following requirements:

a) the requirements for hardware safety integrity comprising:
   - the architectural constraints on hardware safety integrity (see 6.6.3.3); and
   - the requirements for the probability of dangerous random hardware failures (see 6.6.3.2);

b) the requirements for systematic safety integrity (see 6.4) comprising;
   - the requirements for the avoidance of failures, and
   - the requirements for the control of systematic faults;

c) the requirements for SRECS behaviour on detection of a fault (see 6.3);

d) the requirements for the design and development of safety-related software (see 6.10 and 6.11).

6.2.3 The design of the SRECS shall take into account human capabilities and limitations (including reasonably foreseeable misuse) and be suitable for the actions assigned to operators, maintenance staff and others who might interact with the SRECS. The design of all operator interfaces shall follow good human-factor practice (see the IEC 61310 series) and shall accommodate the likely level of training or awareness of operators, in particular, for mass-produced subsystems where the operator can be a member of the public.

NOTE The design goal should be that reasonably foreseeable mistakes made by operators or maintenance staff are prevented or eliminated by design. Where this is not possible, other means should also be applied (e.g. manual action with secondary confirmation before completion) to minimize the possibility of operator errors and ensure that foreseeable mistakes do not lead to increased risk.

6.2.4 Maintainability and testability shall be considered during the design and integration to facilitate the implementation of these properties in the SRECS.

6.2.5 The SRECS design, including its diagnostic and fault reaction functions, shall be documented. This documentation shall:

– be accurate, complete and concise;

– be suitable for its intended purpose;

– be accessible and maintainable;

– be version controlled.

6.2.6 The outcome of the activities performed during SRECS design, development and implementation shall be verified at appropriate stages.

6.3 Requirements for behaviour (of the SRECS) on detection of a fault in the SRECS

6.3.1 The detection of a dangerous fault in any subsystem that has a hardware fault tolerance of more than zero shall result in the performance of the specified fault reaction function.

The specification may allow isolation of the faulty part of the subsystem to continue safe operation of the machine while the faulty part is repaired. In this case, if the faulty part is not repaired within the estimated maximum time as assumed in the calculation of the probability of random hardware failure (see 6.7.8), then a second fault reaction shall be performed to maintain a safe condition.

Where the SRECS is designed for online repair, isolation of a faulty part shall only be applicable where this does not increase the probability of dangerous random hardware failure of the SRECS above that specified in the SRS.

After the occurrence of faults that reduce the hardware fault tolerance to zero, the requirements of 6.3.2 apply.

NOTE The mean time to restoration (see IEV 191-13-08) that is considered in the reliability model will need to take into account the diagnostic test interval, the repair time and any other delays prior to restoration.

6.3.2 Where a diagnostic function(s) is necessary to achieve the required probability of dangerous random hardware failure and the subsystem has a hardware fault tolerance of zero, then the fault detection and specified fault reaction shall be performed before the hazardous situation addressed by the SRCF can occur.

> **EXCEPTION to 6.3.2:** In the case of a subsystem implementing a particular SRCF where the hardware fault tolerance is zero and the ratio of the diagnostic test rate to the demand rate exceeds 100, then the diagnostic test interval of that subsystem shall be such as to enable the subsystem to meet the requirement for the probability of dangerous random hardware failure.

6.3.3 Where performance of a fault reaction function as part of a SRCF that is specified as SIL 3 has resulted in the machine being stopped, subsequent normal operation of the machine via the SRECS (e.g. enabling re-start of the machine) shall not be possible until the fault has been repaired or rectified. For SRCFs with a specified safety integrity of less than SIL 3, the behaviour of the machine after performance of a fault reaction function (e.g. re-starting normal operation) shall depend on the specification of relevant fault reaction functions (see 5.2.3).

6.4 Requirements for systematic safety integrity of the SRECS

NOTE These requirements are applicable at the 'system level' where subsystems are interconnected to realise a SRECS. For requirements relevant to subsystem realisation, see 6.7.8.

6.4.1 Requirements for the avoidance of systematic hardware failures

6.4.1.1 The following measures shall be applied:

a) the SRECS shall be designed and implemented in accordance with the functional safety plan (see 4.2);

b) proper selection, combination, arrangements, assembly and installation of subsystems, including cabling, wiring and any interconnections;

c) use of the SRECS within the manufacturer's specification;

d) use of manufacturer's application notes, for example catalogue sheets, installation instructions, and use of good engineering practice (see also ISO 13849-2, Clause D.1);

e) use of subsystems that have compatible operating characteristics (see also ISO 13849-2, Clause D.1);

f) the SRECS shall be protected in accordance with IEC 60204-1;

g) prevention of the loss of functional earth connection(s) in accordance with IEC 60204-1;

h) undocumented modes of component operation shall not be used (e.g. 'reserved' registers of programmable equipment); and

i) consideration of foreseeable misuse, environmental changes or modification(s).

6.4.1.2 In addition, at least one of the following techniques and/or measures shall be applied taking into account the complexity of the SRECS and the SIL(s) for those functions to be implemented by the SRECS:

a) SRECS hardware design review (e.g. by inspection or walk-through): to establish by reviews and/or analysis any discrepancies between the specification and implementation;

NOTE 1 In order to reveal discrepancies between the specification and implementation, any points of doubt or potential weak points concerning the realisation, the implementation and the use of the product are documented so they can be resolved; taking into account that on an inspection procedure the author is passive and the inspector is active whilst on a walk-through procedure the author is active and the inspector is passive.

b) advisory tools such as computer-aided design packages capable of simulation or analysis, and/or the use of computer-aided design tools to perform the design procedures systematically with the use of pre-designed elements that are already available and tested;

NOTE 2 The integrity of these tools can be demonstrated by specific testing, or by an extensive history of satisfactory use, or by independent verification of their output for the particular SRECS that is being designed. See 6.11.3.4.

c) simulation: perform a systematic and complete assimilation of a SRECS design in terms of both functional performance and the correct dimensioning and interaction of its subsystems.

EXAMPLE The function of the SRECS can be simulated on a computer via a software behavioural model (see 6.11.3.4) where individual subsystems or subsystem elements each have their own simulated behaviour, and the response of the circuit in which they are connected is examined by looking at the marginal data of each subsystem or subsystem element.

6.4.2 Requirements for the control of systematic faults

The following measures shall be applied:

a) use of de-energization: the SRECS shall be designed so that with loss of its electrical supply a safe state of the machine is achieved or maintained;

b) measures to control the effect of temporary subsystem failures: the SRECS shall be designed so that, for example:

- voltage variation (e.g. interruptions, dips) to an individual subsystem or a part of a subsystem does not lead to a hazard (e.g. a voltage interruption that affects a motor circuit shall not cause an unexpected start-up when the supply is restored), and

  NOTE 1 See also relevant requirements of IEC 60204-1. In particular:

  overvoltage or undervoltage should be detected early enough so that all outputs can be switched to a safe condition by the power-down routine or a switch-over to a second power unit; and/or

  where necessary, overvoltage or undervoltage should be detected early enough so that the internal state can be saved in non-volatile memory, so that all outputs can be set to a safe condition by the power-down routine, or all outputs can be switched to a safe condition by the power-down routine or a switch-over to a second power unit.

- the effects of electromagnetic interference from the physical environment or a subsystem(s) do not lead to a hazard;

c) measures to control the effects of errors and other effects arising from any data communication process, including transmission errors, repetitions, deletion, insertion, re-sequencing, corruption, delay and masquerade;

NOTE 2 Further information can be found in IEC 60870-5-1, EN 50159-1, EN 50159-2 and IEC 61508-2.

NOTE 3 The term 'masquerade' means that the true contents of a message are not correctly identified. For example, a message from a non-safety component is incorrectly identified as a message from a safety component.

d) when a dangerous fault occurs at an interface, the fault reaction function shall be performed before the hazard due to this fault can occur. When a fault that reduces the hardware fault tolerance to zero occurs, this fault reaction shall take place before the estimated MTTR (see 6.7.4.4.2 g) is exceeded.

The requirements of item d) apply to interfaces that are inputs and outputs of subsystems and all other parts of subsystems that include or require cabling during integration (for example output signal switching devices of a light curtain, output of a guard position sensor).

NOTE 4 This does not require that a subsystem or subsystem element on its own has to detect a fault on its outputs(s). The fault reaction function may also be initiated by any subsequent subsystem after a diagnostic test is performed.

6.4.3 Electromagnetic (EM) immunity

In addition to the requirements of IEC 61000-6-2 and the EM phenomena given in Annex E, the following performance criterion for functional safety shall be satisfied by a SRECS:

– unsafe conditions or hazards shall not be introduced; and

– no loss of the SRCF(s); or

– the SRCF(s) implemented by the SRECS may be disturbed temporarily or permanently provided that a safe state of the machine is maintained or achieved before a hazard can occur. Where the EM phenomena can result in the destruction of components, it shall be ensured (e.g. by analysis) that functional safety is not affected, including by lower value(s) of EM phenomena that can cause partial destruction.

NOTE Consideration should be given to the behaviour of the SRECS in response to EM phenomena at all value(s) up to those given in Annex E.

6.5 Selection of safety-related electrical control system

Where a supplier provides a SRECS for a specific function referenced in the safety requirements specification, a pre-designed SRECS may be selected instead of a custom design providing that it meets the requirements of the safety requirements specification and 6.3, 6.4 and 6.6.1.

NOTE Selection of a pre-designed SRECS is an alternative to the design and development of a specific SRECS in accordance with 6.6.

6.6 Safety-related electrical control system (SRECS) design and development

6.6.1 General requirements

6.6.1.1 The SRECS shall be designed and developed in accordance with the SRECS safety requirements specification (see 5.2).

6.6.1.2 A clearly structured design process shall be followed and documented (see 6.6.2).

6.6.1.3 Where the use of diagnostics is necessary to achieve the required safety integrity when a fault is detected, the SRECS shall perform the specified fault reaction function (see 5.2 and 6.3).

6.6.1.4 Where a SRECS or part of a SRECS (i.e. its subsystem(s)) is to implement both SRCFs and other functions, then all its hardware and software shall be treated as safety-related unless it can be shown that the implementation of the SRCFs and other functions is sufficiently independent (i.e. that the normal operation or failure of any other functions do not affect the SRCFs).

NOTE Sufficient independence of implementation can be established by showing that the probability of a dependent failure between the non-safety and safety-related parts is equivalent to that of the safety integrity level of the SRECS.

6.6.1.5 For a SRECS or its subsystems that implements safety-related control functions of different safety integrity levels, its hardware and software shall be treated as requiring the highest safety integrity level unless it can be shown that the implementation of the safety-related control functions of the different safety integrity levels is sufficiently independent.

NOTE Sufficient independence of implementation can be established by showing that the probability of a dependent failure between the parts implementing SRCFs of different integrity levels is equivalent to that of the safety integrity level achieved by the SRECS.

6.6.1.6 Interconnections (e.g. wiring, cabling) other than digital data communication shall be considered to be part of one of the subsystems to which they are connected (see also item d) of 6.4.2).

6.6.1.7 Where digital data communication is used as a part of a SRECS implementation it shall satisfy the relevant requirements of IEC 61508-2 in accordance with the SIL target(s) of the SRCF(s).

6.6.1.8 The information for use of the SRECS shall specify those techniques and measures necessary during the design life of the SRECS to maintain the safety integrity level.

6.6.2 Design and development process

The design and development shall follow a clearly defined process that shall take into account all aspects covered by the process shown in Figure 2.

NOTE The approach of this standard is to apply a structured design process to the SRECS beginning from the requirements that are specified in the Safety Requirements Specification. Figure 3 shows the workflow of the design process and the terminology that applies to the different levels.

6.6.2.1 System architecture design

6.6.2.1.1 Each SRCF as specified in the SRECS safety requirements specification shall be decomposed to a structure of function blocks, for example as shown in Figure 3. This structure shall be documented comprising:

– the description of the structure;

– the safety requirements (functional, integrity,) for each function block;

– definition of inputs and outputs of each function block.

NOTE 1 The decomposition process should lead to a structure of function blocks that fully describes the functional and integrity requirements of the SRCF. This process should be applied down to that level that permits the functional and integrity requirements determined for each function block to be allocated to subsystems, where the allocation to a subsystem of the complete functional requirements of a function block is possible. However, it is possible to allocate more than one function block to a single subsystem, but it is not possible to allocate one function block to several subsystems where it is intended that these subsystems have separate functional and integrity requirements. Where the intention is to allocate the functional requirements of one function block to redundant subsystem elements, refer to 6.7.4.

NOTE 2 The inputs and outputs of each function block are the information that is transferred, for example speed, position, mode of operation, etc.

NOTE 3 The function blocks are a representation of functions of the SRCF (see 3.2.16) and do not include SRECS diagnostic functions (see 3.2.17). For the purposes of this standard, the diagnostic functions are considered as separate functions that may have a different structure to the SRCF (see 6.8).

6.6.2.1.2 An initial concept for an architecture of the SRECS shall be created in accordance with the structure of the function blocks.

NOTE There should be ongoing collaboration between the developer of the safety-related control architecture, the organization responsible for configuration of the devices and the developer of the software. As the software safety requirements and the possible software architecture become more precise, there may be an impact on the SRECS hardware architecture, and for this reason close co-operation between the SRECS architecture designer, the subsystem supplier(s), software developer and, as necessary, the machinery designer or the user can help to reduce the potential for systematic failure(s).

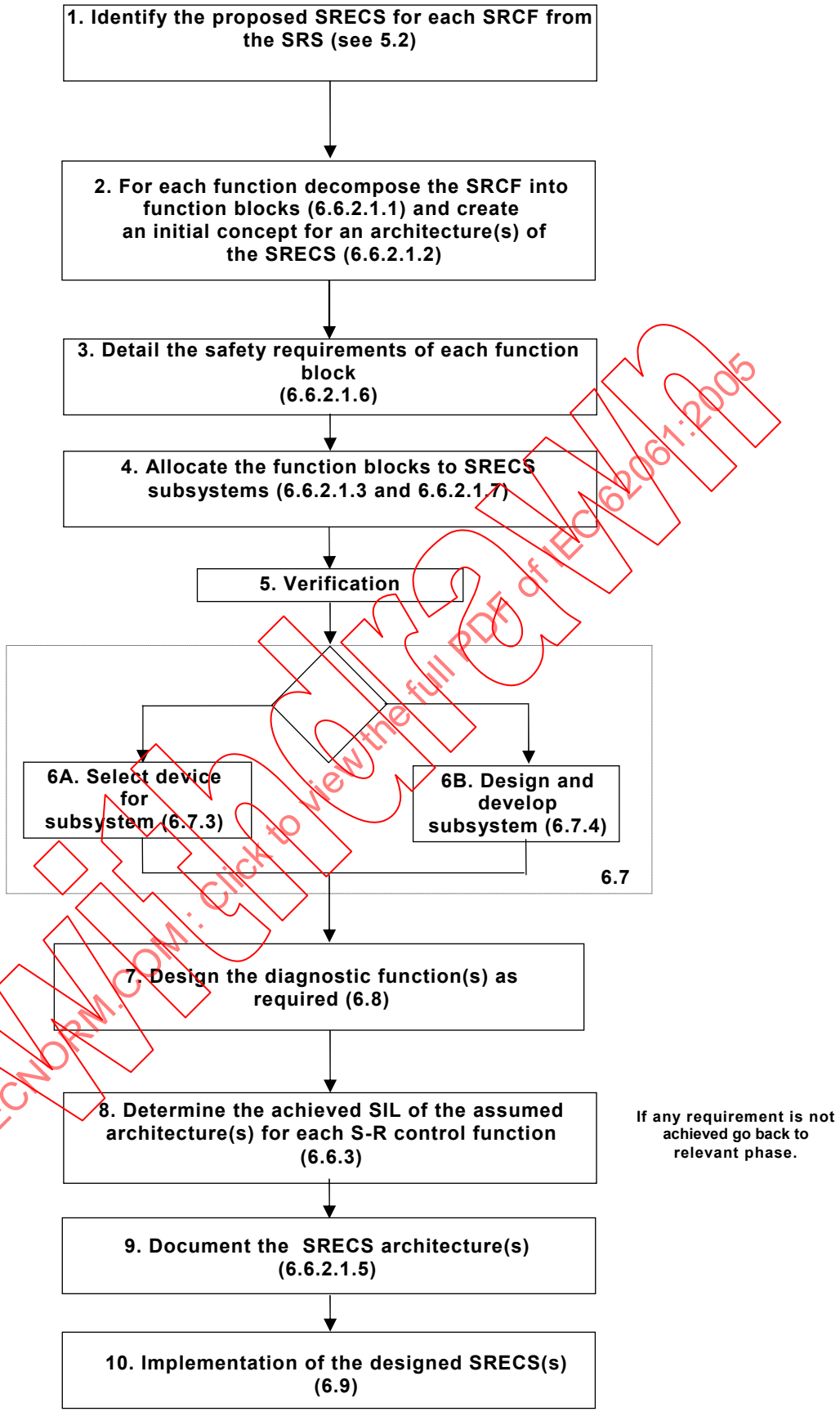


Figure 2 – Workflow of the SRECS design and development process

6.6.2.1.3 Each function block shall be allocated to a subsystem within the architecture of the SRECS. More than one function block may be allocated to one subsystem.

6.6.2.1.4 Each subsystem and the function blocks allocated to it shall be clearly identified.

6.6.2.1.5 The architecture shall be documented describing its subsystems and their interrelationship.

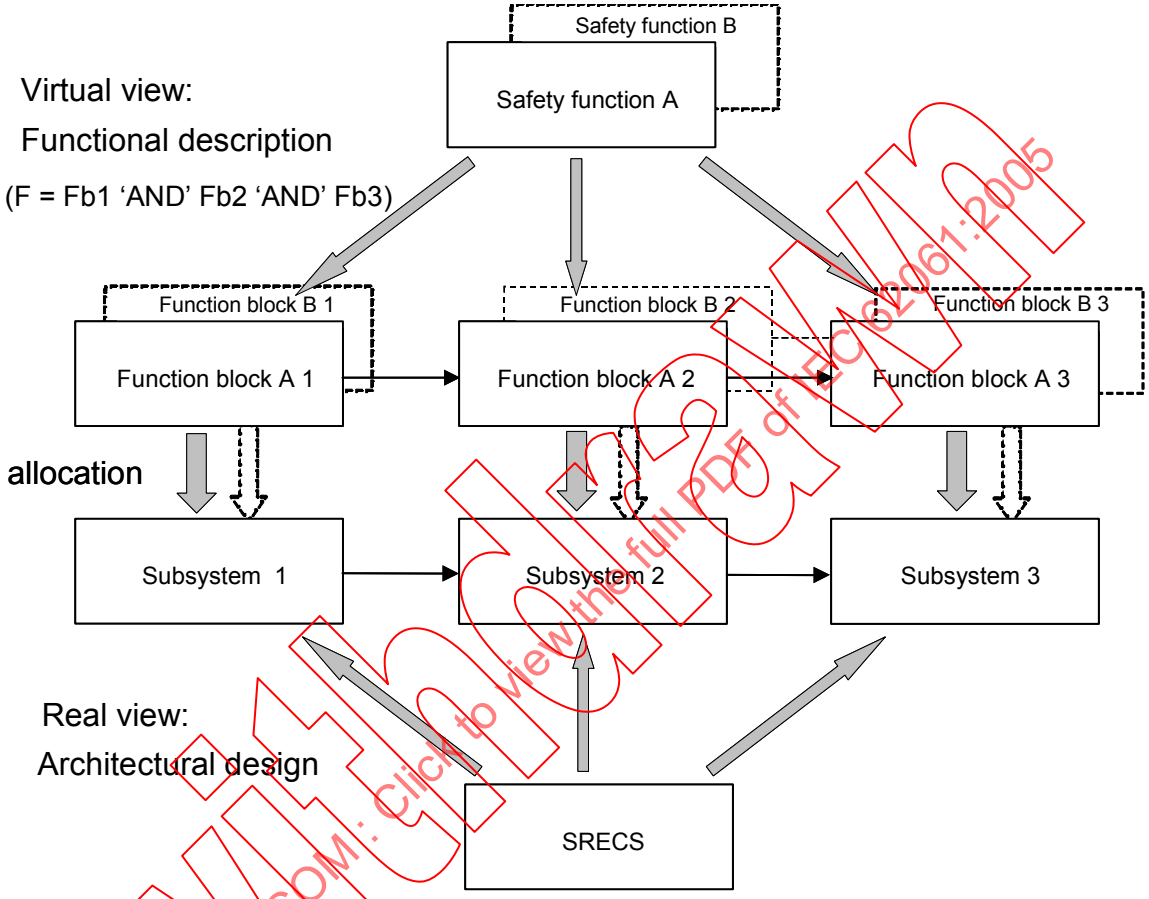


Figure 3 – Allocation of safety requirements of the function blocks to subsystems (see 6.6.2.1.1)

6.6.2.1.6 The safety requirements for each function block shall be as specified in the safety requirements specification of the corresponding SRCF in terms of

– functional requirements (e.g. input information, internal operation (logic) and output of the function block);

– safety integrity requirements.

6.6.2.1.7 The safety requirements for a subsystem shall be those of the function block(s) allocated to it. If more than one function block is allocated to a subsystem, then the highest integrity requirement applies (see 6.6.3). These requirements shall be documented as the subsystem safety requirements specification.

6.6.3 Requirements for the estimation of the safety integrity achieved by a SRECS

6.6.3.1 General

The SIL that can be achieved by the SRECS shall be considered separately for each SRCF to be performed by the SRECS.

The SIL that can be achieved by the SRECS shall be determined from the probability of dangerous random hardware failure, architectural constraints, and systematic safety integrity of the subsystems that comprise the SRECS. The SIL that is achieved is less than or equal to the lowest value of the SILCLs of any of the subsystems for systematic safety integrity and architectural constraints.

6.6.3.2 Hardware safety integrity

6.6.3.2.1 The probability of dangerous failure of each SRCF due to dangerous random hardware failures shall be equal to or less than the target failure value as specified in the safety requirements specification.

NOTE The target values associated with SILs are given in Table 3.

6.6.3.2.2 The probability of dangerous failure of each SRCF due to dangerous random hardware failures shall be estimated taking into account:

a) the architecture of the SRECS as it relates to each SRCF under consideration;

NOTE This involves deciding which failure modes of the subsystems are in a series configuration (i.e. any failure causes failure of the relevant SRCF to be carried out) and which are in a parallel (redundant) configuration (i.e. co-incident failures are necessary for the relevant SRCF to fail).

b) the estimated rate of failure of each subsystem to perform its allocated function block(s) in any modes which would cause a dangerous failure of the SRECS.

6.6.3.2.3 The estimation of the probability of dangerous failure shall be based on the probability of dangerous random hardware failure of each relevant subsystem as derived using the information required in 6.7.2.2 including, where appropriate 6.7.2.2 (k), for digital data communication processes between subsystems. The probability of dangerous random hardware failure of the SRECS is the sum of the probabilities of dangerous random hardware failure of all subsystems involved in the performance of the SRCF and shall include, where appropriate, the probability of dangerous transmission errors for digital data communication processes:

$$PFH_{\mathrm{D}} = PFH_{\mathrm{D1}} + \ldots + PFH_{\mathrm{Dn}} + P_{\mathrm{TE}}$$

NOTE 1 This approach is based on the definition of a function block which states that a failure of any function block will result in a failure of the SRCF (see 3.2.16).

NOTE 2 Interconnections other than digital data communication are considered to be a part of the subsystems.

6.6.3.3 Architectural constraints

The SIL achieved by the SRECS according to the architectural constraints is less than or equal to the lowest SILCL of any subsystem (see 6.7.6) involved in the performance of the SRCF.

NOTE For example, a SRECS comprises two series connected subsystems (subsystem 1 and subsystem 2) where the SFF and fault tolerance of each subsystem are assumed to be as shown in Table 4. The estimated PFH_{D} for the SRECS is 8×10^{-8}, which corresponds to SIL 3. However, according to Table 5 the architectural constraint of subsystem 2 limits the SIL that can be achieved by the SRECS to SIL 2.

Table 4 – Characteristics of subsystems 1 and 2 used in this example (see Note above)

Subsystem	Hardware fault tolerance	SFF	SIL claim limit according to architectural constraints (see Table 5)
1	1	95 %	SIL 3
2	1	80 %	SIL 2

6.6.3.4 Systematic safety integrity

The SIL achieved by the SRECS is less than or equal to the lowest SILCL of any subsystem involved in the performance of the SRCF.

NOTE The measures described in 6.7.9 give a SILCL of up to SIL 3 for systematic safety integrity of a subsystem realised according to 6.7.4.

6.7 Realisation of subsystems

6.7.1 Objective

The objective is to realise a subsystem that fulfils all safety requirements of the allocated function blocks (see Figure 3). Two approaches are considered:

– selection of a device that is sufficient to fulfil the requirements for that subsystem, i.e. it shall fulfil the safety requirements specification of each of its allocated function blocks and the requirements of this standard; or

– design and development of a subsystem by combining function block elements and specifying how they are arranged and how they interact.

6.7.2 General requirements for subsystem realisation

6.7.2.1 The subsystem shall be realised by either selection (see 6.7.3) or design (see 6.7.4) in accordance with its safety requirements specification (see 6.6.2.1.7), taking into account all the requirements of 6.2. Subsystem(s) incorporating complex components shall comply with IEC 61508-2 and IEC 61508-3 as appropriate for the required SIL.

EXCEPTION: Where a subsystem design includes a complex component as a subsystem element, 6.7.4.2.3 is applicable.

6.7.2.2 The following information shall be available for each subsystem:

a) a functional specification of those functions and interfaces of the subsystem which can be used by SRCFs;

b) the estimated rates of failure (due to random hardware failures) declared in any modes which could cause a dangerous failure of the SRECS;

NOTE 1 For electromechanical subsystems, the probability of failure should be estimated taking into account the number of operating cycles declared by the manufacturer and the duty cycle (see 5.2.3). This information should be based upon a B10 value (i.e. the expected time at which 10% of the population will fail). See also IEC 61810-2[1].

c) constraints on the subsystem for

– the environment and operating conditions which should be observed in order to maintain the validity of the estimated rates of failure due to random hardware failures; and

[1] To be published.

– the lifetime of the subsystem which should not be exceeded in order to maintain the validity of the estimated rates of failure due to random hardware failures;

d) any test and/or maintenance requirements;

e) the diagnostic coverage and the diagnostic test interval (when required, see Note 2);

NOTE 2 Item e) relates to diagnostic functions that are external to the subsystem. This information is only required when credit is claimed in the reliability model of the SRECS for the action of the diagnostic functions performed in the subsystem.

f) any additional information (e.g. repair times) which is necessary to allow the derivation of a mean time to restoration (MTTR) following detection of a fault by the diagnostics.

NOTE 3 Items b) to f) are needed to allow the probability of failure per hour of the SRCF to be estimated.

g) the SILCL due to architectural constraints (see 6.7.6) or:

– all information which is necessary to enable the derivation of the safe failure fraction (SFF) of the subsystem as applied in the SRECS; and

NOTE 4 The required information is the possible failure modes of the subsystem. Based on the failure modes of the subsystem, it can be decided whether the subsystem failure causes a safe or a dangerous failure of the SRECS.

NOTE 5 For details on estimation of the SFF see 6.7.7.

– the hardware fault tolerance of the subsystem;

h) any limits on the application of the subsystem which should be observed in order to avoid systematic failures;

i) the highest safety integrity level that can be claimed for a SRCF which uses the subsystem on the basis of:

– measures and techniques used to prevent systematic faults being introduced during the design and implementation of the hardware and software of the subsystem;

– the design features that make the subsystem tolerant against systematic faults.

NOTE 6 Items h) and i) are needed to determine the highest safety integrity level that can be claimed for a SRCF according to the architectural constraints. Also, these items can be used to provide a link (see Tables 4 and 5) to the category requirements of ISO 13849-1 in terms of both fault detection and hardware fault tolerance.

j) any information which is required to identify the hardware and software configuration of the subsystem in order to enable the configuration management of a SRECS in accordance with 6.11.3.2;

k) the probability of dangerous transmission errors for digital data communication processes, where applicable.

6.7.3 Requirements for selection of existing (pre-designed) subsystems

6.7.3.1 Where a supplier provides a subsystem for a specific SRCF referenced in the safety requirements specification, such a pre-designed subsystem may be selected instead of a custom design providing that it satisfies the safety requirements specification for the subsystem, 6.4.3 and 6.7.3.2 or 6.7.3.3.

6.7.3.2 Subsystems incorporating complex components shall comply with IEC 61508-2 and IEC 61508-3 as appropriate for the required SIL.

EXCEPTION: Where a subsystem design includes a complex component as a subsystem element, 6.7.4.2.3 is applicable.

6.7.3.3 Subsystems comprising low complexity components only shall comply with 6.7.4.4.1, 6.7.6.2, 6.7.6.3, 6.7.7, 6.7.8 and 6.8 of this standard.

6.7.4 Design and development of subsystems

6.7.4.1 Objectives

6.7.4.1.1 The first objective is to design a subsystem that fulfils the safety requirements of the allocated function block(s).

6.7.4.1.2 The second objective is to create an architecture in terms of subsystem elements that work together in combination to fulfil the functional and safety integrity requirements of all function blocks allocated to the subsystem.

6.7.4.2 General requirements

6.7.4.2.1 The subsystem shall be designed in accordance with its safety requirements specification.

6.7.4.2.2 The subsystem shall be such as to meet all of the requirements a) to c) as follows:

a) the requirements for hardware safety integrity comprising:
   - the architectural constraints on hardware safety integrity (see 6.7.6), and
   - the requirements for the probability of dangerous random hardware failures (see 6.7.8);

b) the requirements for systematic safety integrity comprising:
   - the requirements for the avoidance of failures (see 6.7.9.1), and the requirements for the control of systematic faults (see 6.7.9.2), or
   - evidence that the equipment is 'proven-in-use'. In this case, the subsystem shall fulfil the relevant requirements of IEC 61508-2 (see IEC 61508-2, 7.4.7.5 to 7.4.7.12).

c) the requirements for subsystem behaviour on detection of a fault (fault reaction)(see 6.3).

6.7.4.2.3 Where the design of a subsystem incorporates a complex component (as a subsystem element) which satisfies all relevant requirements of IEC 61508-2 and IEC 61508-3 in relation to the SILCL, it can be considered as a low complexity component in the context of a subsystem design since its relevant failure modes, behaviour on detection of a fault, rate of failure, and other safety-related information are known. Such components shall only be used in accordance with its specification and the relevant information for use provided by its supplier.

6.7.4.3 Subsystem design and development process

The subsystem design and development shall follow a clearly defined process that shall take into account all aspects covered by the process shown in Figure 4.

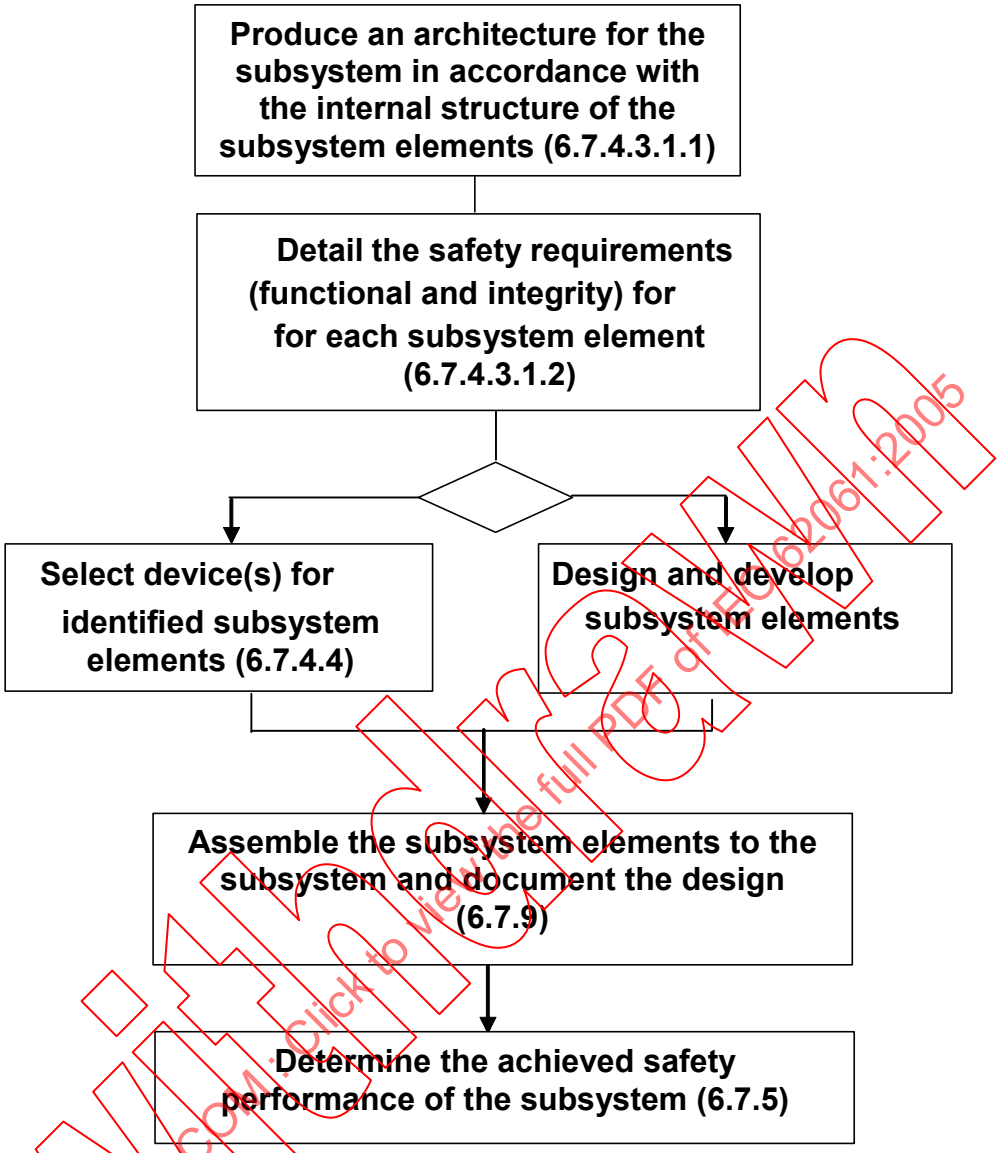


Figure 4 – Workflow for subsystem design and development (see box 6B of Figure 2)

6.7.4.3.1 Subsystem architecture design

6.7.4.3.1.1 During subsystem architecture design, the decomposition process should lead to a structure of function block elements that fully represent the functional requirements of the function block. This process should be applied down to that level which permits the functional requirements determined for each function block element to be allocated to subsystem elements (see example in Figure 5).

NOTE The workflow of the design process is shown in Figure 4.

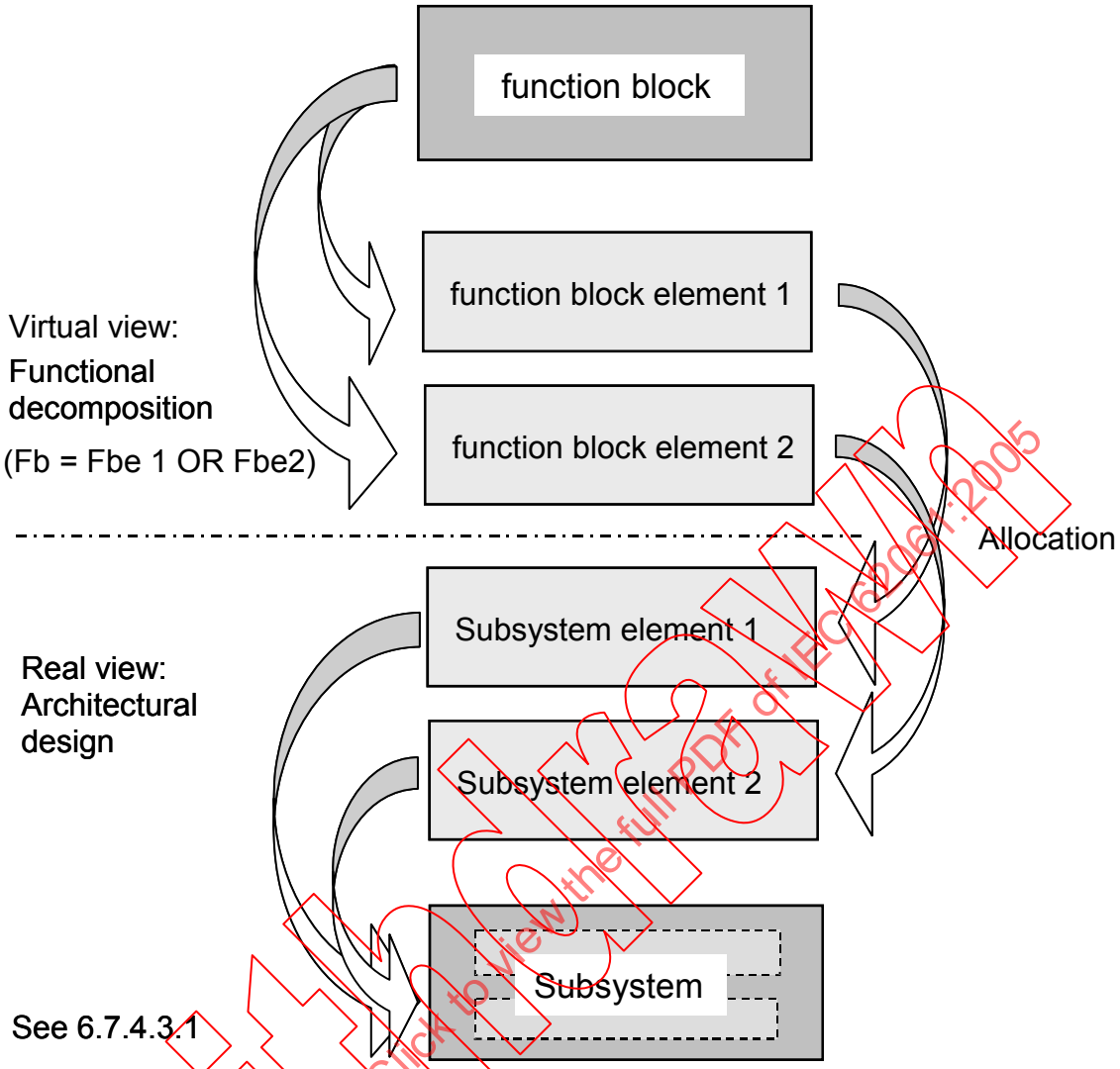


Figure 5 – Decomposition of a function block into redundant function block elements and their associated subsystem elements

6.7.4.3.1.2 The subsystem architecture shall be documented in terms of its elements and their interrelationships. Where necessary this shall also include information relating to function block elements that are allocated to subsystem elements.

6.7.4.4 Requirements for the selection and design of subsystem elements

6.7.4.4.1 Subsystem elements shall be suitable for their intended use and shall conform to relevant international standards where such exist.

6.7.4.4.2 The following information shall be available for each subsystem element:

a) a functional specification of the subsystem element;

b) specification of the interfaces of the subsystem element (e.g. electrical characteristics);

c) each failure mode and its probability of occurrence, and, where relevant (e.g. complex components used in accordance with 6.7.4.2.3), the diagnostic coverage and probability of dangerous failure.

NOTE For electromechanical subsystems, the probability of failure should be estimated taking into account the number of operating cycles declared by the manufacturer and the duty cycle of the application (see 5.2.3). This information should be based upon a B10 value (i.e. the expected time at which 10 % of the population will fail). See also IEC 61810-2[2].

d) constraints on the subsystem element for

– the environment and operating conditions which should be observed in order to maintain the validity of the information given in item c); and

– the lifetime of the subsystem element which should not be exceeded in order to maintain the validity of the information given in item (c);

e) any periodic proof test and/or maintenance requirements;

f) features that can contribute to diagnostics (e.g. mechanically linked contacts);

g) any additional information (e.g. repair times) which is necessary to allow the derivation of a mean time to restoration (MTTR) following detection of a fault by the diagnostics;

h) any limits on the application of the subsystem element which should be observed in order to avoid systematic failures;

i) hardware fault tolerance.

6.7.5 Determination of the safety performance of the subsystem

The safety performance of a subsystem is characterized by the SILCL determined by its architectural constraints (6.7.6), its SILCL due to systematic integrity (6.7.9) and its probability of dangerous random hardware failure (6.7.8).

NOTE 1 The SILCL of a subsystem sets a limit for the maximum safety integrity level that can be claimed for a safety-related control function using this subsystem.

NOTE 2 Information about all three aspects is necessary to determine the SIL achieved by the safety-related control system implementing the allocated SRCF.

6.7.6 Architectural constraints on hardware safety integrity of subsystems

6.7.6.1 In the context of hardware safety integrity, the highest safety integrity level that can be claimed for a SRCF is limited by the hardware fault tolerances and safe failure fractions of the subsystems that carry out that SRCF. Table 5 specifies the highest safety integrity level that can be claimed for a SRCF that uses a subsystem taking into account the hardware fault tolerance and safe failure fraction of that subsystem. The architectural constraints given in Table 5 shall be applied to each subsystem. With respect to these architectural constraints:

a) a hardware fault tolerance of N means that N+1 faults could cause a loss of the SRCF. In determining the hardware fault tolerance, no account is taken of other measures that can control the effects of faults such as diagnostics; and

b) where one fault directly leads to the occurrence of one or more subsequent faults, these shall be considered as a single fault;

c) in determining hardware fault tolerance, certain faults may be excluded, provided that the likelihood of them occurring is very low in relation to the safety integrity requirements of the subsystem. Any such fault exclusions shall be justified and documented (see also 6.7.7).

[2] To be published.

6.7.6.2 The architectural constraints of Table 5 shall apply to each subsystem implementing a function block of an SRCF.

6.7.6.3 A subsystem that comprises only a single subsystem element shall satisfy the requirements of Table 5. In particular, for such a subsystem that has a hardware fault tolerance of zero (i.e. N = 0) then a SFF of greater than 99 % shall be achieved by a SRECS diagnostic function(s).

NOTE This requirement is necessary to ensure an appropriate form of the architectural constraints is applied to subsystems that comprise only a single subsystem element in order to justify a SILCL of SIL 3.

Table 5 – Architectural constraints on subsystems: maximum SIL that can be claimed for a SRCF using this subsystem

Safe failure fraction	Hardware fault tolerance (see Note 1)		
	0	1	2
< 60 %	Not allowed (for exceptions see Note 3)	SIL1	SIL2
60 % – < 90 %	SIL1	SIL2	SIL3
90 % – < 99 %	SIL2	SIL3	SIL3 (see Note 2)
≥ 99 %	SIL3	SIL3 (see Note 2)	SIL3 (see Note 2)

NOTE 1 A hardware fault tolerance of N means that N+1 faults could cause a loss of the safety-related control function.

NOTE 2 A SIL 4 claim limit is not considered in this standard. For SIL 4 see IEC 61508-1.

NOTE 3 See 6.7.6.4 or for subsystems where fault exclusions have been applied to faults that could lead to a dangerous failure, see 6.7.7.

6.7.6.4 Electromechanical subsystems, which have a safe failure fraction of less than 60 % and zero hardware fault tolerance, that use well-tried components (see Note) in accordance with ISO 13849-1:2006 Category 1 PLC shall be considered to achieve a SILCL of SIL1.

NOTE A well-tried component for a safety-related application is a component which has been:

a) widely used in the past with successful results in similar applications, or

b) made and verified using principles which demonstrate its suitability and reliability for safety-related applications.

6.7.6.5 Where a subsystem is designed according to ISO 13849-1:1999 and validated according to ISO 13849-2:2003, the following relationship in the context of architectural constraints alone can be applied in accordance with Table 6. It is assumed that a subsystem with a particular category complying with ISO 13849-1:1999 has the associated hardware fault tolerance and safe failure fraction as indicated in Table 6.

NOTE To achieve a required SIL, it is also necessary to fulfil the requirements according to probability of dangerous failure and systematic safety integrity.

Table 6 – Architectural constraints: SILCL relating to categories

Category	Hardware fault tolerance	SFF	Maximum SIL claim limit according to architectural constraints
	It is assumed that subsystems with the stated category have the characteristics given below		
1	0	< 60 %	See Note 1
2	0	60 % – 90 %	SIL 1 (see Note 2)
3	1	< 60 %	SIL 1
	1	60 % – 90 %	SIL 2
4	>1	60 % – 90 %	SIL 3 (see Note 3)
	1	> 90 %	SIL 3 (see Note 4)

NOTE 1 Subsystems that have a SFF of <60% but are designed in accordance with Category 1 of ISO 13849-1:1999 and validated in accordance with ISO 13849-2:2003 are assumed to achieve a SILCL of SIL1.

NOTE 2 The case for Category 2 where SFF is > 90 % is assumed not to be achieved by the design requirements of ISO 13849-1:1999.

NOTE 3 The diagnostic coverage is assumed to be less than 90 % for Category 4 subsystems where greater than single hardware fault tolerance (i.e. accumulated faults) is considered.

NOTE 4 Category 4 requires a SFF of more than 90 % but less than 99 % when single hardware fault tolerance is considered.

NOTE 5 Category B in accordance with ISO 13849-1:1999 is not considered sufficient to achieve SIL 1.

6.7.7 Estimation of safe failure fraction (SFF)

6.7.7.1 The SFF shall be estimated where it is required to determine the SILCL due to architectural constraints.

6.7.7.2 To estimate the SFF, an analysis (e.g. fault tree analysis, failure mode and effects analysis) of each subsystem shall be performed to determine all relevant faults and their corresponding failure modes. Whether a failure is a safe or a dangerous failure depends on the SRECS and the intended safety-related control functions, including fault reaction function. The probability of each failure mode shall be determined based on the probability of the associated fault(s) taking into account the intended use and may be derived from sources such as:

a) dependable failure rate data collected from field experience by the manufacturer and relevant to the intended use;

b) component failure data from a recognised industry source (see references in Annex D) and relevant to the intended use;

c) failure mode data given in Annex D;

d) failure rate data derived from the results of testing and analysis.

EXCEPTION: For a subsystem with a hardware fault tolerance of zero and where fault exclusions have been applied to faults that could lead to a dangerous failure, then the SILCL due to architectural constraints of that subsystem is constrained to a maximum of SIL 2.

6.7.7.3 The use of fault exclusions shall be justified (e.g. by analysis) and documented.

NOTE It is permissible to exclude faults in accordance with 3.3 and D.5 of ISO 13849-2:2003.

6.7.8 Requirements for the probability of dangerous random hardware failures of subsystems

6.7.8.1 General requirements

6.7.8.1.1 The probability of dangerous random hardware failure shall be equal to or less than the target failure value as specified in the subsystem safety requirements specification (see 6.6.2.1.7).

6.7.8.1.2 The probability of dangerous failure of each subsystem due to random hardware failures to perform the allocated function blocks shall be estimated taking into account:

a) the architecture of the subsystem as it relates to the allocated function blocks under consideration;

NOTE 1 This involves deciding whether there is hardware fault tolerance or not.

b) the rate of failure of each subsystem element in any modes which would cause a dangerous failure of the subsystem but which are detected by diagnostic tests (see 6.3);

c) the rate of failure of each subsystem element in any modes which would cause a dangerous failure of the subsystem which are undetected by the diagnostic tests (see 6.3);

d) the susceptibility of the subsystem to common cause failures which would cause a dangerous failure of the subsystem (see Notes 2 and 3);

NOTE 2 Where comparison of redundant components is used for fault detection, failure of the fault detection means can occur when the redundant components fail at the same time in the same mode. This can occur due to a common cause referred to as a common cause failure (CCF) that is expressed as a beta (ß) factor.
A simplified approach to estimate the susceptibility to common cause failures is given in 6.7.8.3. For further guidance on quantifying the effect of hardware-related common cause failures, see also IEC 61508-6, Annex D.

e) the diagnostic coverage of the diagnostic tests (see 3.2.38) and the associated diagnostic test interval;

f) the intervals at which proof tests are undertaken to reveal dangerous faults which are undetected by diagnostic tests and/or the mission time of the subsystem element(s) which should not be exceeded in order to maintain the validity of the information given in items b) and c);

g) the repair times for detected faults where the subsystem is designed for online repair.

NOTE 3 The maximum repair time will constitute one part of the time to restoration (see IEV 191-10-05), including also the time taken to detect a fault and any time period during which repair is not possible (see IEC 61508-6, Annex B for an example of how the mean time to restoration can be used to calculate the probability of failure). For situations where the repair can only be carried out during a specific period of time, while the machine is shut down and in a safe state, it is particularly important that full account is taken of the time period when no repair can be carried out, especially when this is relatively large.

NOTE 4 A simplified approach for the estimation of the probability of dangerous random hardware failure of subsystems is given in 6.7.8.2. Other methods are available and the most appropriate method will depend on the circumstances. Available methods include:
a) fault tree analysis (see B.6.6.5 of IEC 61508-7 and IEC 61025);
b) Markov models (see C.6.4 of IEC 61508-7 and IEC 61165-13);
c) reliability block diagrams (see C.6.5 of IEC 61508-7).

NOTE 5 Failures due to common cause effects and data communication processes can result from effects other than actual failures of hardware components (e.g. electromagnetic interference, software errors, etc.). See 6.7.9.

6.7.8.1.3 For subsystems or subsystem elements where the probability of failure is given in relation to a number of operating cycles, these values shall be transformed into time-related values by using the specified duty cycle for the relevant SRCFs (see 5.2.3).

6.7.8.1.4 The diagnostic test interval of any subsystem having a hardware fault tolerance of more than zero shall be such as to enable the subsystem to meet the requirement for the probability of random hardware failure (see 6.3.1).

NOTE This diagnostic test interval should be such that a fault is detected before the occurrence of a subsequent fault that may lead to dangerous failure of the subsystem and exceeds the target failure measure.

6.7.8.1.5 The diagnostic test interval of any subsystem having a hardware fault tolerance of zero shall be such that the requirements of 6.3.2 are fulfilled.

6.7.8.1.6 Where a low complexity subsystem is designed according to ISO 13849-1:1999 and validated according to ISO 13849-2:2003 and also meets the requirements for architectural constraints (see 6.7.6) and systematic safety integrity (see 6.7.9), the threshold values of probability of dangerous failure (PFH_D) given in Table 7 can be used to estimate the hardware safety integrity (see 6.6.3.2).

Table 7 – Probability of dangerous failure

Category	Hardware fault tolerance	DC	PFH_D threshold values (per hour) that can be claimed for the subsystem PFH_D (MTTF subsystem, T_{test}, DC) (See Note 1)
	It is assumed that subsystems with the stated category have the characteristics given below		
1	0	0 %	To be provided by supplier or use generic data (see Annex D)
2	0	60 % – 90 %	≥ 10^{-6}
3	1	60 % – 90 %	≥2 x 10^{-7}
4	>1	60 % – 90 %	≥ 3 x 10^{-8}
	1	> 90 %	≥ 3 x 10^{-8}

NOTE 1 The PFH_D threshold value is a function of the subsystem MTTF (to be derived by the subsystem manufacturer or from relevant component data handbooks), test/check cycle time as specified in the safety requirements specification (this information is also required for subsystem validation in accordance with ISO 13849-2:2003, 3.5) and the diagnostic coverage as shown in this table (these values are based on the requirements of the categories described in ISO 13849-1:1999).

NOTE 2 Category B in accordance with ISO 13849-1:1999 cannot be considered sufficient to achieve SIL 1.

6.7.8.2 Simplified approach for the estimation of probability of dangerous random hardware failures of subsystems

6.7.8.2.1 General

This subclause describes a simplified approach to the estimation of probability of dangerous random hardware failures for a number of basic subsystem architectures and gives formulae that can be used for subsystems assembled from either low complexity subsystem elements or complex subsystem elements. The formulae are in themselves a simplification of reliability analysis theory and are intended to provide estimates that are biased towards the safe direction. The precondition for the validity for all formulae given in this subclause is that $1 >> \lambda \times T_1$, where T_1 is the smaller of the proof test interval or the lifetime, and the subsystem is operating in the "high demand or continuous mode" (see 3.2.27). See also 6.8.6.

NOTE 1 The results that are obtained represent a limitation upon probability of dangerous random hardware failures of subsystems and where this is unacceptable, it is possible to apply more accurate modelling techniques (see 6.7.8.1.1).

NOTE 2 For equations (A) to (D) given in 6.7.8.2 constant and sufficiently low ($1>> \lambda \times T$) failure rates (λ) of the subsystem elements are assumed (this means that the mean time to dangerous failure has to be much greater than the proof test interval or the lifetime of the subsystem). Therefore, the following basic equations can be used:

- λ = 1/MTTF, where MTTF is expressed in hours.

For electromechanical devices the failure rate is determined using the B_{10} value and the number of operating cycles C (expressed as the number of operating cycles per hour) of the application as specified (see 5.2.3).

- λ = 0,1 x C/B_{10}.

NOTE 3 Terms used are as follows:

- $\lambda = \lambda_S + \lambda_D$; where λ_S is the rate of safe failure and λ_D is the rate of dangerous failure.
- $PFH_D = \lambda_D \times 1h$; average probability of dangerous failure within one hour.
- T_2: diagnostic test interval.
- T_1: proof test interval or lifetime whichever is the smaller.

6.7.8.2.2 Basic subsystem architecture A: zero fault tolerance without a diagnostic function

In this architecture, any dangerous failure of a subsystem element causes a failure of the SRCF. For architecture A, the probability of dangerous failure of the subsystem is the sum of the probabilities of dangerous failure of all subsystems elements:

$$\lambda_{DssA} = \lambda_{De1} + \ldots + \lambda_{Den} \qquad \text{(A)}$$

$$PFH_{DssA} = \lambda_{DssA} \times 1h$$

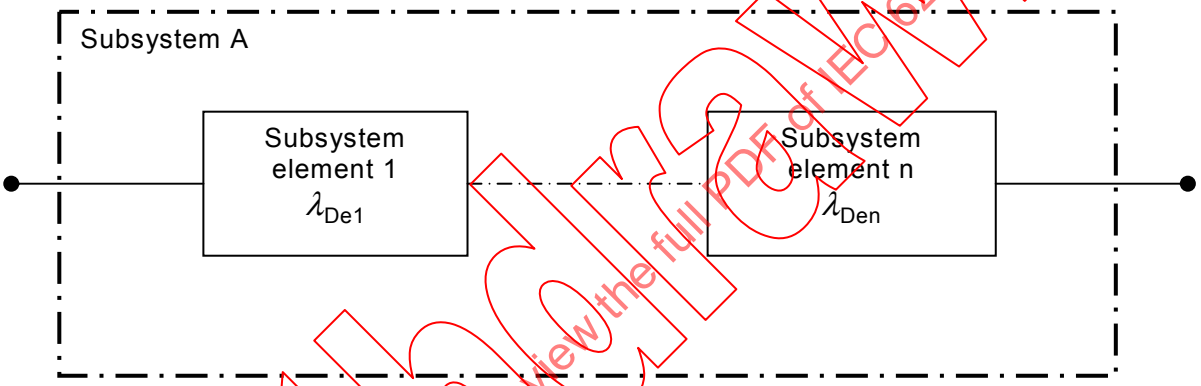


Figure 6 – Subsystem A logical representation

NOTE Figure 6 is a logical representation of the subsystem A architecture and should not be interpreted as its physical implementation.

6.7.8.2.3 Basic subsystem architecture B: single fault tolerance without a diagnostic function

This architecture is such that a single failure of any subsystem element does not cause a loss of the SRCF. Thus, there would have to be a dangerous failure in more than one element before failure of the SRCF can occur. For architecture B, the probability of dangerous failure of the subsystem is:

$$\lambda_{DssB} = (1 - \beta)^2 \times \lambda_{De1} \times \lambda_{De2} \times T_1 + \beta \times (\lambda_{De1} + \lambda_{De2})/2 \qquad \text{(B)}$$

$$PFH_{DssB} = \lambda_{DssB} \times 1h$$

where
T_1 is the proof test interval or lifetime whichever is the smaller.
β is the susceptibility to common cause failures.

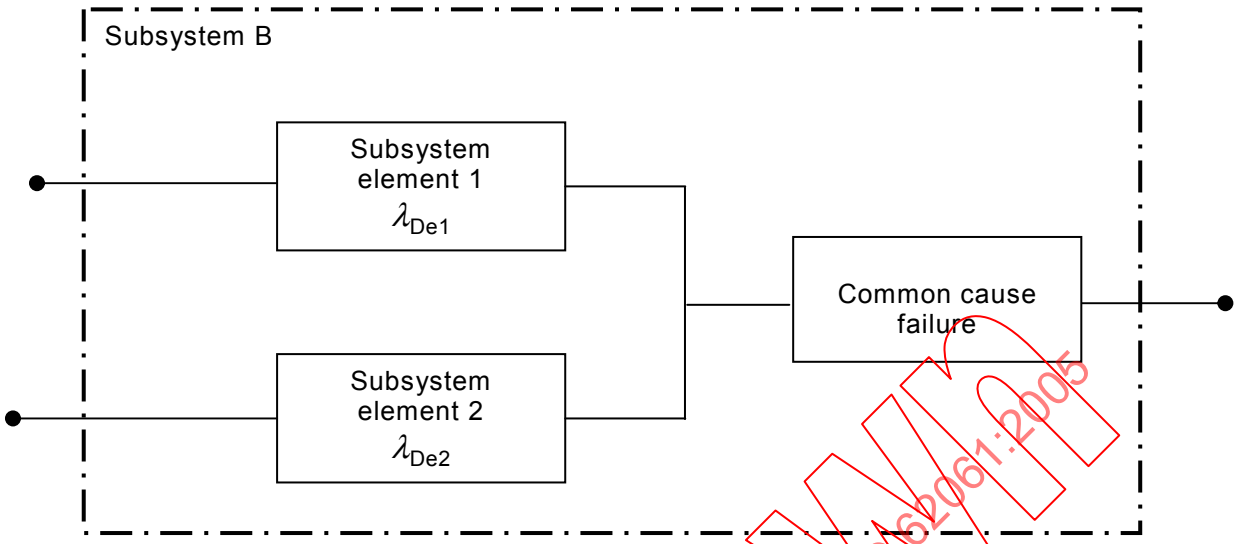


Figure 7 – Subsystem B logical representation

NOTE Figure 7 is a logical representation of the subsystem B architecture and should not be interpreted as its physical implementation.

6.7.8.2.4 Basic subsystem architecture C: zero fault tolerance with a diagnostic function

Any undetected dangerous fault of the subsystem element leads to a dangerous failure of the SRCF. Where a fault of a subsystem element is detected, the diagnostic function(s) initiates a fault reaction function (see 6.3.2). For architecture C, the probability of dangerous failure of the subsystem is:

$$\lambda_{DssC} = \lambda_{De1}(1 - DC_1) + \ldots + \lambda_{Den}(1 - DC_n) \quad \text{(C)}$$

$$PFH_{DssC} = \lambda_{DssC} \times 1h$$

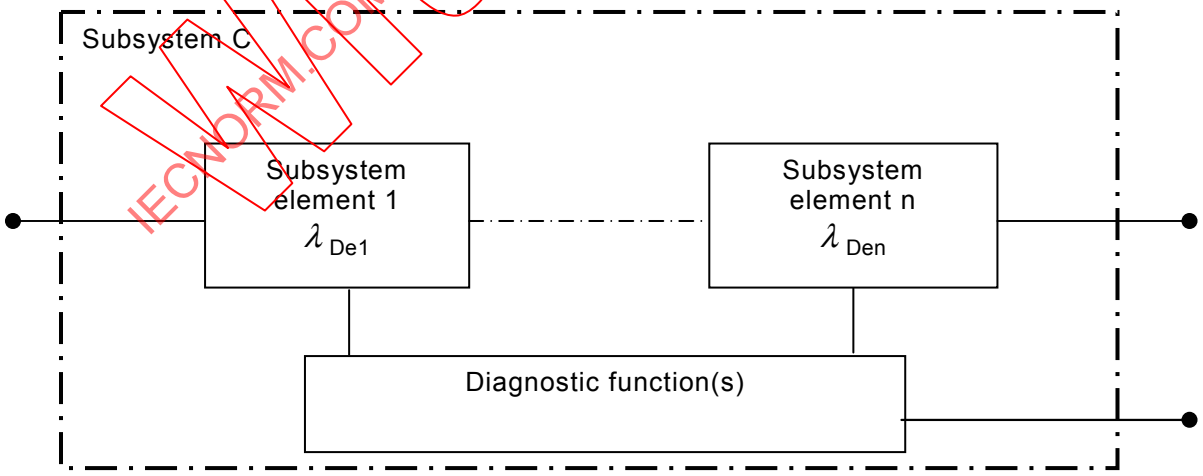


Figure 8 – Subsystem C logical representation

NOTE Figure 8 is a logical representation of the subsystem C architecture and should not be interpreted as its physical implementation. The diagnostic function shown may be carried out by

- the subsystem which requires diagnostics; or
- other subsystems of the SRECS; or
- subsystems not involved in the performance of the safety-related control function.

6.7.8.2.5 Basic subsystem architecture D: single fault tolerance with a diagnostic function(s)

This architecture is such that a single failure of any subsystem element does not cause a loss of the SRCF, where

T_2 is the diagnostic test interval;

T_1 is the proof test interval or lifetime whichever is the smaller.

β is the susceptibility to common cause failures; $\lambda_D = \lambda_{DD} + \lambda_{DU}$; where λ_{DD} is the rate of detectable dangerous failures and λ_{DU} is the rate of undetectable dangerous failure.

$\lambda_{DD} = \lambda_D \times DC$

$\lambda_{DU} = \lambda_D \times (1 - DC)$

For subsystem elements of different design:

λ_{De1} is the dangerous failure rate of subsystem element 1;

DC_1 is the diagnostic coverage of subsystem element 1;

λ_{De2} is the dangerous failure rate of subsystem element 2;

DC_2 is the diagnostic coverage of subsystem element 2.

$$\lambda_{DssD} = (1-\beta)^2 \{[\lambda_{De1} \times \lambda_{De2} \times (DC_1 + DC_2)] \times T_2/2 + [\lambda_{De1} \times \lambda_{De2} \times (2 - DC_1 - DC_2)] \times T_1/2\} + \beta \times (\lambda_{De1} + \lambda_{De2})/2 \quad \text{(D.1)}$$

$PFH_{DssD} = \lambda_{DssD} \times 1h$

For subsystem elements of the same design:

λ_{De} is the dangerous failure rate of subsystem element 1 or 2;

DC is the diagnostic coverage of subsystem element 1 or 2.

$$\lambda_{DssD} = (1-\beta)^2 \{[\lambda_{De}^2 \times 2 \times DC] \times T_2/2 + [\lambda_{De}^2 \times (1 - DC)] \times T_1\} + \beta \times \lambda_{De} \quad \text{(D.2)}$$

$PFH_{DssD} = \lambda_{DssD} \times 1h$

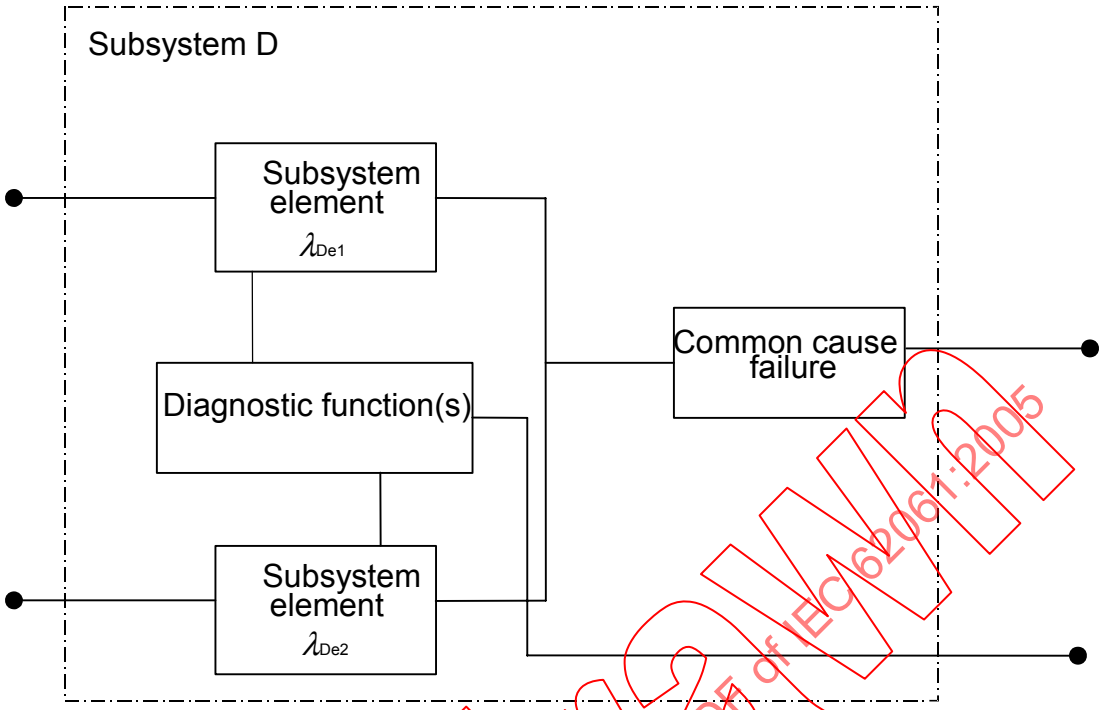


Figure 9 – Subsystem D logical representation

NOTE 1 Figure 9 is a logical representation of the subsystem D architecture and should not be interpreted as its physical implementation. The diagnostic function(s) shown may be carried out by

– the subsystem which requires diagnostics; or

– other subsystems of the SRECS; or

– subsystems not involved in the performance of the safety-related control function.

NOTE 2 The fault reaction for this subsystem is assumed to be termination of the relevant operation as required in 6.3.1. When an online repair is incorporated in the design where the fault reaction is to report the fault but not to terminate the relevant operation, a new PFH_D for the subsystem after occurrence of a first fault should be determined for the remaining architecture.

6.7.8.3 Simplified approach to estimation of contribution of common cause failure (CCF)

6.7.8.3.1 Knowledge of the susceptibility of a subsystem to CCF is required to contribute to the estimation of the probability of dangerous random hardware failure of a subsystem (see 6.7.8.1).

6.7.8.3.2 Where a redundant architecture is used to achieve the required probability of dangerous random hardware failure of a subsystem and a CCF(s) can remove the effect of that redundancy, the probability of dangerous random hardware failure based on the probability of occurrence of the common cause shall be added to the probability of dangerous random hardware failure of a subsystem based on the use of redundancy.

6.7.8.3.3 The probability of occurrence of the CCF will usually be dependent upon a combination of technology, architecture, application and environment. The use of Annex F will be effective in avoiding many types of CCF.

6.7.8.3.4 Annex F contains a scoring table and an associated methodology that can be used to estimate the effectiveness of measures applied in the design of a subsystem to limit susceptibility to CCF.

6.7.9 Requirements for systematic safety integrity of subsystems

The SILCL due to systematic safety integrity of a subsystem is up to SIL 3 when the requirements in 6.7.9.1 and 6.7.9.2 are fulfilled.

NOTE These requirements are applicable at the 'subsystem level' where subsystem elements are interconnected to realise a subsystem. For other requirements relevant to SRECS realisation, see 6.4.

6.7.9.1 Requirements for the avoidance of systematic failures

6.7.9.1.1 The following measures shall be applied:

a) proper selection, combination, arrangements, assembly and installation of components, including cabling, wiring and any interconnections: apply manufacturer's application notes and use of good engineering practice;

b) use of the subsystem and subsystem elements within the manufacturer's specification and installation instructions;

c) compatibility: use components with compatible operating characteristics;

d) withstanding specified environmental conditions: design the subsystem so that it is capable of working in all expected environments and in any foreseeable adverse conditions, for example temperature, humidity, vibration and electromagnetic interference (EMI) (see ISO 13849-2, Clause D.1);

e) use of components that are in accordance with an appropriate standard and have their failure modes well-defined: to reduce the risk of undetected faults by the use of components with specific characteristics;

f) use of suitable materials and adequate manufacturing: selection of material, manufacturing methods and treatment in relation to, for example stress, durability, elasticity, friction, wear, corrosion, temperature, conductivity, dielectric strength;

g) correct dimensioning and shaping: consider the effects of, for example, stress, strain, fatigue, temperature, surface roughness, manufacturing tolerances.

6.7.9.1.2 In addition, one or more of the following measures shall be applied taking into account the complexity of the subsystem:

a) hardware design review (e.g. by inspection or walk-through): to reveal by reviews and/or analysis discrepancies between the specification and implementation;

NOTE 1 In order to reveal discrepancies between the specification and implementation, any points of doubt or potential weak points concerning the realisation, the implementation and the use of the product are documented so they may be resolved; taking into account that on an inspection procedure the author is passive and the inspector is active whilst on a walk-through procedure the author is active and the inspector is passive.

b) computer-aided design tools capable of simulation or analysis: perform the design procedure systematically and include appropriate automatic construction elements that are already available and tested;

NOTE 2 The integrity of these tools can be demonstrated by specific testing, or by an extensive history of satisfactory use, or by independent verification of their output for the particular subsystem that is being designed. See 6.11.3.4.

c) simulation: perform a systematic simulation of a subsystem design in terms of both the functional performance and the correct dimensioning of their components.

NOTE 3 The function of the subsystem can be simulated on a computer via a software behavioural model (see 6.11.3.4) where individual components of the circuit each have their own simulated behaviour, and the response of the subsystem in which they are connected is examined by looking at the marginal data of each component.

6.7.9.2 Requirements for the control of systematic failures

6.7.9.2.1 The following measures shall be applied:

a) measures to control the effects of insulation breakdown, voltage variations and interruptions, overvoltage and undervoltage: subsystem behaviour in response to insulation breakdown, voltage variations and interruptions, overvoltage and undervoltage conditions shall be pre-determined so that the subsystem can achieve or maintain a safe state of the SRECS;

NOTE 1 See also relevant requirements of IEC 60204-1. In particular:

– overvoltage should be detected early enough so that all outputs can be switched to a safe condition by the power-down routine or a switch-over to a second power unit; and/or

– the control circuit voltage should be monitored and a power-down initiated, or a switch-over to a second power unit, if it is not within its specified range; and/or

– overvoltage or undervoltage should be detected early enough so that the internal state can be saved in non-volatile memory (if necessary), so that all outputs can be set to a safe condition by the power-down routine or a switch-over to a second power unit.

b) measures to control or avoid the effects of the physical environment (for example, temperature, humidity, water, vibration, dust, corrosive substances, electromagnetic interference and its effects): subsystem behaviour in response to the effects of the physical environment shall be pre-determined so that the SRECS can achieve or maintain a safe state of the machine. See also IEC 60204-1;

c) measures to control or avoid the effects of temperature increase or decrease, if temperature variations can occur: the subsystem should be designed so that, for example, over-temperature can be detected before it begins to operate outside specification.

NOTE 2 Further information can be found in IEC 61508-7, Clause A.10.

6.7.9.2.2 In addition, the following measures, as appropriate, shall be applied for the control of systematic failures:

– failure detection by on-line monitoring;
– tests by comparison of redundant hardware;
– diverse hardware;
– operation in the positive mode (e.g. a limit switch is pushed when a guard is opened);
– oriented mode of failure;
– over-dimensioning by a suitable factor, where the manufacturer can demonstrate that derating will improve reliability.

NOTE 1 Where over-dimensioning is appropriate, an over-dimensioning factor of at least 1,5 should be used.

NOTE 2 Further information can be found in ISO 13849-2, Clause D.3.

6.7.10 Subsystem assembly

The subsystem elements shall be combined to form the subsystem in accordance with 6.7.4.3.1.2 and the detailed design documented.

6.8 Realisation of diagnostic functions

6.8.1 Each subsystem shall be provided with associated diagnostic functions that are necessary to fulfil the requirements for architectural constraints (6.7.6) and the probability of dangerous random hardware failures (6.7.8).

6.8.2 The diagnostic functions are considered as separate functions that may have a different structure than the SRCF and may be performed by

– the same subsystem which requires diagnostics; or

– other subsystems of the SRECS; or

– subsystems of the SRECS not performing the SRCF.

NOTE See also Note 3 of 6.6.2.1.

6.8.3 Diagnostic functions shall satisfy the following that are applicable to their associated SRCFs:

– requirements for the avoidance of systematic failures (see 6.7.9.1); and

– requirements for the control of systematic failures (see 6.7.9.2).

6.8.4 The probability of failure of the SRECS diagnostic function(s) shall be taken into account when estimating the probability of dangerous failure of the SRCF.

NOTE 1 See also Note 3 of 6.6.2.1.

NOTE 2 Timing constraints applicable to the testing of the subsystem performing a diagnostic function may differ from those applicable to SRCFs and, in general, the test interval should meet requirements applicable to a subsystem with a hardware fault tolerance of 1.

NOTE 3 Failure of a diagnostic function(s) should be detected and an appropriate reaction should be taken to ensure that the contribution of the diagnostic function to the safety integrity of the SRCF is maintained. The failure of a diagnostic function(s) may be detected by on-line testing, cross-checking by redundant hardware, etc.

6.8.5 A clear description of the SRECS diagnostic function(s), their failure detection/reaction, and an analysis of their contribution towards the safety integrity of the associated SRCFs shall be provided.

6.8.6 To apply the simplified approach for the estimation of probability of dangerous random hardware failures of subsystems (6.7.8.2), the following shall apply:

– where a SRECS diagnostic function(s) is necessary to achieve the required probability of dangerous random hardware failure and the subsystem has a hardware fault tolerance of zero, then the fault detection and specified fault reaction shall be performed before the hazard due to this fault can occur; and

– SRECS diagnostic function(s) shall as a minimum be implemented so that the probability of random hardware failure and the systematic safety integrity are the same as those specified for the corresponding SRCF(s); or

NOTE 1 Architectural constraints on hardware safety integrity need not apply to the realisation of diagnostic function(s).

– where the probability of dangerous random hardware failure is of an order of magnitude greater than that specified for the SRCF, then a test shall be performed to determine whether diagnostic function(s) or diagnosing device(s) remain operational. It is assumed that such a test of the diagnostic function(s) or diagnosing device(s) be carried out at a minimum of 10 times during the interval between proof tests applied to the subsystem.

NOTE 2 A test of the diagnostic function(s) should as far as practicable cover 100 % of those parts implementing the diagnostic function(s).

NOTE 3 Where a diagnostic function is implemented by the logic solver of the SRECS it can be unnecessary to perform a separate test of the diagnostic function as its failure can be revealed as a failure of the SRCF.

NOTE 4 A test can be performed by either external means (e.g. test equipment) or internal dynamic checks (e.g., embedded within the logic solver) of the SRECS.

6.9 Hardware implementation of the SRECS

The SRECS shall be implemented in accordance with the documented SRECS design.

6.9.1 SRECS interconnection

6.9.1.1 The SRECS shall be interconnected so as to satisfy appropriate parts of the SRECS safety requirements specification and those requirements relevant to conductors, cabling and wiring practices in IEC 60204-1.

6.9.1.2 Measures for avoiding and controlling failures of interconnecting conductors and cables shall be realised in accordance with 6.4.1 and 6.4.2.

6.10 Software safety requirements specification

6.10.1 General

Where software is to be used in any part of a SRECS implementing a safety-related control function(s), a software safety requirements specification shall be developed and documented.

6.10.2 Requirements

6.10.2.1 A software safety requirements specification shall be developed for each subsystem on the basis of the SRECS specification and architecture.

6.10.2.2 The specification of the requirements for software safety for each subsystem shall be derived from (1) the specified safety requirements of the SRCF, (2) the requirements resulting from the SRECS architecture and (3) any requirements of the functional safety plan (see 4.2). This information shall be made available to the application software developer.

6.10.2.3 The specification of the requirements for application software safety shall be sufficiently detailed to allow the design and implementation of the SRECS to achieve the required safety integrity, and to allow verification.

6.10.2.4 The application software developer shall review the information in the specification to ensure that the requirements are adequately specified. In particular, the software developer shall conform to this standard by including the following:

– SRCFs;
– configuration or architecture of the system;
– capacity and response time performance;
– equipment and operator interfaces;
– all relevant modes of operation of the machine as specified in the safety requirements specification;
– diagnostic tests of external devices (e.g. sensors and final elements).

6.10.2.5 The specified requirements for software safety shall be expressed and structured so that they are:

– clear, verifiable, testable, maintainable and operable, commensurate with the safety integrity level;

– traceable back to the specification of the safety requirements of the SRECS;

– free of ambiguous terminology and descriptions.

6.10.2.6 The software safety requirements specification shall express the required properties of each subsystem by providing information allowing proper equipment selection. The requirements for the following software-based SRCFs shall be specified:

– the logic (i.e. the functionality) of all function blocks assigned to each subsystem;

– input and output interfaces assigned for each function block;

– format and value ranges of input and output data and their relation to function blocks;

– relevant data to describe any limits of each function block, for example maximum response time, limit values for plausibility checks;

– diagnostic functions of other devices within the SRECS (e.g. sensors and final elements) to be implemented by that subsystem;

– functions that enable the machine to achieve or maintain a safe state;

– functions related to the detection, annunciation and handling of faults;

– functions related to the periodic testing of SRCFs on-line and off-line;

– functions that prevent unauthorized modification of the SRECS;

– interfaces to non SRCFs; and

– capacity and response time performance.

NOTE Interfaces include both off-line and online programming facilities.

6.10.2.7 Where appropriate, semi-formal methods such as logic, function-block, or sequence diagrams shall be used in the documentation.

NOTE Guidance on software documentation is given in IEC 61506, ISO/IEC 15910 and ISO/IEC 9254.

6.11 Software design and development

6.11.1 Embedded software design and development

Embedded software incorporated into subsystems shall comply with IEC 61508-3 as appropriate for the required SIL.

NOTE 1 See also 6.7.3.2

NOTE 2 Annex C is provided to assist in the design and development of embedded software used to implement SRCFs within a SRECS.

6.11.2 Software based parameterization

6.11.2.1 Software based parameterization of safety-related parameters shall be considered as a safety-related aspect of SRECS design that is described in the software safety requirements specification (see 6.10). Parameterization shall be carried out using a dedicated tool provided by the supplier of the SRECS or the related subsystem(s). This tool shall have its own identification (name, version, etc.). The parameterization tool shall prevent unauthorized modification, for example by using a password.

6.11.2.2 The integrity of all data used for parameterization shall be maintained. This shall be achieved by applying measures to

– control the range of valid inputs;
– control data corruption before transmission;
– control the effects of errors from the parameter transmission process;
– control the effects of incomplete parameter transmission; and
– control the effects of faults and failures of hardware and software of the tool used for parameterization.

6.11.2.3 The tool used for parameterization shall fulfil the following requirements:

– all relevant requirements for a subsystem according to this standard to ensure correct parameterization; or
– a special procedure shall be used for setting the safety-related parameters. This procedure shall include confirmation of input parameters to the SRECS by either:
  • retransmitting the modified parameters to the parameterization tool; or
  • other means to confirm the integrity of the parameters;

  and subsequent confirmation (e.g. by a suitably skilled person and an automatic check by a parameterization tool);

  NOTE This is of particular importance where parameterization is carried out using a device not specifically intended for this purpose (e.g. personal computer or equivalent).
– the software modules used for encoding/decoding within the transmission/retransmission process and software modules used for visualization of the safety-related parameters to the user shall as a minimum use diversity in function(s) to avoid systematic failures.

6.11.2.4 Documentation of software based parameterization shall indicate data used (e.g. pre-defined parameter sets) and information necessary to identify the parameters associated with the SRECS, the person(s) carrying out the parameterization together with other relevant information such as date of parameterization.

6.11.2.5 The following verification activities shall be applied for software based parameterization:

– verification of the correct setting for each safety-related parameter (minimum, maximum and representative values);
– verification that the safety-related parameters are checked for plausibility, for example by detection of invalid values, etc.;
– verification that unauthorized modification of safety-related parameters is prevented;

– verification that the data/signals for parameterization are generated and processed in such a way that faults cannot lead to a loss of SRCF(s).

NOTE This is of particular importance where the parameterization is carried out using a device not specifically intended for this purpose (e.g. personal computer or equivalent),

6.11.3 Application software design and development

NOTE This subclause is based on IEC 61508-3.

6.11.3.1 General requirements

6.11.3.1.1 The requirements of IEC 61508-3 apply to Full Variability Languages (FVL). The following requirements shall be applied to applications software based upon Limited Variability Languages (LVL).

6.11.3.1.2 The outcome of the activities performed during the application software development shall be verified at appropriate stages.

6.11.3.1.3 The design method and application language chosen to satisfy the required SIL of the SRCF shall possess features relevant for the application that facilitate:

a) abstraction, modularity and other features which control complexity; wherever possible, the software shall be based on well-proven logic functions which may include user library functions and well-defined rules for linking logic functions;

b) expression of

   – functionality, ideally as a logical description or as algorithmic functions;

   – information flow between modular elements;

   – sequencing and time related requirements;

   – timing constraints;

   – data structures and their properties, including data types, validity of data ranges;

c) comprehension by developers and others who need to understand the design, both from a functional understanding of the application and from a knowledge of the constraints of the SRECS technology;

d) verification and validation, including structural testing (white box) of the application software, functional testing (black box) of the integrated application program and interface testing (grey box) of the interaction with the SRECS and its application specific hardware configuration;

e) safe modification.

6.11.3.1.4 Testing shall be the main verification method used for the application software. Test planning shall address the following:

– the policy for verification of the integration of software and hardware;

– test cases and test results;

– types of tests to be performed;

– test equipment including tools, support software and configuration description;

– test criteria on which the completion of the test shall be judged;

– physical location(s) (e.g. factory or site);
– dependence on external functionality;
– the amount of test cases necessary; and
– completeness with respect to the related functions or requirements.

6.11.3.1.5 Where the application software is to implement both non-safety and safety-related control functions, then all of the application software shall be treated as safety-related, unless adequate independence between the functions can be demonstrated in the design.

6.11.3.1.6 The design shall include data integrity checks and reasonableness checks at the application layer (e.g. checks in communication links, bounds checking on sensor inputs, bounds checking on data parameters).

6.11.3.1.7 The application software design shall include self-monitoring of control flow and data flow unless such functions are included in the embedded software. On failure detection, appropriate actions shall be performed to achieve or maintain a safe state.

6.11.3.1.8 Where previously developed software library functions are to be used as part of the design, their suitability in satisfying the specification of requirements for software safety shall be justified. Suitability shall be based upon evidence of satisfactory operation in similar applications that have been demonstrated to have similar functionality, or shall be subject to the same verification and validation procedures as would be expected for any newly developed safety-related software. Constraints from the previous software environment (for example operating system and compiler dependencies) shall be evaluated.

6.11.3.1.9 Any modifications or changes to application software shall be subject to an impact analysis that identifies all software modules affected and the necessary re-verification activities to confirm that the software safety requirements specification is still satisfied.

6.11.3.2 Software configuration management

6.11.3.2.1 The functional safety plan shall define the strategy for the development, integration, verification and validation of the software.

6.11.3.2.2 Software configuration management shall:

– ensure that all necessary operations have been carried out to demonstrate that the required software safety integrity has been achieved;
– maintain accurately and with unique identification all documents related to configuration items that are necessary to maintain the integrity of the SRECS. Configuration items shall include at least the following:
  - safety analysis and requirements;
  - software specification and design documents;
  - software source code modules;
  - test plans and results;
  - pre-existing software modules and packages which are to be incorporated into the SRECS;
  - all tools and development environments that are used to create or test, or carry out any action on the application software;

– apply change-control procedures to:
  - prevent unauthorized modifications;
  - document modification requests;
  - analyze the impact of proposed modifications, and to approve or reject the request(s);
  - document the details of, and the authorization for, all approved modifications;
  - document the software configuration at appropriate points in the software development;
– document the following information to permit a subsequent audit: release status, the justification for and approval of all modifications, and the details of the modification;
– formally document the release of the application software. Master copies of the software and all associated documentation shall be kept to permit maintenance and modification throughout the operational lifetime of the released software.

6.11.3.3 Requirements for software architecture

NOTE 1 The software architecture defines the major components and subsystems of system and application software, how they are interconnected, and how the required attributes should be achieved. Examples of application software modules include application functions that are replicated throughout the machine, machine input/output, override and inhibit components, data validity checking and range checks, etc.

NOTE 2 The software architecture is also affected by the underlying architecture of the subsystem provided by the supplier.

6.11.3.3.1 The software architecture design shall be based on the required SRECS safety specification within the constraints of the system architecture of the SRECS and the subsystem design.

6.11.3.3.2 The software architecture design shall:

a) provide a comprehensive description of the internal structure and of the operation of the SRECS and of its components (see Note);
b) include the specification of all identified software components, and the description of connection and interactions between identified components (software and hardware);
c) include the internal design and architecture of all identified components that are not black boxes;
d) identify the software modules included in the SRECS but not used in any mode of safety-related operation.

NOTE It is of particular importance that the architecture documentation be up-to-date and complete with respect to the SRECS.

6.11.3.3.3 A set of techniques and measures necessary during design of the application software to satisfy the specification shall be described and justified. These techniques and measures shall aim at ensuring the predictability of the behaviour of the SRECS and shall be consistent with any constraints identified in the SRECS documentation.

6.11.3.3.4 Measures used for maintaining the integrity of all data shall be described and justified. Such data may include machine input-output data, communications data, operation interface data, maintenance data and internal database data.

6.11.3.4 Requirements for support tools, user manual and application languages

6.11.3.4.1 A suitable set of tools, including configuration management, simulation, and test harness tools shall be selected. The availability of suitable tools (not necessarily those used during initial system development) to supply the relevant services over the lifetime of the SRECS shall be considered. The suitability of the tools shall be explained and documented.

NOTE The selection of development tools depends on the nature of the software development activities, the embedded software and the software architecture. Verification and validation tools such as code analyzers, and simulators may be needed.

6.11.3.4.2 Wherever necessary a sub-set of the application programming language shall be defined.

6.11.3.4.3 Application software shall be designed taking into account constraints and known weaknesses included in the SRECS and subsystem(s) user manuals.

6.11.3.4.4 The application language selected shall either:

– be processed using a translator/compiler which shall be assessed to establish its fitness for purpose;

– be completely and unambiguously defined or restricted to unambiguously defined features;

– correspond to the characteristics of the application;

NOTE An application's characteristics refer, for example to any performance constraints.

– contain features that facilitate the detection of programming mistakes; and

– support features that match the design method;

or, the deficiencies of the language used shall be documented in the software architecture design description and the fitness for purpose of the language shall be explained including additional measures necessary to address the identified shortcomings of the language.

6.11.3.4.5 The procedures for use of the application language shall specify good configuration practice, proscribe unsafe generic software features (for example, undefined language features, unstructured designs, etc), identify checks that can be used to detect errors in the configuration and specify procedures for documentation of the application program. As a minimum, the following information shall be contained in the application program documentation:

a) legal entity (for example company, author(s), etc);

b) description;

c) traceability to application functional requirements;

d) traceability to standard library function;

e) inputs and outputs; and

f) configuration management.

6.11.3.5 Requirements for application software design

6.11.3.5.1 The following information shall be available prior to the start of detailed application software design:

– the software safety requirements specification;

– the description of the software architecture design including identification of the application logic and fault tolerant functionality, a list of input and output data, the generic software modules and support tools to be used and the procedures for configuring the application software with the available materials to provide the application functionality for the defined I/O; and

– the plan for validating the software safety.

6.11.3.5.2 The application software shall be produced in a structured way to achieve:

– modularity of application functionality and of I/O control data;

– testability of functionality (including fault tolerant features) and of internal structure;

– the capacity for safe modification through provision of adequate traceability and explanation of application functions and associated constraints.

6.11.3.5.3 For each major component/subsystem in the description of the application software architecture design (see 6.11.3.5.1), refinement of the design shall be based on:

– functions which are used in a recurring fashion throughout the design;

– mapping of the input/output information of application software modules;

– realisation of the application functions from the generic software functions and I/O mapping.

6.11.3.5.4 The design of each application software module and the structural tests to be applied to each application software module shall be specified.

6.11.3.5.5 Appropriate software and SRECS integration tests shall be specified to ensure that the application program satisfies the specified requirements for application software safety. The following shall be considered:

– the division of the application software into manageable integration sets;

– test cases;

– types of tests to be performed;

– test environment, tools, configuration and programs;

– test criteria on which the completion of the test shall be judged; and

– procedures for corrective action on failure of test.

6.11.3.6 Requirements for application code development

6.11.3.6.1 The application software shall:

– be readable, understandable and testable;

– satisfy the relevant design principles;

– satisfy the relevant requirements specified during safety planning.

6.11.3.6.2 The application software shall be reviewed to ensure conformance to the specified design, the coding rules and the requirements of safety planning.

NOTE Application software review includes such techniques as software inspections or walk-throughs, code analysis or mathematical proof. These techniques should be used in conjunction with testing and/or simulation to provide assurance that the application software satisfies its associated specification.

6.11.3.7 Requirements for application module testing

NOTE Testing that the application software correctly satisfies its test specification is a verification activity. It is the combination of code review and structural testing that provides assurance that an application software module satisfies its associated specification, i.e. it is verified.

6.11.3.7.1 The configuration of each input and output point shall be checked through review, testing, or simulation to confirm that the I/O data is mapped to the correct application logic.

6.11.3.7.2 Each software module shall be checked through a process of review, simulation and testing to determine that the intended function is correctly executed and unintended functions are not executed.

6.11.3.7.3 The tests shall be suitable for the specific module being tested and shall:

– ensure each branch of any application software modules is exercised;
– ensure boundary data is exercised;
– ensure sequences are correctly implemented, including relevant synchronisation conditions.

6.11.3.7.4 The results of the application software module testing shall be documented.

6.11.3.7.5 Where software has already been assessed or when a significant amount of positive operating experience is available, the amount of testing may be reduced.

6.11.3.8 Requirements for application software integration testing

NOTE Testing that the software is correctly integrated is a verification activity.

6.11.3.8.1 The application software tests shall verify that all application software modules and components/subsystems interact correctly with each other and with the underlying embedded software to perform their intended function and do not perform unintended functions that could jeopardize any safety function.

6.11.3.8.2 The results of application software integration testing shall be documented, stating:

– the test results; and
– whether the objectives of the test criteria have been met.

6.11.3.8.3 If there is a failure, the reasons for the failure and corrective action taken shall be included in the test results documentation.

6.11.3.8.4 During application software integration, any modification or change to the software shall be subject to a safety impact analysis that shall determine:

– all software modules impacted; and
– all necessary re-verification and re-design activities.

6.12 Safety-related electrical control system integration and testing

NOTE SRECS integration is usually carried out prior to installation but, in some cases, the SRECS integration cannot be carried out until after installation (for example, when the application software development is not finalized until after installation).

6.12.1 General requirements

6.12.1.1 The SRECS shall be integrated according to the specified SRECS design. As part of the integration of all subsystems and subsystem elements into the SRECS, the SRECS shall be tested according to the specified integration tests. These tests shall verify that all modules interact correctly to perform their intended function and not perform unintended functions.

6.12.1.2 The integration of safety-related application software into the SRECS shall include tests that are specified during the design and development phase to ensure the compatibility of the application software with the hardware and embedded software platform such that the functional and safety performance requirements are satisfied.

NOTE 1 This does not imply testing of all input combinations. Testing all equivalence classes (see B.5 and C.5.7 of IEC 61508-7) can suffice. Static analysis, dynamic analysis or failure analysis can reduce the number of test cases to an acceptable level. Use of structured design or semi-formal methods can facilitate testing and verification.

NOTE 2 Use of structured design or semi-formal methods can permit a reduced depth and number of test cases.

NOTE 3 Statistical evidence may also be used to permit a reduced depth and number of test cases.

6.12.1.3 Appropriate documentation of the integration testing of the SRECS shall be produced, stating the test results and whether the objectives and criteria specified during the design and development phase have been met. If there is a failure, the reasons for the failure shall be documented, corrective action taken and re-testing carried out.

6.12.1.4 During the integration and testing, any modification or change to the SRECS shall be subject to an impact analysis that shall identify all components affected and additional verification.

6.12.1.5 During SRECS integration testing, the following shall be documented:

a) the version of the test specification used;

b) the criteria for acceptance of the integration tests;

c) the version of the SRECS being tested;

d) the tools and equipment used along with calibration data;

e) the results of each test;

f) all discrepancies between expected and actual results;

g) the analysis made and the decisions taken on whether to continue the test or issue a change request, in the case where discrepancies occur.

6.12.2 Tests to determine systematic safety integrity during SRECS integration

6.12.2.1 Testing to reveal faults and to avoid failures during integration of the application software and hardware shall be applied. During the tests, reviews shall be carried out to see whether the specified characteristics of the SRECS have been achieved.

6.12.2.2 The following tests shall be applied:

a) functional tests where data that adequately characterizes the operation are applied to the SRECS. The outputs shall be observed and their response is compared with that given by the specification. Deviations from the specification and indications of an incomplete specification shall be documented; and

b) dynamic tests to verify the dynamic behaviour under realistic functional conditions and reveal failures to meet the SRECS functional specification, and to assess utility and robustness of the SRECS.

NOTE The functions of a system or program are executed in a specified environment with specified test data that has been derived systematically from the SRECS SRS according to established criteria. This exposes the behaviour of the SRECS and permits a comparison with the specification. The aim is to determine whether the SRECS and/or its subsystems carries out correctly all the functions required by the specification. The technique of forming equivalence classes is an example of the criteria for black-box test data. The input data space is subdivided into specific input value ranges (equivalence classes) with the aid of the specification. Test cases are then formed from the:

- data from permissible ranges;

- data from inadmissible ranges;

- data from the range limits;

- extreme values;

- and combinations of the above classes.

Other criteria can be effective in order to select test cases in the various test activities (module test, integration test and system test).

6.13 SRECS installation

6.13.1 Objective

The objectives of the requirements of this subclause are for installation of a SRECS to ensure that it is suitable for its intended use and that it is ready for validation.

6.13.2 Requirements

6.13.2.1 A SRECS shall be installed in accordance with the functional safety plan for the final system validation (see item h) of 4.2.1).

6.13.2.2 Appropriate records of the installation of the SRECS shall be produced, stating any test results. If there is a failure, the reasons for the failure shall be recorded.

7 Information for use of the SRECS

7.1 Objective

Information on the SRECS shall be provided to enable the user to develop procedures to ensure that the required functional safety of the SRECS is maintained during use and maintenance of the machine.

7.2 Documentation for installation, use and maintenance

NOTE 1 See also Clause 6 of ISO 12100-2 that provides general information that should be considered during drafting of accompanying documents.

NOTE 2 One or more items of the documentation described in this subclause may have been developed in order to satisfy other aspects of this standard.

The documentation shall provide information for installation, use and maintenance of the SRECS. This shall include:

a) a comprehensive description of the equipment, installation and mounting.

b) a statement of the intended use of the SRECS and any measures that can be necessary to prevent reasonably foreseeable misuse.

c) information on the physical environment (e.g. lighting, vibration, noise levels, atmospheric contaminants) where appropriate.

d) overview (block) diagram(s) where appropriate.

e) circuit diagram(s).

f) proof test interval or lifetime.

g) a description (including interconnection diagrams) of the interaction (if any) between the SRECS function (s) and the machine electrical control system function(s).

h) a description of the necessary measures to ensure the separation of the SRECS function(s) from the machine electrical control system function(s).

i) a description of the safeguarding and of the means provided to maintain safety where it is necessary to suspend the SRCF(s) (e.g. for manual programming, program verification).

j) programming information, where relevant.

k) description of the maintenance requirements applicable to the SRECS including:

   1) a log for recording the maintenance history of the machine;

   2) the routine actions which need to be carried out to maintain the functional safety of the SRECS, including routine replacement of components with a pre-defined life;

   3) the maintenance procedures to be followed when faults or failures occur in the SRECS, including:

      – procedures for fault diagnosis and repair;

      – procedures for confirming correct operation subsequent to repairs;

      – maintenance recording requirements.

   4) the tools necessary for maintenance and re-commissioning, and the procedures for maintaining the tools and equipment.

   5) a specification for periodic testing, preventive maintenance and corrective maintenance.

NOTE 3 Periodic tests are those functional tests necessary to confirm correct operation and to detect faults.

NOTE 4 Preventive maintenance are the measures taken to maintain the required performance of the SRECS.

NOTE 5 Corrective maintenance includes the measures taken after the occurrence of specific fault(s) that bring the SRECS back into the as-designed state.

8 Validation of the safety-related electrical control system

NOTE Validation of the SRECS may form a part of the validation activities applied to the overall machine design.

8.1 Objective

This Clause specifies the requirements for the validation process to be applied to the SRECS. This includes inspection and testing of the SRECS to ensure that it achieves the requirements stated in the safety requirements specification.

8.2 General requirements

8.2.1 The validation of the SRECS shall be carried out in accordance with a prepared plan (see 4.2).

NOTE 1 In some cases, the safety validation cannot be completed until after installation (for example, when the application software development is not finalized until after installation).

NOTE 2 Validation of a programmable SRECS comprises validation of both hardware and software. The requirements for validation of software are contained in 6.11.3.

8.2.2 Each SRCF specified in the SRECS requirements specification (see 5.2), and all the SRECS operation and maintenance procedures shall be validated by test and/or analysis.

8.2.3 Appropriate documentation of the SRECS safety validation testing shall be produced, which shall state for each SRCF:

a) the version of the SRECS safety validation plan being used and the version of the SRECS tested;

b) the SRCF under test (or analysis), along with the specific reference to the requirement specified during the SRECS safety validation planning;

c) tools and equipment used, along with calibration data;

d) the results of each test;

e) discrepancies between expected and actual results.

8.2.4 When discrepancies occur, corrective action and re-testing shall be carried out as necessary and documented.

8.3 Validation of SRECS systematic safety integrity

8.3.1 The following shall be applied:

a) functional testing to reveal failures during the specification, design and integration phases, and to avoid failures during validation of SRECS software and hardware shall be applied. This shall include verification (e.g., by inspection or test) to assess whether the SRECS is protected against adverse environmental influences and shall be based upon the safety requirements specification;

NOTE 1 See also 6.12.2.1.

b) interference immunity testing to ensure that the SRECS is able to satisfy 5.2.3. Testing for immunity to electromagnetic interference need not be performed on SRECS subsystems or subsystem elements where adequate immunity of the SRECS for its intended application can be shown by analysis;

NOTE 2 The SRECS should, wherever practicable, be loaded with a typical application program, and all the peripheral lines (all digital, analogue and serial interfaces as well as the bus connections and power supply, etc.) are subjected to standard noise signals. In order to obtain a quantitative statement, it is sensible to approach any limits carefully.

c) fault insertion testing shall be performed where the required safe failure fraction ≥ 90 %. These tests shall introduce or simulate faults in the SRECS hardware and the response documented.

8.3.2 In addition, one or more of the following groups of analytical techniques shall be applied taking into account the complexity of the SRECS and the assigned SIL:

a) static and failure analysis;

NOTE 1 This combination of analytical techniques is only considered suitable for SRECS that implement SRCFs with an assigned SIL not exceeding SIL2.

NOTE 2 Further information can be found in IEC 61508-7, B.6.4 and B.6.6.

b) static, dynamic and failure analysis;

NOTE 3 This combination of analytical techniques is not recommended for SRECS that implement SRCFs with an assigned SIL below SIL2.

NOTE 4 Further information can be found in IEC 61508-7, B.6.4, B.6.5 and B.6.6.

c) simulation and failure analysis.

NOTE 5 This combination of analytical techniques is only considered suitable for SRECS that implement SRCFs with an assigned SIL not exceeding SIL2.

NOTE 6 Further information can be found in IEC 61508-7, B.3.6 and B.6.6.

8.3.3 In addition, one or more of the following groups of testing techniques shall be applied taking into account the complexity of the SRECS and the assigned SIL:

a) black-box testing: a test(s) of the dynamic behaviour under real functional conditions to reveal failures to meet the SRECS functional specification, and to assess utility and robustness of the SRECS;

NOTE 1 See also 6.12.2.1.

b) fault insertion (injection) testing shall be performed where the required safe failure fraction <90 %. These tests shall introduce or simulate faults in the SRECS hardware and the results documented;

c) "worst-case" testing shall be performed to assess the extreme (i.e. worst) cases specified by application of the analytical techniques (see 8.3.2);

NOTE 2 The operational capacity of the SRECS and its component dimensioning is tested under worst-case conditions. The environmental conditions are changed to their highest permissible marginal values. The most essential responses of the SRECS are inspected and compared with the safety requirements specification.

d) field experience: the use of field experience from different applications as one of the measures to avoid faults during SRECS validation.

NOTE 3 See also 6.12.2.

9 Modification

9.1 Objective

9.2 This Clause specifies the modification procedure(s) to be applied when modifying the SRECS during design, integration and validation (e.g. during SRECS installation and commissioning).Modification procedure

9.2.1 The request for a modification of the SRECS can arise from, for example:

– safety requirements specification changed;

– conditions of actual use;

– incident/accident experience;

– change of material processed;

– modifications of the machine or of its operating modes.

NOTE Interventions (e.g. adjustment, setting, repairs) on the SRECS made in accordance with the information for use or instruction manual for the SRECS are not considered to be a modification in the context of this Clause.

9.2.2 The reason(s) for the request for a modification of the SRECS shall be documented.

9.2.3 The effect of the requested modification shall be analyzed to establish the effect on the functional safety of the SRECS.

9.2.4 The modification impact analysis and the effect on the functional safety of the SRECS shall be documented.

9.2.5 All accepted modifications that have an effect on the SRECS shall initiate a return to an appropriate design phase for its hardware and/or for its software (e.g. specification, design, integration, installation, commissioning, and validation). All subsequent phases shall then be carried out in accordance with the procedures specified for the specific phases in this standard. All relevant documents shall be revised, amended and reissued accordingly.

9.2.6 Based on those revised documents, a complete action plan shall be prepared and documented before carrying out any modification.

9.3 Configuration management procedures

9.3.1 The configuration management procedures shall be implemented in accordance with the functional safety plan (see 4.2.1) taking into account the following:

a) a plan of each modification process;

b) a documentation of the decision making process and each SRECS-relevant decision;

c) a chronological documentation (e.g. a logbook) of the change request procedures including

- identified hazards which can be affected;
- description of the change request (hardware and/or software);
- reason(s) for the change request (see also 9.2.1);
- decision made (and authorization for each decision);
- the impact analysis;
- re-verification (of each phase) and revalidation;
- all documents affected by the change request activities;
- all activities which were carried out during the change process and the persons/entities who were responsible for them;

d) documentation of the following information to permit a subsequent audit:

- configuration status;
- release status;
- the justification for and approval of all modifications;
- the details of the modification.

9.3.2 The procedures for an appropriate change-control-process should consider the requirements of

a) procedures for defining a unique baseline of each version of the SRECS;

b) definition of all configuration items of a baseline. This shall include at least

1) safety requirements analysis and specification;
2) relevant design documents;
3) hardware and/or software modules;
4) test plans and results;
5) verification and validation reports;
6) pre-existing software components which are to be incorporated into the SRECS;
7) tools and development environments which are used for create and test;
8) accurately maintaining with unique identification of all configuration items which are necessary to maintain the integrity of the SRECS;
9) change control procedures to:
   - prevent unauthorized modifications,
   - document change requests,
   - analyse the impact of a proposed change request and approve or reject the request,
   - document the details of and the authorization for all approved modifications,
   - establish a configuration baseline at appropriate points in the hardware or software development and to document the (partial) integration testing which justifies the baseline,
   - guarantee the composition of and the building of all hardware or software baselines (including the rebuilding of earlier baselines);
10) an effect analysis, which should assess the impact of each change request. This analysis shall include also an appropriate hazard analysis and shall take into account all other modification activities of a SRECS;
11) returning to an appropriate design phase for the hardware and/or software (e.g. specification, design, integration, installation, commissioning and validation) of the SRECS for all accepted modifications that have an impact on the SRECS. All subsequent phases shall then be carried out in accordance with this standard;
12) carrying out of all necessary operations to demonstrate that the required safety integrity has been reached;
13) authorization to carry out the required change request activity shall be dependent on the results of the impact analysis.

9.3.3 The documentation of the change control process shall contain at least

a) a plan of each modification process;
b) a documentation of each of the above mentioned organizational requirements and procedures;
c) a documentation of the decision making process and each SRECS-relevant decision made;
d) a chronological documentation (logbook) of the change request procedures including
   - identified hazards which may be affected;
   - description of the change request (hardware and/or software);
   - reason(s) for the change request (see also 9.2.1);
   - decision made (and authorization for each decision);
   - the impact analysis;
   - reverification (of each phase) and revalidation;

- all documents affected by the change request activities;
- all activities which were carried out during the change process and the persons/entities who were responsible for them;

e) documentation of the following information to permit a subsequent audit:

- configuration status;
- release status;
- the justification for and approval of all modifications;
- the details of the modification.

10 Documentation

10.1 The documentation shall:

– be accurate and concise;

– be easy to understand by those persons having to make use of it;

– suit the purpose for which it is intended;

– be accessible and maintainable.

10.2 The designer of the SRECS should distinguish between the documentation that is relevant to the user and that which is relevant to its design and construction.

10.3 The documents shall have titles or names indicating the scope of the contents.

10.4 The documents shall have a revision index (version numbers) to make it possible to identify different versions of the document.

NOTE See also IEC 82045-1: 2001 for further information on methods that can be used for the management of documentation.

10.5 Table 8 summarizes the information and documentation to be available, where appropriate.

Table 8 – Information and documentation of a SRECS

Information required	Subclause
Functional safety plan	4.2.1
Specification of requirements for SRCFs	5.2
Functional safety requirements specification for SRCFs	5.2.3
Safety integrity requirements specification for SRCFs	5.2.4
SRECS design	6.2.5
Structured design process	6.6.1.2
SRECS design documentation	6.6.1.8
Structure of function blocks	6.6.2.1.1
SRECS architecture	6.6.2.1.5
Subsystem safety requirements specification	6.6.2.1.7
Subsystem realisation	6.7.2.2
Subsystem architecture (elements & their interrelationships)	6.7.4.3.1.2
Fault exclusions claimed when estimating fault tolerance/SFF	6.7.6.1c)/6.7.7.3
Subsystem assembly	6.7.10
Software safety requirements specification	6.10.1

Information required	Subclause
Software based parameterization	6.11.2.4
Software configuration management items	6.11.3.2.2
Suitability of software development tools	6.11.3.4.1
Documentation of the application program	6.11.3.4.5
Results of application software module testing	6.11.3.7.4
Results of application software integration testing	6.11.3.8.2
Documentation of SRECS integration testing	6.12.1.3
Documentation of SRECS installation	6.13.2.2
Documentation for installation, use and maintenance	7.2
Documentation of SRECS validation testing	8.2.4
Documentation for SRECS configuration management	9.3.1

Annex A
(informative)

SIL assignment

A.1 General

This informative Annex provides one example of a qualitative approach for risk estimation and SIL assignment that can be applied to SRCFs for machines. Examples of other techniques that may be used for SIL assignment are given in IEC 61508-5 and will be outlined in a proposed future IEC TC 44 Technical Specification.

NOTE 1 The methodology described in this Annex uses qualitative estimation of risk and is intended to be generally applied for the assignment of a SIL(s) to SRCF(s) of machines. The risk parameters (see Figure A.2) used whilst applying this methodology to particular machines and their specific hazards should be subject to agreement with those involved to ensure that the SRECS can provide adequate risk reduction.

NOTE 2 In a large number of machine specific standards ("C" type standards in CEN) risk estimation has been carried out to select a required Category in accordance with ISO 13849-1:1999 for safety-related parts of machine control systems. It is noted that, for simplification, the following relationships are commonly used: required Category 1 to required SIL 1, required Category 2 to required SIL 1, required Category 3 to required SIL 2 and required Category 4 to required SIL 3. More comprehensive methods of mapping between required Categories of ISO 13849-1:1999 and required SILs used in this international standard are under consideration.

For each specific hazard, the safety integrity requirements should be determined separately for the safety-related control function(s) to be performed by the SRECS (see 5.2.4.2).

Figure A.1 is an example of a practical way of carrying out a risk assessment at a specific hazard leading to estimation of a SIL requirement for a SRECS function. This methodology should be performed for each risk that is to be reduced by a safety-related control function that is to be implemented by a SRECS. Figure A.1 should be used in conjunction with the guidance information in this Annex.

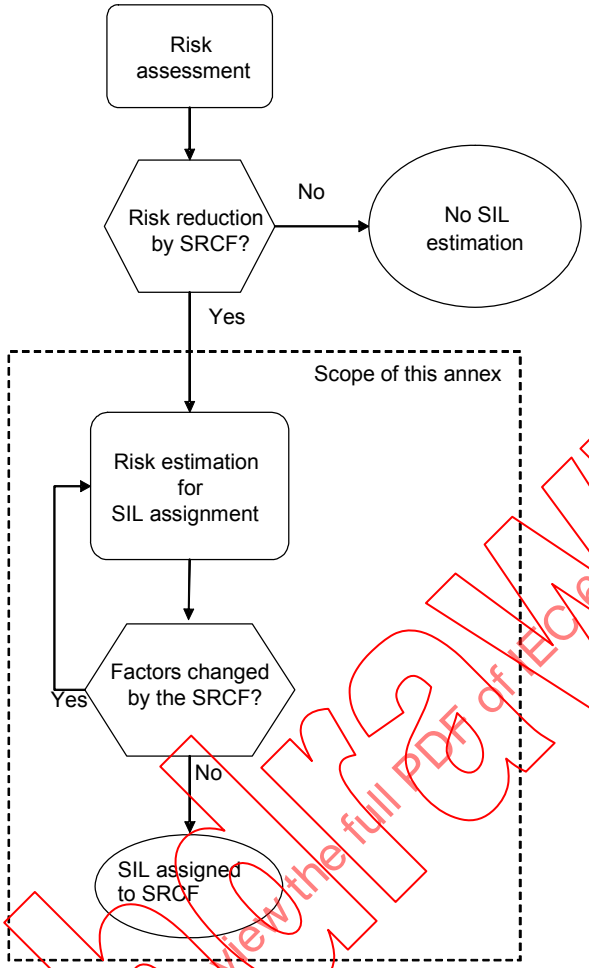


Figure A.1 – Workflow of SIL assignment process

Risk estimation is an iterative process, this means that the process will need to be carried out more than once.

Figure A.1 shows a feedback arrow to risk estimation. This is required because the provision of a particular protective measure to implement an SRCF may have an affect on the risk parameters (e.g. the use of a protective light curtain may result in a greater frequency of access). A failure of the light curtain will then expose the operator to a greater risk than originally envisaged. This requires that the process should be repeated following the same method but using the amended risk parameter(s).

At the end of the process shown in Figure A.1, the SIL estimated is the SIL requirement for the safety-related control function.

A.2 Risk estimation and SIL assignment

A.2.1 Hazard identification/indication

Indicate the hazards, including those from reasonable foreseeable misuse, whose risks are to be reduced by implementing an SRCF. List them in the hazard column in Table A.5.

A.2.2 Risk estimation

Risk estimation should be carried out for each hazard by determining the risk parameters that as shown in Figure A.2 should be derived from the following:

– severity of harm, Se; and
– probability of occurrence of that harm, which is a function of:
  - frequency and duration of the exposure of persons to the hazard, Fr;
  - probability of occurrence of a hazardous event, Pr; and
  - possibilities to avoid or limit the harm, Av.

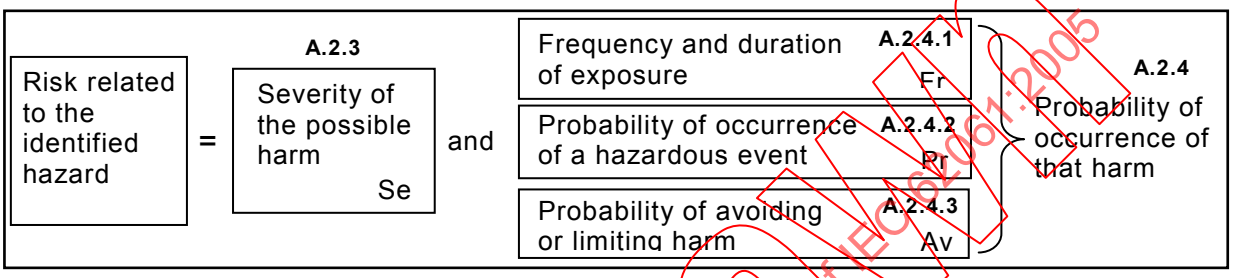


Figure A.2 – Parameters used in risk estimation

The estimates entered into Table A.5 should normally be based on worst-case considerations for the SRCF. However, in a situation where, for example, an irreversible injury is possible but at a significantly lower probability than a reversible one, then each severity level should have a separate line on the table. It may be the case that a different SRCF is implemented for each line. If one SRCF is implemented to cover both lines, then the highest target SIL requirement should be used.

A.2.3 Severity (Se)

Severity of injuries or damage to health can be estimated by taking into account reversible injuries, irreversible injuries and death. Choose the appropriate value of severity from Table A.1 based on the consequences of an injury, where:

4 means a fatal or a significant irreversible injury such that it will be very difficult to continue the same work after healing, if at all;

3 means a major or irreversible injury in such a way that it can be possible to continue the same work after healing. It can also include a severe major but reversible injury such as broken limbs;

2 means a reversible injury, including severe lacerations, stabbing, and severe bruises that requires attention from a medical practitioner;

1 means a minor injury including scratches and minor bruises that require attention by first aid.

Select the appropriate row for consequences (Se) of Table A.1. Insert the appropriate number under the Se column in Table A.5.

Table A.1 – Severity (Se) classification

Consequences	Severity (Se)
Irreversible: death, losing an eye or arm	4
Irreversible: broken limb(s), losing a finger(s)	3
Reversible: requiring attention from a medical practitioner	2
Reversible: requiring first aid	1

A.2.4 Probability of occurrence of harm

Each of the three parameters of probability of occurrence of harm (i.e. Fr, Pr and Av) should be estimated independently of each other. A worst-case assumption needs to be used for each parameter to ensure that SRCF(s) are not incorrectly assigned a lower SIL than is necessary . Generally, the use of a form of task-based analysis is strongly recommended to ensure that proper consideration is given to estimation of the probability of occurrence of harm.

A.2.4.1 Frequency and duration of exposure

Consider the following aspects to determine the level of exposure:

- need for access to the danger zone based on all modes of use, for example normal operation, maintenance; and
- nature of access, for example manual feed of material, setting.

It should then be possible to estimate the average interval between exposures and therefore the average frequency of access.

It should also be possible to foresee the duration, for example if it will be longer than 10 min. Where the duration is shorter than 10 min, the value may be decreased to the number in the row below in Table A.2. This does not apply to frequency of exposure ≤ 1 h, which should not be decreased at any time.

NOTE The duration is related to the performance of activities that are carried out under the protection of the SRCF. The requirements of IEC 60204-1 and ISO 14118 with regard to power isolation and energy dissipation should be applied for major interventions.

This factor does not include consideration of the failure of the SRCF.

Select the appropriate row for frequency and duration of exposure (Fr) of Table A.2. Insert the appropriate number under the Fr column in Table A.5.

Table A.2– Frequency and duration of exposure (Fr) classification

Frequency and duration of exposure (Fr)	
Frequency of exposure	**Frequency, Fr** (see A.2.4.1)
≤ 1 per h	5
< 1 per h to ≥ 1 per day	5
< 1 per day to ≥ 1 per 2 weeks	4
< 1 per 2 weeks to ≥ 1 per year	3
< 1 per year	2

A.2.4.2 Probability of occurrence of a hazardous event

The probability of occurrence of harm should be estimated independently of other related parameters Fr and Av. A worst-case assumption should be used for each parameter to ensure that SRCF(s) are not incorrectly assigned a lower SIL than is necessary. To prevent this occurring the use of a form of task-based analysis is strongly recommended to ensure that proper consideration is given to estimation of the probability of occurrence of harm.

This parameter can be estimated by taking into account:

a) Predictability of the behaviour of component parts of the machine relevant to the hazard in different modes of use (e.g. normal operation, maintenance, fault finding).

   This will necessitate careful consideration of the control system especially with regard to the risk of unexpected start up. Do not take into account the protective effect of any SRECS. This is necessary in order to estimate the amount of risk that will be exposed if the SRECS fails. In general terms, it must be considered whether the machine or material being processed has the propensity to act in an unexpected manner.

   The machine behaviour will vary from very predictable to not predictable but unexpected events cannot be discounted.

   NOTE 1 Predictability is often linked to the complexity of the machine function.

b) The specified or foreseeable characteristics of human behaviour with regard to interaction with the component parts of the machine relevant to the hazard. This can be characterised by:

   – stress (e.g. due to time constraints, work task, perceived damage limitation); and/or

   – lack of awareness of information relevant to the hazard. This will be influenced by factors such as skills, training, experience, and complexity of machine/process.

   These attributes are not usually directly under the influence of the SRECS designer, but a task analysis will reveal activities where total awareness of all issues, including unexpected outcomes, cannot be reasonably assumed.

   "Very high" probability of occurrence of a hazardous event should be selected to reflect normal production constraints and worst case considerations. Positive reasons (e.g. well-defined application and knowledge of high level of user competences) are required for any lower values to be used.

   NOTE 2 Any required or assumed skills, knowledge, etc. should be stated in the information for use.

Select the appropriate row for probability of occurrence of hazardous event (Pr) of Table A.3. Indicate the appropriate number under the Pr column in Table A.5.

Table A.3– Probability (Pr) classification

Probability of occurrence	Probability (Pr)
Very high	5
Likely	4
Possible	3
Rarely	2
Negligible	1

A.2.4.3 Probability of avoiding or limiting harm (Av)

This parameter can be estimated by taking into account aspects of the machine design and its intended application that can help to avoid or limit the harm from a hazard. These aspects include, for example

– sudden, fast or slow speed of appearance of the hazardous event;

– spatial possibility to withdraw from the hazard;

– the nature of the component or system, for example a knife is usually sharp, a pipe in a dairy environment is usually hot, electricity is usually dangerous by its nature but is not visible; and

– possibility of recognition of a hazard, for example electrical hazard: a copper bar does not change its aspect whether it is under voltage or not; to recognize if one needs an instrument to establish whether electrical equipment is energised or not; ambient conditions, for example high noise levels can prevent a person hearing a machine start.

Select the appropriate row for probability of avoidance or limiting harm (Av) of Table A.4. Insert the appropriate number under the Av column in Table A.5.

Table A.4– Probability of avoiding or limiting harm (Av) classification

Probabilities of avoiding or limiting harm (AV)	
Impossible	5
Rarely	3
Probable	1

A.2.5 Class of probability of harm (Cl)

For each hazard, and as applicable, for each severity level add up the points from the Fr, Pr and Av columns and enter the sum into the column Cl in Table A.5.

Table A.5– Parameters used to determine class of probability of harm (Cl)

Serial no.	Hazard	Se	Fr	Pr	Av	Cl
1						
2						
3						
4						

A.2.6 SIL assignment

Using Table A.6, where the severity (Se) row crosses the relevant column (Cl), the intersection point indicates whether action is required. The black area indicates the SIL assigned as the target for the SRCF. The lighter shaded areas should be used as a recommendation that other measures (OM) be used.

Table A.6 – SIL assignment matrix

Severity (Se)	Class (Cl)				
	4	5-7	8-10	11-13	14-15
4	SIL 2	SIL 2	SIL 2	SIL 3	SIL 3
3		(OM)	SIL 1	SIL 2	SIL 3
2			(OM)	SIL 1	SIL 2
1				(OM)	SIL 1

EXAMPLE: For a specific hazard with an Se assigned as 3, an Fr as 4, an Pr as 5 and an Av as 5 then:

$$Cl = Fr + Pr + Av = 4 + 5 + 5 = 14$$

Using Table A.6, this would lead to a SIL 3 being assigned to the SRCF that is intended to mitigate against the specific hazard.

Figure A.3 shows an example of documentation that may be used to record the results of a SIL assignment exercise using this informative Annex.

Risk assessment and safety measures

Document No.:

Part of:

Product: _____

Issued by: _____

Date: _____

☐ Pre risk assessment
☐ Intermediate risk assessment
☐ Follow up risk assessment

Black area = Safety measures required
Grey area = Safety measures recommended

Consequences	Severity Se	Class Cl 4	5 - 7	8 - 10	11 - 13	14 - 15	Frequency, Fr		Probability of hzd. event, Pr		Avoidance Av	
Death, loosing an eye or arm	4	SIL 2	SIL 2	SIL 2	SIL 3	SIL 3	≥ 1 per hr	5	Common	5		
Permanent, loosing fingers	3		OM	SIL 1	SIL 2	SIL 3	<1 per hr - ≥ 1 per day	5	Likely	4		
Reversible, medical attention	2			OM	SIL 1	SIL 2	<1 per day - ≥1 per 14days	4	Possible	3	Impossible	5
Reversible, first aid	1				OM	SIL 1	<1 per 2wks - ≥1 per yr	3	Rarely	2	Possible	3
							<1 per yr	2	Negligible	1	Likely	1

Ser. No.	Hzd. No.	Hazard	Se	Fr	Pr	Av	Cl	Safety measure	Safe

Comments

Figure A.3 – Example proforma for SIL assignment process

Annex B
(informative)

Example of safety-related electrical control system (SRECS) design using concepts and requirements of Clauses 5 and 6

B.1 General

The structured approach to the design of a SRECS used by this standard defines a methodology whereby functional and safety integrity requirements for safety-related control functions are decomposed into a number of sub-functions. This process is used to implement into the machinery sector a technical framework for functional safety and Figure B.1 describes the terminology used at each of these levels that is important when integrating a SRECS design at a machine installation.

This design methodology can by verification and validation processes be used to demonstrate that a SRECS fulfils the safety requirements specification described in Clause 5.

The following example of a SRECS design is intended to clarify the principles of functional decomposition and the realisation of a specified safety-related control function in accordance with the requirements of Clause 6. Consequently this example is simplified and does not consider additional measures that can be required in practice, for example hold-to-run devices.

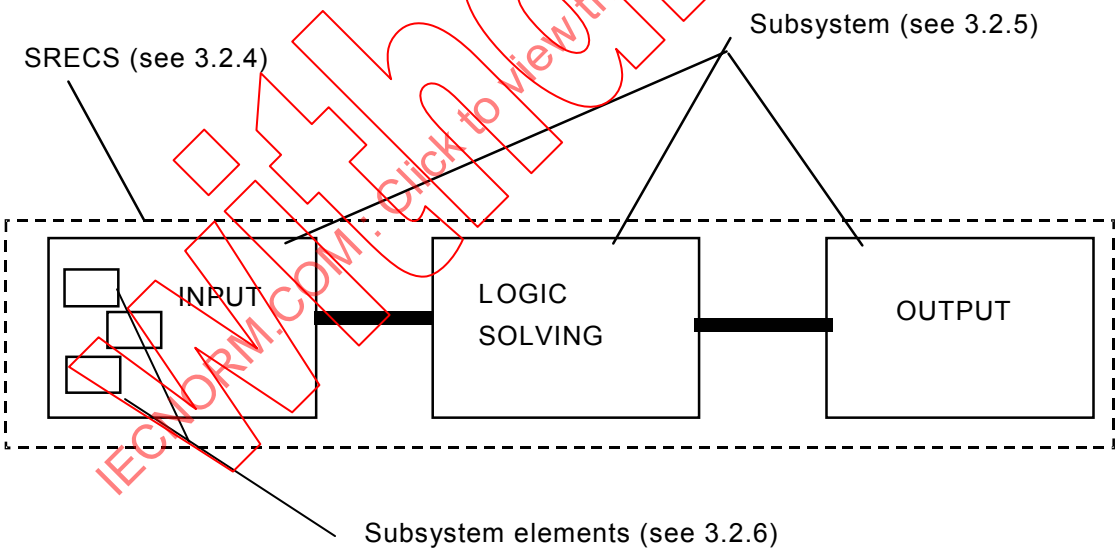


Figure B.1 – Terminology used in functional decomposition

In general, the terms presented in Figure B.1 are intended to delineate the design process into two key stages namely:

– SRECS design that may be carried out by a machinery designer or a control systems integrator; and

– subsystem (and subsystem element) design that is applicable to the vendors of electrical equipment and controlgear (e.g. contactors, interlocking switches, programmable logic controllers) and the machine designers or control system integrators.

B.2 Example

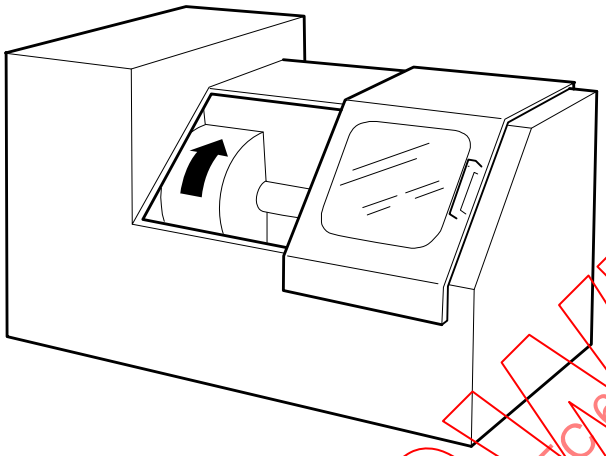


Figure B.2 – Example machine

The methodology used in this standard is based upon a structured top-down approach to the specification of safety-related control functions and the design of the SRECS that implements those functions.

Step 1: SRCF safety requirements specification (Clause 5)

From a SRCF safety requirements specification the following information can be derived:

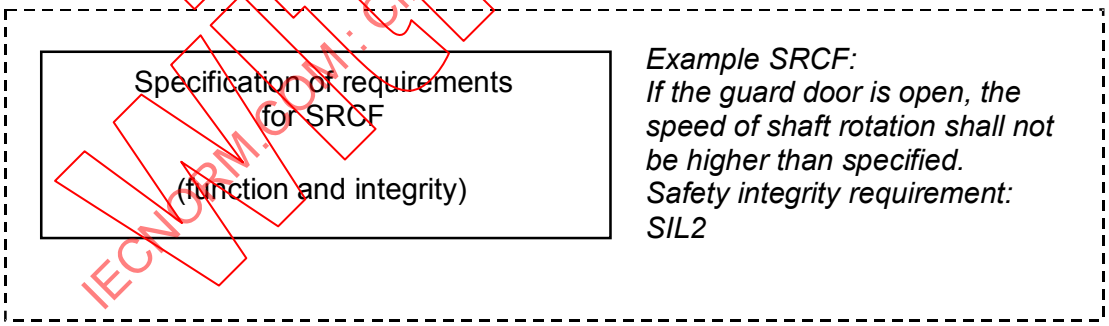


Figure B.3 – Specification of requirements for an SRCF

Step 2: SRECS design and development process (see 6.6.2)

Step 2.1: The safety-related control function as specified in the safety requirements specification is decomposed to a structure of function blocks.

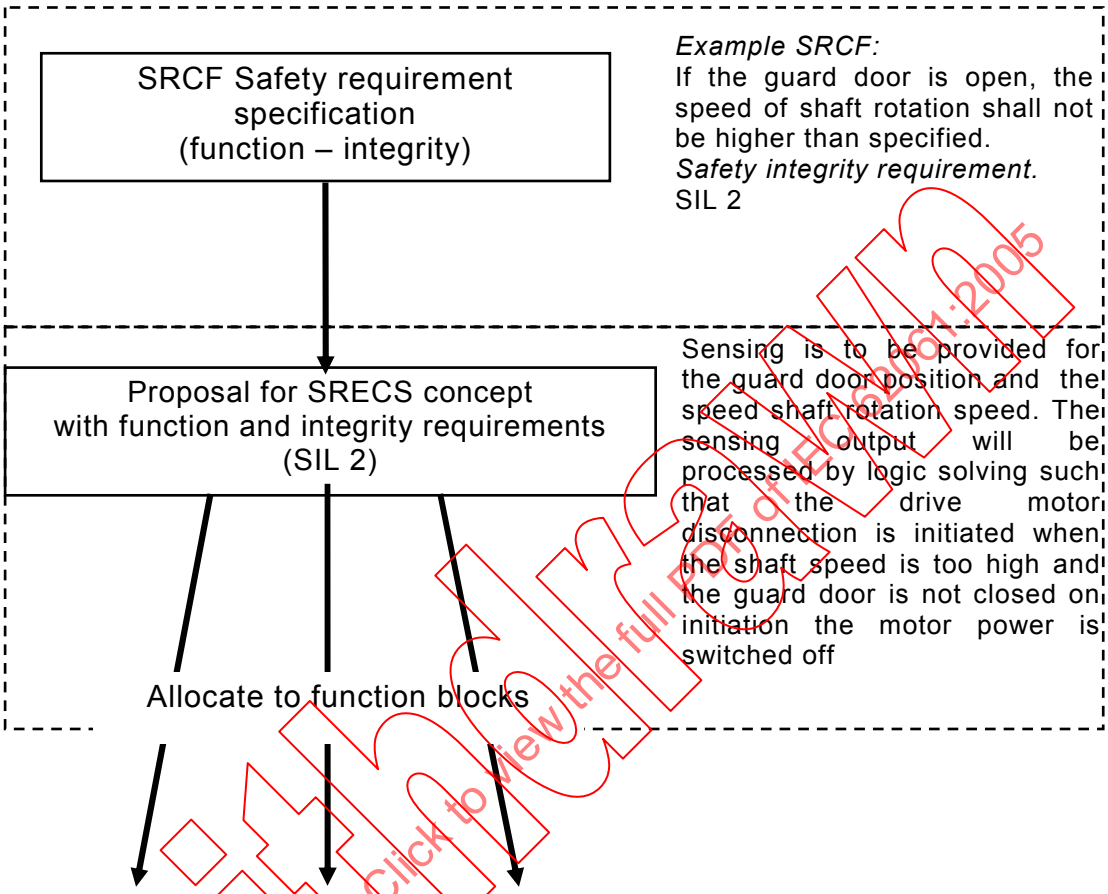


Figure B.4 – Decomposition to a structure of function blocks

Step 2.2: The structure of function blocks provides an initial concept for an architecture of the SRECS. The safety requirements for each function block are derived from the safety requirements specification of the corresponding safety-related control function.

The element(s) that implement each function block must achieve at least the same SIL capability as that assigned to the SRCF. This is shown in Figure B.5 as a SIL 2 capability (i.e. FB1 SILCL2, etc.).

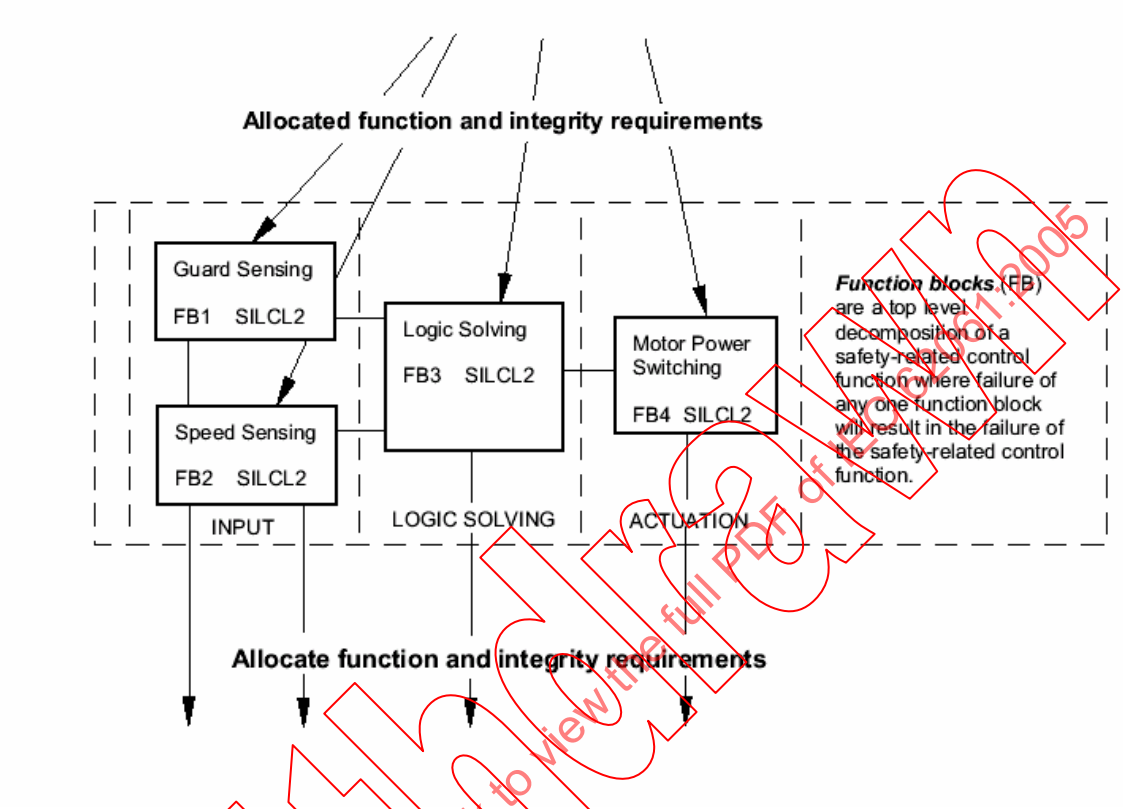


Figure B.5 – Initial concept of an architecture for a SRECS

Step 3: Each function block is allocated to a subsystem within the architecture of the SRECS. Each subsystem may consist of subsystem elements and, as necessary, diagnostic functions to ensure that faults can be detected and appropriate action taken (see 6.2).

The architecture should describe the SRECS in terms of its subsystems and their interrelationship. For this example there are a number of alternatives that can be used for realisation of the SRECS and its subsystems architecture.

Example 1: In this example (see Figure B.6), the diagnostic functions are embedded within each subsystem.

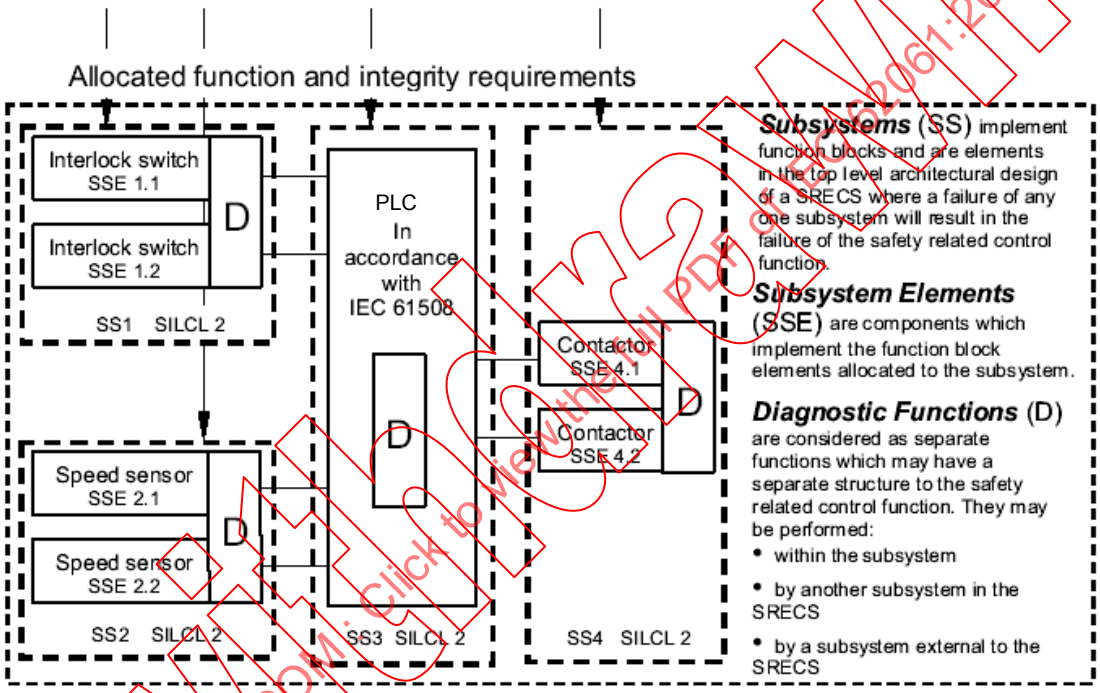


Figure B.6 – SRECS architecture with diagnostic functions embedded within each subsystem (SS1 to SS4)

Example 2: In this example (see Figure B.7), the diagnostic functions are embedded within a programmable logic controller (PLC) in SS3 that satisfies relevant aspects of IEC 61508.

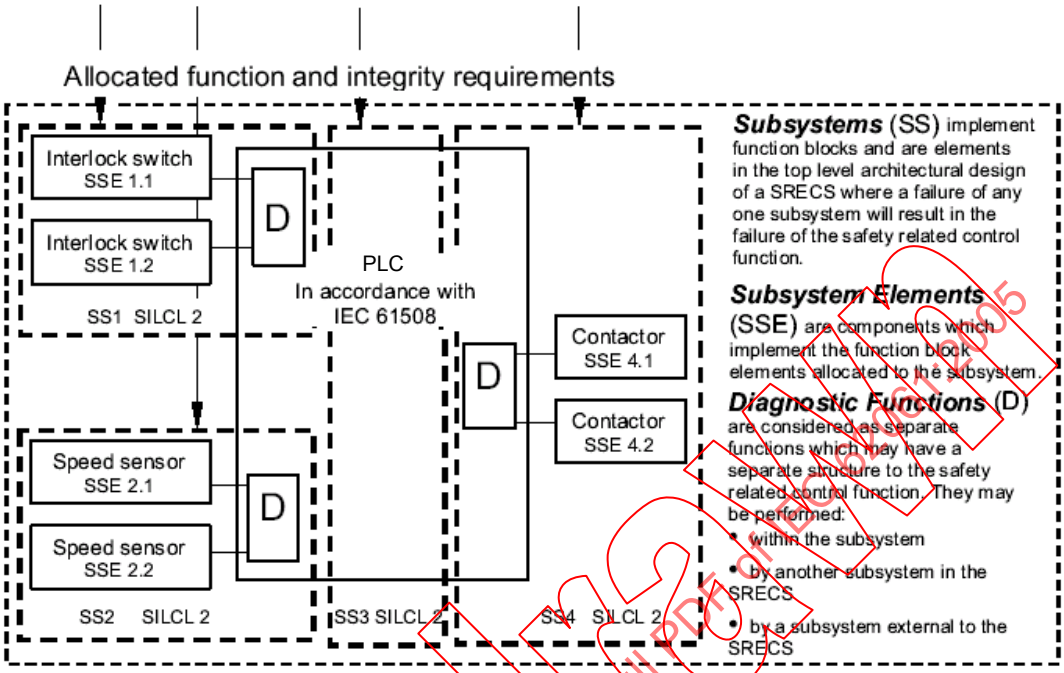


Figure B.7 – SRECS architecture with diagnostic functions embedded within subsystem SS3

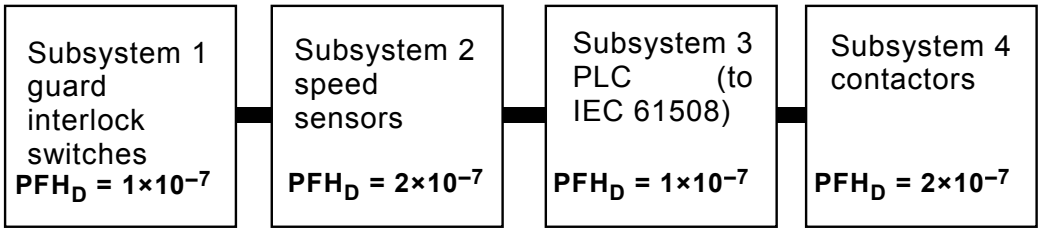
Step 4: Estimation of the SIL achieved by the SRECS (see 6.6.3)

The SIL that can be claimed for the SRECS shall be less than or equal to the lowest value of the SILCLs of any of the subsystems.The probability of dangerous random hardware failure of the SRECS (PFH_{DSRECS}) is the sum of the probabilities of dangerous failure per hour of all subsystems (PFH_{D1} to PFH_{Dn}) involved in the performance of the safety-related control function and shall include, where appropriate, the probability of dangerous transmission errors (P_{TE}) for digital data communication processes as:

$$PFH_{DSRECS} = PFH_{D1} + \ldots + PFH_{Dn} + P_{TE}$$

For this example, the target failure value for the safety-related control function is SIL 2 and from Table 3 (see 5.2.4.2) this is equivalent to a probability of dangerous failure per hour (PFH_D) in the range ≥ 10^{-7} to < 10^{-6}. Therefore, assuming that the probabilities of dangerous failure per hour of each of the subsystems are as shown below, the sum of the probabilities of dangerous failure per hour of all subsystems can be estimated as shown in Figure B.8.

Therefore, in this example, the design of SRECS can be shown to satisfy all the requirements to implement the assigned safety-related control function at SIL 2.



PFH_{DSRECS} = (1 x 10^{-7}) + (2 x 10^{-7}) + (1 x 10^{-7}) + (2 x 10^{-7}) = 6 x 10^{-7}

Figure B.8 – Estimation of PFH_D for a SRECS

Annex C
(informative)

Guide to embedded software design and development

NOTE This informative Annex is provided to indicate the basic approach required in order to satisfy the requirements of IEC 61508-3. It cannot in itself provide conformance with IEC 61508-3 without applying further measures.

C.1 General

This Annex is provided to assist persons in the design and development of embedded software for implementing safety-related control functions within a SRECS.

The major objective dealt with here is general guidance on the prevention of embedded software failures and any other unexpected behaviour of embedded software that might lead to the creation of dangerous faults in the system.

In order to satisfy these objectives, consideration is given to the following points:

– a description of the main characteristics that software elements of a SRECS should possess to guarantee its quality and safety (software element guidelines);

– the establishment of all relevant technical activities and provisions associated with software development, for those involved in software design. These can then be used to guide the designer during the production of this type of software (software development process guidelines);

– a reference framework for software evaluation. This allows the software designer and/or analyst to decide that software elements satisfy the safety requirements of the SRECS or SRECS subsystem to be analysed (software verification guidelines).

This Annex provides a set of basic guidelines, coherent with the IEC 61508-3, that are adapted to embedded software for microprocessors.

C.2 Software element guidelines

This Clause presents the guidelines that an embedded software element of a SRECS or SRECS subsystem should fulfil to be safe in operation and of satisfactorily high quality. To obtain such a software element, a number of activities, a certain organisation and a number of principles should all be established. This should take place as early as possible in the development cycle.

C.2.1 Interface with system architecture

The list of constraints imposed by hardware architecture on software should be defined and documented. Consequences of any hardware/software interaction on the safety of the machine or system being monitored should be identified and evaluated by the designer, and taken into account in the software design.

NOTE Constraints include: protocols and formats, input/output frequencies, by rising and falling edge or by level, input data using reverse logic, etc. Listing these constraints allows them to be taken into account at the start of the development activity, and reduces the risk of incompatibilities between software and hardware when the former is installed in the target hardware.

C.2.2 Software specifications

Software specifications should take the following points into account:

– safety-related control functions with a quantitative description of the performance criteria (precision, exactness) and temporal constraints (response time), all with tolerances or margins when possible;

– system configuration or architecture;

– instructions relevant to hardware safety integrity (logic solvers, sensors, actuators, etc.);

– instructions relevant to software integrity;

– constraints related to memory capacity and system response time;

– operator and equipment interfaces;

– instructions for software self-monitoring and for hardware monitoring carried out by the software;

– instructions that allow all the safety-related control functions to be verified while the systems are working (e.g. on-line testing, capture time for fleeting signals, coincidence with scan rate).

NOTE 1 The instructions for monitoring, developed taking safety objectives and operating constraints (duration of continuous operation, etc.) into account, can include devices such as watch dogs, central processing unit (CPU) load monitoring, feedback of output to input for software self-monitoring. For hardware monitoring, CPU and memory monitoring, etc. instructions for safety-related control function verification: for example, the possibility of periodically verifying the correct operation of safety devices should be included in the specifications.

Functional requirements should be specified for each functional mode. The transition from one mode to the other should be specified.

NOTE 2 Functional modes can include nominal modes, and one or more degraded modes. The objective is to specify the behaviour in all situations in order to avoid unexpected behaviours in non-nominal contexts.

C.2.3 Pre-existent software

The term "pre-existent" software refers to source modules that have not been developed specifically for the system at hand, and are integrated into the rest of the software. These include software elements developed by the designer for previous projects, or commercially available software (e.g. modules for calculations, algorithms for data sorting).

When dealing with this type of software, and especially in the case of commercial software elements, the designer does not always have access to all the elements needed to satisfy the previous requirements (e.g. what tests have been carried out, is the design documentation available). Specific co-ordination with the analyst can therefore be necessary at the earliest possible moment.

The designer should indicate the use of pre-existent software to the analyst, and the designer should demonstrate that pre-existent software has the same level as the other software elements. Such a demonstration should be done:

a) either by using the same verification activities on the pre-existent software as on the rest of the software; and/or

b) through practical experience where the pre-existent software has functioned on a similar system in a comparable executable environment (e.g. it may be necessary to evaluate the consequences of a change of the compiler or of a different software architecture format).

NOTE 1 The goal of indicating the use of pre-existent software is to open up consultation with the analyst as early as possible about any eventual difficulties that this type of software might cause. The integration of pre-existent source modules can be the cause of certain anomalies or unsafe behaviour if they were not developed with the same rigour as the rest of the software.

Pre-existent software should be identified using the same configuration management and version control principles that are applied to the rest of the software.

NOTE 2 Configuration management and version control should be exercised over all the software components, regardless of their origin.

C.2.4 Software design

Description of the software design should include a description of:

– the software architecture that defines the structure decided to satisfy specifications;
– inputs and outputs (e.g. in the form of an internal and external data dictionary), for all the modules making up the software architecture;
– the interrupts;
– the global data;
– each software module (inputs/outputs, algorithm, design particularities, etc.);
– module or data libraries used;
– pre-existent software used.

Software should be modular and written in a logical manner in order to facilitate its verification or maintenance:

– each module or group of modules should correspond, if possible, to a function in the specification(s);
– interfaces between modules should be as simple as possible.

NOTE The general characteristic of correct software architecture can be summed up in the following way: a module should possess a high level of functional cohesion and a simple interface with its environment.

Software should:

– limit the number or extent of global variables;
– control the layout of arrays in memory (to avoid a risk of array overflows).

C.2.5 Coding

The source code should:

– be readable, understandable, and subject to tests;
– satisfy design specifications of the software module;
– obey the coding manual instructions.

C.3 Software development process guidelines

C.3.1 Development process: software lifecycle

The objective of the following guidance applicable to the software lifecycle is to obtain a formalized description of the organization of software development and, in particular, the different technical tasks making up this development.

The software development lifecycle should be specified and documented (e.g. in a software quality plan). The lifecycle should include all the technical activities and phases necessary and sufficient for software development.

Each phase of the lifecycle should be divided into its elementary tasks and should include a description of:

– inputs (documents, standards, etc.);
– outputs (documents produced, analytical reports, etc.);
– activities to be carried out;
– verifications to be performed (analyses, tests, etc.).

C.3.2 Documentation : documentation management

The documentation should conform to Clause 10 of this standard.

C.3.3 Configuration and software modification management

Management of the configuration and therefore of the version is an indispensable part of any development which may require approval. Indeed, approval is only valid where a given configuration can be identified. Configuration management includes configuration identification activities, modification management, the establishment of reference points and the archiving of software elements, including the associated data (documents, records of tests, etc.). Throughout the entire project lifecycle, the principal objectives are to provide:

– a defined and controlled software configuration that guarantee physical archiving and that can be used to reproduce an executable code coherently (with future software production or modification in mind);
– a reference basis for modifications management;
– a means of control so that any problems are properly analysed, and that the approved modifications are properly carried out.

Concerning the modifications, their reasons could arise from, for example:

– functional safety below that specified;
– systematic fault experience;
– new or amended safety legislation;
– modifications to the machine or its use;
– modification to the overall safety requirements;
– analysis of operations and maintenance performance, indicating that the performance is below target.

C.3.4 Configuration and archiving management

A procedure for configuration management and modifications management should be defined and documented. This procedure should, as a minimum, include the following items:

– articles managed by the configuration, at least: software specification, preliminary and detailed software design, source code modules, plans, procedures and results of the validation tests;
– identification rules (of a source module, of a software version, etc.);
– treatment of modifications (recording of requests, etc.).

For each article of configuration, it should be possible to identify any changes that may have occurred and the versions of any associated elements.

NOTE 1 The purpose is to be able to trace the historical development of each article: what modifications have been made, why, and when.

Software configuration management should allow a precise and unique software version identification to be obtained. Configuration management should associate all the articles (and their version) needed to demonstrate the functional safety.

All articles in the software configuration should be covered by the configuration management procedure before being tested or being requested by the analyst for final software version evaluation.

NOTE 2 The objective here is to ensure that the evaluation procedure be performed on software with all elements in a precise state. Any subsequent change may necessitate revision of the software so that it can be identifiable by the analyst.

Procedures for the archiving of software and its associated data should be established (methods for storing backups and archives).

NOTE 3 These backups and archives can be used to maintain and modify software during its functional lifetime.

C.3.5 Software modifications management

Any software modification which has an impact on the functional safety of the SRECS should be subject to the rules established for modification and configuration management such that the development process be recommenced at the highest "upstream" point needed to take the modification into account without diminishing the functional safety.

NOTE In particular, the documentation should also be updated, and all necessary verification activities carried out. This guarantees that the software will keep all its initial properties after any modification.

C.4 Development tools

Tools used during the development procedure (compiler, linker, tests, etc.) should be identified (name, reference, version, etc.) in the documentation associated with the software version (e.g. in the version control documentation).

NOTE Different versions of tools do not necessarily produce the same results. Precise identification of tools thus directly demonstrates the continuity of the process of generation of an executable version in the event that a version is modified.

C.5 Reproduction, delivery

C.5.1 Executable code production

Any option or change in the generation, during the software production should be recorded (e.g. in the version sheet) so that it is possible to say how and when the software was generated.

C.5.2 Software installation and exploitation

All failures linked to safety-related control functions brought to the attention of the designer of the system should be recorded and analysed.

NOTE This means that the designer is aware of any safety-related failures that are communicated to him and that he takes the appropriate action (e.g. warning other users, software modification, etc.).

C.6 Software verification and validation

The purpose of verification activities is to demonstrate that software elements stemming from a given phase of the development cycle conform to the specifications established during the previous phases and to any applicable standards or rules. They also serve as a means of detecting and accounting for any errors that might have been introduced during software development.

Software verification is not simply a series of tests, even though this is the predominant activity for the relatively small software element considered in this Annex**.** Other activities such as reviews and analyses, whether associated with these tests or not, are also considered to be verification activities. In certain cases, they can replace some tests (e.g. in the event that a test cannot be carried out because it would cause deterioration of a hardware component).

C.7 General verification and validation guidelines

The analyst should be able to carry out the evaluation of software conformity by conducting any audits or expertises deemed useful during the different software development phases.

All technical aspects of software lifecycle processes are subject to evaluation by the analyst. The analyst should be allowed to consult all verification reports (tests, analyses, etc.) and all technical documents used during software development.

NOTE 1 The intervention of the analyst at the specification phase is preferable to an *a posteriori* intervention since it should limit the impact of any decisions made. On the other hand, financial and human aspects of the project are not subject to evaluation.

NOTE 2 It is in the interest of the applicant to provide satisfactory evidence of all activities carried out during software development.

NOTE 3 The analyst should have all the necessary elements at his or her disposal in order to formulate an opinion.

Evaluation of software conformity is performed for a specific, referenced software version. Any modification of previously evaluated software that has received a final opinion from the analyst should be pointed out to the latter so that any additional evaluation activities can be carried out to update this opinion.

NOTE 4 Any modification can modify software behaviour; the evaluation performed by the analyst can therefore only be applied to a precise software version.

C.8 Verification and validation review

Analysis activities and software design verification should verify the conformity to specifications.

NOTE 1 The purpose is to ensure that the software specification and design (both detailed and preliminary) are coherent.

An external validation review (with the analyst) should be held at the end of the validation phase.

NOTE 2 This can be used to ascertain whether or not the element satisfies the specifications.

The result of each review should be documented and archived. It should include a list of all actions decided on in the review process, and the review conclusion (decision on whether or not to move on to the next activity). The activities defined in the review should be monitored and treated.

C.9 Software testing

C.9.1 General validation

Before writing the first test sheets, it is important to establish a test strategy in a test plan. This strategy indicates the approach adopted, the objectives that have been set in terms of test coverage, the environments and specific techniques used, the success criteria to be applied, etc.

The test objectives should be adapted to the type of software, and to the specific factors. These criteria determine the types of test to be undertaken – functional tests, limit tests, out of limit tests, performance tests, load tests, external equipment failure tests, configuration tests – as well as the range of objects to be covered by the tests (functional mode tests, safety-related control function tests, tests of each element in the specification, etc.).

Verification of a new software version should include non-regression tests.

NOTE Non-regression tests are used to ensure that the modifications performed on the software have not modified the behaviour of the software in any unexpected way.

C.9.2 Software specification verification: validation tests

The purpose of these verifications is to detect errors associated with the software in the target system environment. Errors detected by this type of verification include: any incorrect mechanism to treat interruptions, insufficient respect of running time requirements, incorrect response from the software operating in transient mode (start-up, input flow, switching in a degraded mode, etc.), conflicts of access to different resources or organizational problems in the memory, inability of integrated tests to detect faults, software/hardware interface errors, stack overflows. Validation tests are the principal component of software specification verification.

The test coverage should be made explicit in a traceability matrix and ensure that:

– each element of the specification, including safety mechanisms, is covered by a validation test; and

– the real-time behaviour of the software in any operational mode can be verified.

Furthermore, the validation should be carried out in conditions representative of the operational conditions of the SRECS or the SRECS subsystem.

NOTE 1 This guarantees that the software reacts as expected in operation. It applies only to cases where the test conditions can be destructive for hardware (e.g. physical fault of a component that cannot be simulated). To be significant, validation should be performed in the operational conditions of the SRECS or SRECS subsystem (i.e. with the final versions of software and hardware, and the software installed in the target system). Any other combination could decrease the efficiency of the test and require analysis of its representation.

Validation results should be recorded in a validation report that should cover at least the following points:

– the versions of software and system that were validated;

– a description of the validation tests performed (inputs, outputs, testing procedures);

– the tools and equipment used to validate or evaluate the results;

– the results showing whether each validation test was a success or failure;
– a validation assessment: identified non-conformities, impact on safety, decision as to whether or not to accept the validation.

A validation report should be made available for each delivered software version and should correspond to the final version of each delivered software element.

NOTE 2 This report can be used to provide proof that tests were indeed carried out, and that the results were correct (or contained explainable deviations). It can also be used to redo tests at a later date, for a future software version or for another project. It provides a guarantee that each delivered version has been validated in its final form. On the other hand, it does not impose a complete validation of each modification of an existing code – an impact analysis can, in certain cases, justify partial validation.

C.9.3 Software design verification: software integration tests

This verification focuses on the correct assembly of software modules and on the mutual relationships between software components. It can be used to reveal errors of the following kind: incorrect initialization of variables and constants, errors in the transfer of parameters, any data alteration, especially global data, incorrect sequencing of events and operations.

Software integration tests should be able to verify:

– correct sequencing of the software execution;
– exchange of data between modules;
– respect of the performance criteria;
– non-alteration of global data.

The test coverage should be given explicitly in a traceability matrix demonstrating the correspondence between the tests to be undertaken and the objectives of the tests defined.

Integration test results should be recorded in a software integration test report, which should, as a minimum, contain the following points:

– the version of the integrated software;
– a description of the tests performed (inputs, outputs, procedures);
– the integration tests results and their evaluation.

C.9.4 Detailed design verification: module tests

Module tests focus on software modules and their conformity with the detailed design. This activity can be indispensable for large and complex software elements, but is only recommended for the relatively small software elements dealt with here. This phase of the verification procedure allows detection of the following types of errors:

– inability of an algorithm to satisfy software specifications;
– incorrect loop operations;
– incorrect logical decisions;
– inability to compute valid combinations of input data correctly;
– incorrect responses to missed or altered input data;
– violation of array boundaries;
– incorrect calculation sequences;

– inadequate precision;
– accuracy or performance of an algorithm.

Each software module should be submitted to a series of tests to verify, using input data, that the module fulfils the functions specified at the detailed design stage.

The test coverage should be provided in a traceability matrix that demonstrates the correspondence between the test results and the objectives of the tests defined.

Annex D
(informative)

Failure modes of electrical/electronic components

The minimum operational environment for a SRECS and its subsystems should be that specified in IEC 60204-1. However, in practice many subsystems (e.g. AOPDs) conform to a product standard that may have specified a more onerous operational environment.

The information given in Table D.1 are examples of the failure mode ratios for electrical/electronic components derived from the reference sources indicated. These values can differ from the information provided in other sources. Generally the failure mode data used should reflect practical application of the components.

NOTE 1 The following table is not an exhaustive list of component failure modes.

NOTE 2 Failure rate data should be provided by the subsystem manufacturers.

Table D.1 – Examples of the failure mode ratios for electrical/electronic components

Component	Failure mode	Typical failure mode ratios %
Switch with positive opening on demand, for example push button, emergency stop device, position switches, cam operated, selector switches	Contacts will not open	20
	Contacts will not close	80
Electromechanical position switch, limit switch, manually operated switch, etc. (not positively opening on demand)	Contacts will not open	50
	Contacts will not close	50
Relay	All contacts remain in the energized position when the coil is de-energized	25
	All contacts remain in the de-energized position when the coil is energized	25
	Contacts will not open	10
	Contacts will not close	10
	Simultaneous short circuit between three contacts of a change-over contact	10
	Simultaneous closing of normally open and normally closed contacts	10
	Short circuit between two pairs of contacts and/or between contacts and coil terminal	10

Component	Failure mode	Typical failure mode ratios %
Circuit breaker, differential circuit breaker, residual current device	All contacts remain in the energized position when the coil is de-energized	25
	All contacts remain in the de-energized position when the coil is energized	25
	Contacts will not open	10
	Contacts will not close	10
	Simultaneous short circuit between three contacts of a change-over contact	10
	Simultaneous closing of normally open and normally closed contacts	10
	Short circuit between two pairs of contacts and/or between contacts and coil terminal	10
Contactor	All contacts remain in the energized position when the coil is de-energized	25
	All contacts remain in the de-energized position when the coil is energized	25
	Contacts will not open	10
	Contacts will not close	10
	Simultaneous short circuit between three contacts of a change-over contact	10
	Simultaneous closing of normally open and normally closed contacts	10
	Short circuit between two pairs of contacts and/or between contacts and coil terminal	10
Fuse	Fails to blow (short circuit)	10
	Open circuit	90
Proximity switch	Permanently low resistance at output	25
	Permanently high resistance at output	25
	Interruption in power supply	30
	No operation of switch due to mechanical failure	10
	Simultaneous short circuit between three terminals of changeover contacts	10
Temperature switch	Contacts will not close	30
	Contacts will not open	10
	Short circuits between adjacent contacts	10
	Simultaneous short circuit between three terminals of change-over contacts	10
	Faulty sensor	20
	Change of the detection or output characteristic	20

Component	Failure mode	Typical failure mode ratios %
Pressure switch	Contacts will not close	30
	Contacts will not open	10
	Short circuits between adjacent contacts	10
	Simultaneous short circuit between three terminals of change-over contacts	10
	Faulty sensor	20
	Change of the detection or output characteristic	20
Solenoid valve	Does not energize	5
	Does not de-energize	15
	Change of switching times	5
	Leakage	65
	Other failure modes (see Note 4)	10
Transformer	Open circuit of individual winding	70
	Short circuit between different windings	10
	Short circuit in one winding	10
	Change in effective turns ratio	10
Inductances	Open circuit	80
	Short circuit	10
	Random change of value	10
Resistors	Open circuit	80
	Short circuit	10
	Random change of value	10
Resistor networks	Open circuit	70
	Short circuit	10
	Short circuit between any connections	10
	Random change of value	10
Potentiometers	Open circuit of individual connection	70
	Short circuit between all connections	10
	Short circuit between any two connections	10
	Random change of value	10
Capacitors	Open circuit	40
	Short circuit	40
	Random change of value	10
	Changing value tan α	10

Component	Failure mode	Typical failure mode ratios %
Discrete semiconductors	Open circuit of any connection	25
	Short circuit between any two connections	25
	Short circuit between all connections	25
	Change in characteristics	25
Non-programmable integrated circuits (non-complex, i.e. less than 1 000 gates and/or less than 24 pins, operational amplifiers, shift registers, and hybrid modules)	Open circuit of any connection	20
	Short circuit between any two connections	20
	"Stuck at" faults	20
	Parasitic oscillation of outputs	20
	Changing values (e.g. input/output voltage of analogue device)	20
Opto-couplers	Open circuit of individual connection	30
	Short circuit between any two input connections	30
	Short circuit between any two output connections	30
	Short circuit between any two connections of input and output	10
Plug and socket, multi-pin connector	Short circuit between any two adjacent pins	10
	Short circuit of any conductor to an exposed conductive part	10
	Open circuit of individual connector pins	80
Terminal block	Short circuit between adjacent terminals	10
	Open circuit of individual terminals	90

NOTE 1 This data has been derived from a number of industry sources including:

MIL-HDBK 217F(Notice 2) Reliability Prediction of Electronic Equipment (28-02-95), Parts Stress Analysis

MIL-HDBK 217F(Notice 2) Reliability Prediction of Electronic Equipment (28-02-95), Appendix A, Parts Count Reliability Prediction

SN 29500 Part 7, Failure Rates of Components, Expected Values for Relays, April 1992

SN 29500 Part 11, Failure Rates of Components, Expected Values for Contactors, August 1990

The documents in the SN 29500 series are publicly available and can be obtained from:

Siemens AG, CT SR SI

Otto-Hahn-Ring 6

D-81739 München

NOTE 2 Electrical failure modes taken from Tables D.5 of ISO 13849-2. Mechanical failure modes (where applicable) are taken from Annexes A, B and C of ISO 13849-2.

NOTE 3 For a number of electrical/electronic components, for example resistors and capacitors, different designs may have a different distribution of failure modes from those given in the table.

NOTE 4 Other failure modes that apply to a solenoid valve include:

- non-switching (sticking in the end or zero position) or incomplete switching (sticking in a random intermediate position);
- spontaneous change of the initial switching position (without an input signal);
- change in the leakage flow rate over a long period of time;
- bursting of the valve housing or breakage of the moving component(s) as well as breakage/fracture of the mounting or housing screws;
- pneumatic/hydraulic faults which cause uncontrolled behaviour for servo and proportional valves.

NOTE 5 Other sources of information on failure rate data and failure mode ratios include:

- UTE C 80-810 RDF 2000: Reliability data handbook – A universal model for reliability prediction of electronic components, PCBs and equipments

Failure mode/mechanism distributions FMD-91, RAC 1991

Annex E
(informative)

Electromagnetic (EM) phenomenon and increased immunity levels for SRECS intended for use in an industrial environment according to IEC 61000-6-2

Table E.1 – EM phenomenon and increased immunity levels for SRECS

Port (see Note 1)	**Phenomenon**	**Basic standard**	**Increased values for additional tests for SRECS performance (see 6.4.3)**
Enclosure	Electrostatic discharge (ESD)	IEC 61000-4-2	6 kV/8 kV contact/air discharge (see Note 2)
	Electromagnetic (EM) field	IEC 61000-4-3	20V/m (80 MHz – 1 GHz) 6 V/m (1,4 GHz – 2 GHz) 3V/m (2 GHz – 2,7 GHz) (see Table E.2 and Note 3)
	Rated power frequency magnetic field	IEC 61000-4-8	30 A/m (see Notes 4 and 5)
AC power	Voltage dip/short interruptions	IEC 61000-4-11	0,5 period 30 % reduction (see Note 5)
	Voltage variations/ interruptions	IEC 61000-4-11	250 periods >95 % reduction (see Note 5)
	Burst	IEC 61000-4-4	4 kV
	Surge	IEC 61000-4-5	2 kV line-to-line/4 kV line-to-ground (see Note 6)
	Conducted radio frequency (RF)	IEC 61000-4-6	10 V at frequencies given (see Table E.3 and Note 3)
DC power (see Note 7)	Burst	IEC 61000-4-4	4 kV
	Surge	IEC 61000-4-5	1 kV line-to-line/2 kV line-to-ground (see Note 6)
	Conducted RF	IEC 61000-4-6	10 V at frequencies given (see Table E.3 and Note 3)
I/O signal/ control lines	Burst	IEC 61000-4-4	2 kV for lines > 3 m
	Surge	IEC 61000-4-5	2 kV line-to-ground (see Note 8)
	Conducted RF	IEC 61000-4-6	10 V at frequencies given (see Table E.3 and Note 3)
Functional earth	Burst	IEC 61000-4-4	2 kV

NOTE 1 A port is a particular interface of the SRECS and its subsystems with the external electromagnetic environment.

NOTE 2 Levels shall be applied in accordance with the environmental conditions described in IEC 61000-4-2 on parts which can be touched by persons other than staff working in accordance with defined procedures for the control of ESD but not to equipment where access is limited to appropriately trained personnel only.

NOTE 3 The increased values shall be applied in frequency ranges generally used for mobile digital radio transmitters, except where reliable measures are used to prevent electromagnetic interference from such equipment. ISM frequencies have to be taken into account on an individual basis.

NOTE 4 Only to magnetically sensitive equipment.

NOTE 5 An increased value is not applied to phenomena where this is not considered necessary for functional safety.

NOTE 6 External protection devices are allowed to achieve immunity.

NOTE 7 DC connections between parts of equipment/system that are not connected to a D.C. distribution network are treated as I/O signal/control ports.

NOTE 8 Only in the case of long-distance lines.

NOTE 9 Reference IEC 61326-3 (in preparation).

NOTE 10 Where product standards (e.g. IEC 61496-1) specify different test levels for specific EMC phenomena in the context of functional safety those different test levels are applicable to those SRECS subsystems.

Table E.2 – Selected frequencies for RF field tests

System	Frequency
GSM	890 – 915 MHz
GSM	1 710 – 1 785 MHz
GSM	1 890 MHz
UMTS	To be developed
Walkie Talkie	To be developed
ISM	433,05 – 434,79 MHz
ISM ???	83,996 – 84,004 MHz
ISM ???	167,992 – 168,008 MHz
ISM???	886,000 – 906,000 MHz

Table E.3 – Selected frequencies for conducted RF tests

System	Frequency
ISM	6,765 – 6,795 MHz
ISM	13,553 – 13,567 MHz
ISM	26,957 – 27,283 MHz
ISM	40,66 – 40,70 MHz

Annex F
(informative)

Methodology for the estimation of susceptibility to common cause failures (CCF)

F.1 General

This informative Annex provides a simple qualitative approach for the estimation of CCF that can be applied to the subsystem design.

F.2 Methodology

The proposed design of a subsystem should be assessed to establish the effectiveness of the measures used to safeguard against CCF. The items in Table F.1 that are applicable should be identified and an overall score established, which is used to determine the common cause failure factor from Table F.2 as a percentage value.

Table F.1 – Criteria for estimation of CCF

Item	Reference	Score
Separation/segregation		
Are SRECS signal cables for the individual channels routed separately from other channels at all positions or sufficiently shielded?	1a	5
Where information encoding/decoding is used, is it sufficient for the detection of signal transmission errors?	1b	10
Are SRECS signal and electrical energy power cables separate at all positions or sufficiently shielded?	2	5
If subsystem elements can contribute to a CCF, are they provided as physically separate devices in their local enclosures?	3	5
Diversity/redundancy		
Does the subsystem employ different electrical technologies for example, one electronic or programmable electronic and the other an electromechanical relay?	4	8
Does the subsystem employ elements that use different physical principles (e.g. sensing elements at a guard door that use mechanical and magnetic sensing techniques)?	5	10
Does the subsystem employ elements with temporal differences in functional operation and/or failure modes?	6	10
Do the subsystem elements have a diagnostic test interval of ≤1 min?	7	10
Complexity/design/application		
Is cross-connection between channels of the subsystem prevented with the exception of that used for diagnostic testing purposes?	8	2
Assessment/analysis		
Have the results of the failure modes and effects analysis been examined to establish sources of common cause failure and have predetermined sources of common cause failure been eliminated by design?	9	9
Are field failures analysed with feedback into the design?	10	9
Competence/training		
Do subsystem designers understand the causes and consequences of common cause failures?	11	4

Item	Reference	Score
Environmental control		
Are the subsystem elements likely to operate always within the range of temperature, humidity, corrosion, dust, vibration, etc. over which it has been tested, without the use of external environmental control?	12	9
Is the subsystem immune to adverse influences from electromagnetic interference up to and including the limits specified in Annex E?	13	9
NOTE An alternative item (e.g. references 1a and 1b) is given in Table F.1 where it is intended that a claim can be made for a contribution towards avoidance of CCF from only the most relevant item.		

Using Table F.1 those items that are considered to affect the subsystem design should be added to provide an overall score for the design that is to be implemented. Where it can be shown that equivalent means of avoiding of CCF can be achieved through the use of specific design measures (e.g. the use of opto-isolated devices rather than shielded cables) then the relevant score can be claimed as this can be considered to provide the same contribution to the avoidance of CCF.

This overall score can be used to determine a common cause failure factor (β) using Table F.2.

Table F.2 – Estimation of CCF factor (β)

Overall score	Common cause failure factor (β)
≤ 35	10 % (0,1)
36 – 65	5 % (0,05)
66 – 85	2 % (0,02)
86 – 100	1 % (0,01)

The value of β derived should be used in the estimation of the probability of dangerous failure as required in 6.7.8.1.

SOMMAIRE

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

_____

SÉCURITÉ DES MACHINES – SÉCURITÉ FONCTIONNELLE DES SYSTÈMES DE COMMANDE ÉLECTRIQUES, ÉLECTRONIQUES ET ÉLECTRONIQUES PROGRAMMABLES RELATIFS À LA SÉCURITÉ

AVANT-PROPOS

1) La Commission Électrotechnique 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 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 tels 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'unification internationale, les Comités nationaux de la CEI s'engagent à appliquer de façon transparente, dans toute la mesure possible, les Normes internationales de la CEI dans leurs normes nationales et régionales. Toute divergence entre la norme de la CEI et la norme nationale ou régionale correspondante doit être indiquée en termes clairs dans cette dernière.
5) La CEI n'a fixé aucune procédure concernant le marquage comme indication d'approbation et sa responsabilité n'est pas engagée quand un matériel est déclaré conforme à l'une de ses normes.
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.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. 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.

La Norme internationale CEI 62061 a été établie par le comité d'études 44 de la CEI: Sécurité des machines – Aspects électrotechniques.

Le texte de la présente norme est issu des documents suivants:

FDIS	Rapport de vote
44/460/FDIS	44/470/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de maintenance 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.

Le contenu des corrigenda de juillet 2005 et avril 2008 a été pris en considération dans cet exemplaire.

INTRODUCTION

Suite à l'automatisation, ainsi qu'à la demande d'une production plus élevée avec une réduction des efforts physiques des opérateurs, les systèmes de commande électriques relatifs à la sécurité (appelés SRECS ci-après) des machines jouent un rôle croissant dans la réalisation de la sécurité d'ensemble des machines. De ce fait, les SRECS eux-mêmes utilisent de plus en plus souvent une technologie électronique complexe.

Auparavant, en l'absence de normes, on a pu observer un manque d'enthousiasme à utiliser les SRECS dans les fonctions relatives à la sécurité pour des phénomènes dangereux significatifs sur les machines, en raison de l'incertitude concernant le fonctionnement d'une telle technologie.

La présente Norme internationale est destinée à être utilisée par les concepteurs de machines, les fabricants et les intégrateurs de systèmes de commande, et autres, impliqués dans la spécification, la conception et la validation d'un SRECS. Elle présente une approche et donne les exigences nécessaires à la réalisation du fonctionnement requis.

La présente norme est spécifique au secteur des machines dans le cadre de la CEI 61508. Elle est destinée à faciliter la spécification du fonctionnement des systèmes de commande électriques relatifs à la sécurité par rapport aux dangers significatifs (voir 3.8 de l'ISO 12100-1) des machines.

La présente norme donne un cadre spécifique au secteur des machines pour la sécurité fonctionnelle d'un SRECS de machine. Elle couvre uniquement les aspects du cycle de vie de sécurité relatifs à l'allocation des exigences de sécurité jusqu'à la validation de la sécurité. Des exigences sont données pour information pour une utilisation sûre des SRECS de machines, lesquelles peuvent aussi être appropriées pour des phases ultérieures de la vie d'un SRECS.

Il existe plusieurs circonstances dans les machines où on utilise les SRECS comme partie des mesures de sécurité développées pour réaliser la réduction de risque. Un exemple typique est l'utilisation d'un protecteur avec dispositif de verrouillage qui, lorsqu'il est ouvert pour autoriser l'accès à la zone dangereuse, signale au système de commande électrique d'arrêter le fonctionnement dangereux de la machine. Egalement en automatisation, le système de commande électrique utilisé pour réaliser le fonctionnement correct du processus machine contribue souvent à la sécurité en réduisant les risques associés aux phénomènes dangereux résultant directement de défaillances du système de commande. La présente norme donne une méthodologie et les exigences pour:

- assigner le niveau d'intégrité de sécurité prescrit pour chaque fonction de commande relative à la sécurité devant être réalisée par les SRECS;
- permettre la conception des SRECS appropriés à la(aux) fonction(s) de commande relative à la sécurité assignée(s);
- intégrer les sous-systèmes relatifs à la sécurité conçus selon l'ISO 13849;
- valider les SRECS.

La présente norme internationale est prévue pour être utilisée dans le cadre de la réduction systématique du risque décrite dans l'ISO 12100-1 et conjointement avec l'appréciation du risque selon les principes décrits dans l'ISO 14121 (EN 1050). Une méthodologie conseillée pour l'attribution des niveaux d'intégrité de sécurité (SIL) est donnée dans l'Annexe informative A.

Des mesures sont indiquées pour coordonner le fonctionnement des SRECS avec la réduction de risque prévue en prenant en compte les probabilités et les conséquences d'anomalies systématiques ou aléatoires dans le système de commande électrique.

La Figure 1 montre les relations de la présente norme avec les autres normes appropriées.

Le Tableau 1 donne des conseils pour l'utilisation recommandée de la présente norme et de la révision de l'ISO 13849-1.

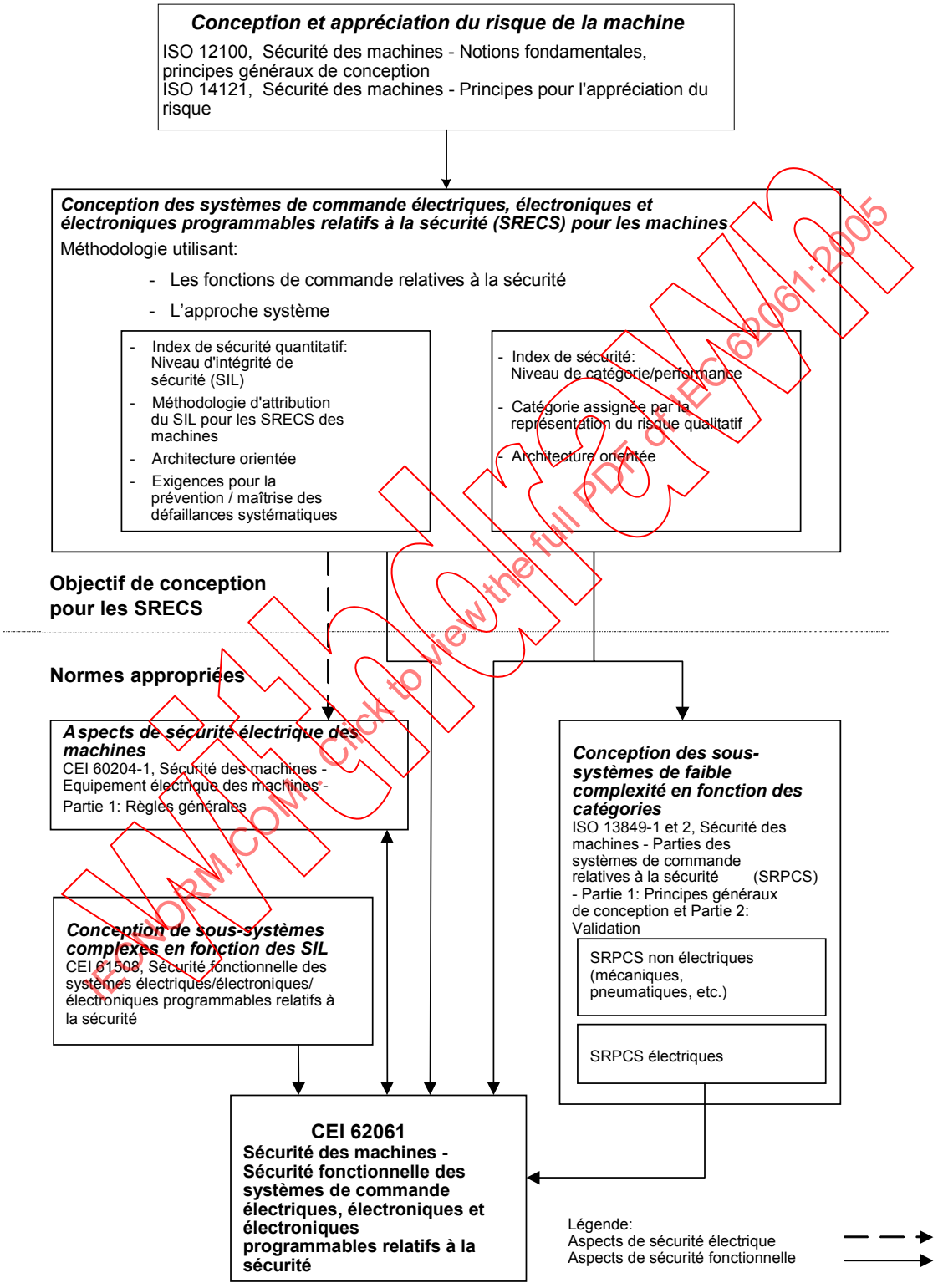


Figure 1 – Relations de la CEI 62061 avec les autres normes appropriées

Information sur l'utilisation recommandée de la CEI 62061 et de l'ISO 13849-1 (révision)

La CEI 62061 et l'ISO 13849-1 (révision) spécifient les exigences pour la conception et la réalisation des systèmes de commande électriques relatifs à la sécurité des machines. Utiliser l'une quelconque de ces normes, en accord avec leurs domaines d'application, peut présumer de la satisfaction des exigences essentielles de sécurité appropriées. Le Tableau 1 résume les domaines d'application de la CEI 62061 et de l' ISO 13849-1(révision).

NOTE L'ISO 13849-1 (révision) est en cours de préparation au sein de l'ISO TC 199 et du CEN TC 114.

Tableau 1 – Utilisation recommandée de la CEI 62061 et de l'ISO 13849-1 (révision)

	Technologie mettant en œuvre la(les) fonction(s) de commande relative(s) à la sécurité		**ISO 13849-1 (révision)**	**CEI 62061**
A	Non électrique, par exemple hydraulique		X	Non couvert
B	Electromécanique, par exemple relais ou électronique non complexe		Limité aux architectures désignées (voir Note 1) et jusqu'à PL=e	Toutes architectures et jusqu'à SIL 3
C	Electronique complexe, par exemple programmable		Limité aux architectures désignées (voir Note 1) et jusqu'à PL=d	Toutes architectures et jusqu'à SIL 3
D	A combiné avec B		Limité aux architectures désignées (voir Note 1) et jusqu'à PL=e	X voir Note 3
E	C combiné avec B		Limité aux architectures désignées (voir Note 1) et jusqu'à PL=d	Toutes architectures et jusqu'à SIL 3
F	C combiné avec A, ou C combiné avec A et B		X voir Note 2	X voir Note 3

"X" indique que ce cas est traité par la norme indiquée en tête de colonne.

NOTE 1 Les architectures désignées sont définies dans l'Annexe B de l'EN ISO 13849-1 (révision) afin de fournir une approche simplifiée de la quantification du niveau de performance.

NOTE 2 Pour l'électronique complexe: utilisation des architectures désignées selon l'EN ISO 13849-1 (révision) jusqu'à PL=d ou toute architecture selon la CEI 62061.

NOTE 3 Pour la technologie non électrique, utilisation des parties en tant que sous-systèmes selon l'EN ISO 13849-1 (révision).

SÉCURITÉ DES MACHINES – SÉCURITÉ FONCTIONNELLE DES SYSTÈMES DE COMMANDE ÉLECTRIQUES, ÉLECTRONIQUES ET ÉLECTRONIQUES PROGRAMMABLES RELATIFS À LA SÉCURITÉ

1 Domaine d'application

La présente Norme internationale spécifie les exigences et donne des recommandations pour la conception, l'intégration et la validation des systèmes de commande électriques, électroniques et électroniques programmables relatifs à la sécurité (SRECS) pour les machines (voir Notes 1 et 2). Elle s'applique aux systèmes de commande utilisés séparément ou en combinaison, pour assurer des fonctions de commande relatives à la sécurité de machines qui ne sont pas portables à la main en fonctionnement, y compris un groupe de machines fonctionnant ensemble d'une manière coordonnée.

NOTE 1 Dans la présente norme, le terme "systèmes de commande électrique" est utilisé à la place de "systèmes de commande électrique, électronique et électronique programmable (E/E/PE)" et "SRECS" est utilisé à la place de "systèmes de commande électrique, électronique et électronique programmable relatifs à la sécurité"

NOTE 2 Dans la présente norme, il est présumé que la conception des sous-systèmes électroniques programmables complexes ou des éléments de sous-systèmes est conforme aux exigences appropriées de la CEI 61508. La présente norme fournit une méthodologie pour l'utilisation, plus que pour le développement, de tels sous-systèmes et éléments de sous-systèmes en tant que partie d'un SRECS.

La présente norme est une norme d'application et n'est pas destinée à limiter ou inhiber les progrès technologiques. Elle ne couvre pas toutes les exigences (par exemple protection, verrouillage non électrique ou commande non électrique) qui sont nécessaires ou prescrites par d'autres normes ou réglementations destinées à protéger les personnes des dangers. Chaque type de machine a des exigences propres qui doivent être prises en compte pour obtenir une sécurité adéquate.

La présente norme:

– ne concerne que les exigences de sécurité fonctionnelle destinées à réduire le risque de blessure ou d'atteinte à la santé des personnes à proximité immédiate de la machine et de celles directement impliquées dans l'utilisation de la machine;

– se limite aux risques résultant directement des phénomènes dangereux de la machine elle-même ou d'un groupe de machines fonctionnant ensemble d'une manière coordonnée;

NOTE 3 Les exigences pour réduire les risques provenant d'autres phénomènes dangereux sont données dans les normes sectorielles appropriées. Par exemple, si une(des) machine(s) fait(font) partie d'une activité processus, il convient que les exigences de sécurité fonctionnelle du système de commande électrique de la machine satisfassent, en plus, à d'autres exigences (par exemple la CEI 61151) sous réserve que le processus soit concerné.

– ne spécifie pas les exigences de fonctionnement des éléments de commande (par exemple hydraulique, pneumatique) non électriques des machines;

NOTE 4 Bien que les exigences de la présente norme soient particulières aux systèmes de commande électriques, le cadre et la méthodologie spécifiés peuvent s'appliquer à des parties de systèmes de commande relatives à la sécurité utilisant d'autres technologies.

– ne couvre pas les phénomènes dangereux électriques provenant du matériel de commande électrique lui-même (par exemple choc électrique – voir la CEI 60204-1);

Les objectifs des articles particuliers à la CEI 62061 sont donnés dans le Tableau 2:

Tableau 2 – Vue générale et objectifs de la CEI 62061

Article	Objectifs
4: Gestion de la sécurité fonctionnelle	Spécifier les activités techniques et de gestion nécessaires pour la réalisation de la sécurité fonctionnelle prescrite des SRECS
5: Exigences pour la spécification des fonctions de commande relatives à la sécurité	Etablir les procédures de spécification des exigences pour les fonctions de commande relatives à la sécurité. Ces exigences s'expriment en termes de spécification des exigences fonctionnelles, et spécification des exigences d'intégrité de sécurité.
6: Conception et intégration des systèmes de commande électrique relatifs à la sécurité	Spécifier les critères de choix et/ou les méthodes de conception et de réalisation des SRECS pour satisfaire aux exigences de sécurité fonctionnelle. Ceci comprend: le choix de l'architecture système le choix des parties matériel et logiciel relatives à la sécurité la conception des parties matériel et logiciel la vérification que les exigences de sécurité fonctionnelle sont satisfaites par les parties matériel et logiciel ainsi conçues
7: Information pour l'utilisation de la machine	Spécifier les exigences pour l'information concernant l'utilisation des SRECS, qui est à fournir avec la machine. Ceci comprend vérifier: la fourniture d'un manuel utilisateur et de procédures pour l'utilisateur la fourniture d'un manuel d'entretien et de procédures d'entretien
8: Validation du système de commande électrique relatif à la sécurité	Spécifier les exigences pour la procédure de validation qui est à appliquer aux SRECS. Ceci comprend vérifier l'examen et l'essai du SRECS mis en service afin de s'assurer qu'il réalise les exigences établies dans la spécification des exigences de sécurité.
9: Modification du système de commande électrique relatif à la sécurité	Spécifier les exigences pour la procédure de modification qui est à appliquer lors de la modification des SRECS. Ceci comprend vérifier que: les modifications de tout SRECS sont correctement planifiées, vérifiées avant de procéder à la modification; la spécification des exigences de sécurité des SRECS est satisfaite après mise en œuvre de la modification.

2 Références normatives

Les documents référencés suivants sont indispensables pour l'application de ce document. Pour des références datées, seule l'édition citée s'applique. Pour les références non datées, c'est la dernière édition du document référencé (y compris tous les amendements) qui s'applique.

CEI 60204–1, *Sécurité des machines – Equipement électrique des machines – Partie 1: Règles générales*

CEI 61000-6-2, *Compatibilité électromagnétique (CEM) – Partie 6-2: Normes génériques – Immunité pour les environnements industriels*

CEI 61310 (toutes les parties), *Sécurité des machines – Indication, marquage et manoeuvre*

CEI 61508-2, *Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité – Partie 2: Prescriptions pour les systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité*

CEI 61508-3, *Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité – Partie 3: Prescriptions concernant les logiciels*

ISO 12100-1:2003, *Sécurité des machines – Notions fondamentales, principes généraux de conception – Partie 1: Terminologie de base , méthodologie*

ISO 12100-2:2003, *Sécurité des machines – Notions fondamentales, principes généraux de conception – Partie 2: Principes techniques*

ISO 13849-1:1999, *Sécurité des machines – Parties des systèmes de commande relatives à la sécurité – Partie 1: Principes généraux de conception*

ISO 13849-2:2003, *Sécurité des machines – Parties des systèmes de commande relatives à la sécurité – Partie 2: Validation*

ISO 14121, *Sécurité des machines – Principes pour l'appréciation du risque*

3 Termes, définitions et abréviations

3.1 Liste alphabétique des définitions

Terme	Numéro de définition
architecture	3.2.35
bloc fonctionnel	3.2.32
composant complexe	3.2.8
composant de faible complexité	3.2.7
contrainte architecturale	3.2.36
couverture du diagnostic	3.2.38
défaillance	3.2.39
défaillance aléatoire du matériel	3.2.44
défaillance dangereuse	3.2.40
défaillance de cause commune	3.2.43
défaillance en sécurité	3.2.41
défaillance systématique	3.2.45
défaut	3.2.30
durée moyenne de fonctionnement avant défaillance (MTTF)	3.2.34
élément de bloc fonctionnel	3.2.33
élément de sous-système	3.2.6
fonction de commande	3.2.14

fonction de commande relative à la sécurité (SRCF)	3.2.16
fonction de diagnostic du SRECS	3.2.17
fonction de sécurité	3.2.15
fonction réaction à l'anomalie du SRECS	3.2.18
intégrité de sécurité	3.2.19
intégrité de sécurité du logiciel	3.2.21
intégrité de sécurité du matériel	3.2.20
intégrité de sécurité systématique	3.2.22
langage de variabilité limitée (LVL)	3.2.49
langage de variabilité totale (FVL)	3.2.48
limite de revendication de SIL	3.2.24
logiciel d'application	3.2.46
logiciel intégré logiciel embarqué	3.2.47
logiciel relatif à la sécurité	3.2.50
machine	3.2.1
mesure de prévention	3.2.12
mode à faible sollicitation	3.2.26
mode à forte sollicitation ou continu	3.2.27
niveau d'intégrité de sécurité (SIL)	3.2.23
phénomène dangereux (pour les machines) risque (au sens de phénomène dangereux)	3.2.10
probabilité de défaillance dangereuse par heure (PFH_D)	3.2.28
proportion de défaillances en sécurité	3.2.42
risque	3.2.13
sécurité fonctionnelle	3.2.9
situation dangereuse	3.2.11
sollicitation	3.2.25
sous-système	3.2.5
système de commande de la machine	3.2.2
système de commande électrique	3.2.3
système de commande électrique relatif à la sécurité (SRECS)	3.2.4
test périodique	3.2.37
tolérance aux anomalies	3.2.31
valeur cible des défaillances	3.2.29
validation	3.2.52
vérification	3.2.51

3.2 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.2.1
machine
ensemble de pièces ou d'organes liés entre eux, dont au moins un est mobile, auxquels sont associés, selon les besoins, des actionneurs, des circuits de commande et de puissance, réunis de façon solidaire en vue d'une application définie, notamment pour la transformation, le traitement, le déplacement et le conditionnement d'un matériau.

Le terme "machine" désigne aussi un ensemble de machines qui, afin de concourir à un même résultat, sont disposées et commandées de manière à être solidaires dans leur fonctionnement

[ISO 12100-1:2003, 3.1]

3.2.2
système de commande de la machine
système qui répond à un signal d'entrée provenant, par exemple du processus, d'autres éléments de la machine, d'un opérateur ou d'un matériel de commande externe, et génère une(des) sortie(s) conduisant la machine à se comporter de la manière prévue

3.2.3
système de commande électrique
ensemble des parties électriques, électroniques et électroniques programmables du système de commande de la machine utilisées pour assurer, par exemple, les fonctions de commande du fonctionnement, de contrôle, de verrouillage, de communication, de protection et les fonctions de commandes relatives à la sécurité

NOTE Les fonctions de commande relatives à la sécurité peuvent être réalisées par un système de commande électrique solidaire ou indépendant des parties du système de commande de la machine qui réalisent les fonctions non relatives à la sécurité.

3.2.4
système de commande électrique relatif à la sécurité (SRECS)
système de commande électrique d'une machine dont la défaillance peut provoquer un accroissement immédiat du (des) risque(s)

NOTE Un SRECS comprend toutes les parties d'un système de commande électrique dont la défaillance peut provoquer une diminution ou une perte de la sécurité fonctionnelle et celui-ci peut comprendre à la fois des circuits électriques de puissance et des circuits de commande.

3.2.5
sous-système
entité de la conception de l'architecture générale du SRECS dans laquelle une défaillance d'un sous-système quelconque entraînera une défaillance d'une fonction de commande relative à la sécurité

NOTE 1 Un sous-système complet peut être constitué à partir d'un nombre d'éléments de sous-système identifiables et séparés qui, lorsqu'ils sont associés, réalisent les blocs fonctionnels alloués au sous-système.

NOTE 2 Cette définition est une restriction à la définition générale donnée dans la CEI 61508-4: ensemble d'éléments qui interagissent selon un modèle précis, un élément pouvant être un autre système, appelé sous-système, lequel peut être composé de matériel, de logiciel en interaction avec l'être humain.

NOTE 3 Cette définition diffère du langage courant où le terme "sous-système" peut signifier une quelconque partie subdivisée d'une entité; le terme "sous-système" est utilisé dans la présente norme dans une hiérarchie terminologique bien déterminée: le "sous-système" est le premier niveau de subdivision d'un système. Les parties résultant de subdivisions ultérieures d'un sous-système sont appelées "éléments de sous-systèmes".

3.2.6
élément de sous-système
partie d'un sous-système comprenant un composant unique ou un quelconque groupe de composants

3.2.7
composant de faible complexité
composant dans lequel

– les modes de défaillance sont bien définis; et

– le comportement en conditions de défaut peut être complètement déterminé

[CEI 61508-4, 3.4.4 modifiée]

NOTE 1 Le comportement du composant de faible complexité dans des conditions de défaut peut être déterminé par des méthodes analytiques et/ou d'essai.

NOTE 2 Un sous-système ou un élément de sous-système qui comprend un ou plusieurs interrupteurs de fin de course, faisant fonctionner, éventuellement via des relais électromécaniques interposés, un ou plusieurs contacteurs destinés à couper l'alimentation de moteurs électriques est un exemple de composant de faible complexité.

3.2.8
composant complexe
composant dans lequel
– les modes de défaillance ne sont pas bien définis; ou
– le comportement en conditions de défaut ne peut être complètement déterminé

3.2.9
sécurité fonctionnelle
partie de la sécurité de la machine et du système de commande de la machine qui dépend du fonctionnement correct des SRECS, des systèmes relatifs à la sécurité basés sur une autre technologie et des dispositifs externes de réduction de risque

[CEI 61508-4, 3.1.9 modifiée]

NOTE 1 La présente norme prend en considération uniquement la sécurité fonctionnelle dépendant du fonctionnement correct du SRECS dans les applications machines.

NOTE 2 Le Guide ISO/CEI 51 définit la sécurité comme l'absence de risque inacceptable.

3.2.10
phénomène dangereux (pour les machines)
risque (au sens de phénomène dangereux)
source potentielle de blessure physique ou d'atteinte à la santé

[ISO 12100-1: 2003, 3.6 modifiée]

NOTE 1 Le terme "risque" peut être qualifié de manière à faire apparaître l'origine (par exemple: risque de choc électrique, risque d'écrasement, risque de coupure, risque d'intoxication, risque d'incendie).

3.2.11
situation dangereuse
situation dans laquelle une personne est exposée à un(des) phénomène(s) dangereux.

[ISO 12100-1:2003, 3.9 modifiée]

3.2.12
mesure de prévention
mesure destinée à réduire le risque

[ISO 12100-1:2003, 3.18 modifiée]

3.2.13
risque
combinaison de la probabilité d'un dommage et de la gravité de ce dommage

ISO 12100-1:2003, 3.11]

3.2.14
fonction de commande
fonction qui évalue les informations ou signaux d'entrée et génère des activités ou informations de sortie

3.2.15
fonction de sécurité
fonction d'une machine dont la défaillance peut provoquer un accroissement immédiat du (des) risque(s)

[ISO 12100-1:2003, 3.28]

NOTE Cette définition diffère des définitions de la CEI 61508-4 et de l'ISO 13849-1.

3.2.16
fonction de commande relative à la sécurité
SRCF
fonction de commande mise en oeuvre par un SRECS avec un niveau d'intégrité spécifié, prévue pour maintenir la condition de sécurité de la machine ou empêcher un accroissement immédiat du(des) risque(s)

3.2.17
fonction de diagnostic du SRECS
fonction prévue pour détecter les anomalies dans un SRECS et fournir une activité ou une information de sortie déterminée en cas de détection d'une anomalie

NOTE Cette fonction est prévue pour détecter des anomalies qui pourraient entraîner une défaillance dangereuse d'une SCRF et initier une fonction réaction à l'anomalie déterminée.

3.2.18
fonction réaction à l'anomalie du SRECS
fonction qui est initiée lorsqu'une anomalie à l'intérieur d'un SRECS est détectée par la fonction diagnostic du SRECS

3.2.19
intégrité de sécurité
probabilité pour qu'un SRECS ou ses sous-systèmes exécutent de manière satisfaisante les fonctions de commande relatives à la sécurité requises dans toutes les conditions spécifiées

[CEI 61508-4, 3.5.2 modifiée]

NOTE 1 Plus le niveau d'intégrité de sécurité de l'entité est élevé, plus la probabilité d'une défaillance de cette entité dans l'exécution de la fonction de commande relative à la sécurité requise est faible.

NOTE 2 L'intégrité de sécurité comprend l'intégrité de sécurité du matériel (voir 3.2.20) ainsi que l'intégrité de sécurité systématique (voir 3.2.22).

3.2.20
Intégrité de sécurité du matériel
partie de l'intégrité de sécurité d'un SRECS ou de ses sous-systèmes comprenant les exigences relatives à la fois à la probabilité de défaillance aléatoire du matériel et de contraintes architecturales

[CEI 61508-4, 3.5.5 modifiée]

3.2.21
intégrité de sécurité du logiciel
partie de l'intégrité de sécurité systématique d'un SRECS ou de ses sous-systèmes relative à la capacité du logiciel dans un système électronique programmable à réaliser ses fonctions de commande relatives à la sécurité dans toutes les conditions spécifiées et pendant une période de temps spécifiée

[CEI 61508-4, 3.5.3 modifiée]

NOTE L'intégrité de sécurité du logiciel ne peut habituellement pas être quantifiée.

3.2.22
intégrité de sécurité systématique
partie de l'intégrité de sécurité d'un SRECS ou de ses sous-systèmes qui se rapporte à sa résistance aux défaillances systématiques (voir 3.2.45) dans un mode dangereux

[CEI 61508-4, 3.5.4 modifiée]

NOTE 1 L'intégrité de sécurité systématique ne peut habituellement pas être quantifiée.

NOTE 2 Les exigences pour l'intégrité de sécurité systématique s'appliquent à la fois pour les aspects matériel et logiciel d'un SRECS ou de ses sous-systèmes.

3.2.23
niveau d'intégrité de sécurité
SIL
niveau discret (parmi trois possibles) permettant de spécifier les exigences concernant l'intégrité de sécurité des fonctions de commande relatives à la sécurité à allouer aux SRECS, le niveau 3 d'intégrité de sécurité possédant le plus haut degré d'intégrité et le niveau 1 possédant le plus bas

[CEI 61508-4, 3.5.6 modifiée]

NOTE Le SIL 4 n'est pas pris en compte dans la présente norme car il n'est pas approprié aux exigences de réduction du risque normalement associées aux machines. Pour les exigences applicables au SIL 4, voir la CEI 61508-1 et la CEI 61508-2.

3.2.24
limite de revendication de SIL (pour un sous-système)
SILCL
SIL maximal qui peut être revendiqué pour un sous-système de SREC en relation avec des contraintes architecturales et l'intégrité de sécurité systématique

3.2.25
sollicitation
événement qui conduit le SRECS à réaliser sa SRCF

3.2.26
mode à faible sollicitation
mode de fonctionnement dans lequel la fréquence des sollicitations sur un SRECS n'est pas supérieure à une par an et pas supérieure à deux fois la fréquence des tests périodiques

NOTE Le matériel qui est conçu uniquement selon les exigences associées à un mode fonctionnement à faible sollicitation dans la CEI 61508-1 et la CEI 61508-2 peut se révéler inadapté à un usage en tant que partie d'un SRECS au sens de la présente norme. Le mode de fonctionnement à faible sollicitation n'est pas considéré approprié pour les applications des SRECS aux machines.

3.2.27
mode à forte sollicitation ou continu
mode de fonctionnement dans lequel la fréquence des sollicitations sur un SRECS est supérieure à une par an ou supérieure à deux fois la fréquence des tests périodiques

[CEI 61508-4, 3.5.12 modifiée]

NOTE 1 Le mode de fonctionnement à faible sollicitation n'est pas considéré approprié pour les applications des SRECS aux machines. De ce fait, dans la présente norme, les SRECS sont uniquement considérés fonctionner dans le mode à forte sollicitation ou continu.

NOTE 2 Le mode sollicitation signifie qu'une fonction de commande relative à la sécurité ne peut être réalisée que suite à une requête (sollicitation) afin que la machine passe dans un état spécifié. Le SRECS n' a aucune influence sur la machine tant qu'il n'y a pas de sollicitation de la fonction de commande relative à la sécurité.

NOTE 3 Le mode continu signifie qu'une fonction de commande relative à la sécurité est réalisée continuellement (sans interruption), c'est-à-dire que le SRECS commande de façon continue la machine et qu'une défaillance (dangereuse) de sa fonction peut entraîner un phénomène dangereux.

3.2.28
probabilité de défaillance dangereuse par heure
***PFH*$_D$**
probabilité moyenne de défaillance dangereuse en 1 h

NOTE Il convient de ne pas confondre le PFH_D avec la probabilité de défaillance sur sollicitation (PFD).

3.2.29
valeur cible des défaillances
PFH_D prévisionnel à réaliser pour satisfaire à une(des) exigence(s) particulière(s) d'intégrité de sécurité

NOTE La valeur cible des défaillances est spécifiée en termes de probabilité de défaillance dangereuse par heure.

[CEI 61508-4, 3.5.13 modifiée]

3.2.30
défaut
condition anormale qui peut entraîner une réduction de capacité ou la perte de capacité d'un SRECS, d'un sous-système ou d'un élément de sous-système à accomplir une fonction requise

[CEI 61508-4, 3.6.1 modifiée]

3.2.31
tolérance aux anomalies
aptitude d'un SRECS, d'un sous-système ou d'un élément de sous-système à continuer d'accomplir une fonction requise en présence d'anomalies ou d'erreurs

[CEI 61508-4, 3.6.3 modifiée]

3.2.32
bloc fonctionnel
plus petit élément d'une SRCF dont la défaillance peut entraîner une défaillance de la SRCF

NOTE 1 Dans la présente norme, une SRCF (F) peut être vue comme un ET logique des blocs fonctionnels (FB), c'est-à-dire F = FB_1 et FB_2 et ... et FB_n.

NOTE 2 Cette définition d'un bloc fonctionnel diffère de celles utilisées dans la CEI 61131-3 et d'autres normes.

3.2.33
élément de bloc fonctionnel
partie de bloc fonctionnel

3.2.34
durée moyenne de fonctionnement avant défaillance
MTTF
espérance mathématique de la durée de fonctionnement avant défaillance

[VEI 191-12-07, modifiée]

NOTE La MTTF représente normalement une valeur moyenne de l'espérance mathématique de la durée de fonctionnement avant défaillance.

3.2.35
architecture
configuration spécifique des éléments matériels et logiciels dans un SRECS

[CEI 61508-4, 3.3.5 modifiée]

3.2.36
contrainte architecturale
ensemble des exigences d'architecture qui limitent le SIL pouvant être revendiqué pour un sous-système

NOTE Les exigences pour les contraintes architecturales sont données en 6.7.6.

3.2.37
test périodique
test qui peut détecter les défaillances et la dégradation d'un SRECS et de ses sous-systèmes de telle sorte que, lorsque nécessaire, le SRECS et ses sous-systèmes puissent être rétablis dans une condition “comme neuf’’ ou dans une condition aussi proche que possible de celle-ci

[CEI 61508-4, 3.8.5 modifiée]

NOTE Un test périodique est prévu afin de confirmer que le SRECS est en condition d'assurer l'intégrité de sécurité spécifiée.

3.2.38
couverture du diagnostic
décroissance de la probabilité de défaillance dangereuse du matériel résultant du fonctionnement des tests de diagnostic automatique

[CEI 61508-4, 3.8.6, modifiée]

NOTE La couverture du diagnostic (DC) peut être calculée en utilisant l'équation suivante:

$$DC = \Sigma \lambda_{DD} / \lambda_{Dtotal}$$

où λ_{DD} est le taux de défaillance dangereuse détectée pour le matériel et λ_{Dtotal} est le taux de toutes les défaillances dangereuses pour le matériel.

3.2.39
défaillance
cessation de l’aptitude d’un SRECS, d'un sous-système ou d'un élément de sous-système à accomplir une fonction requise

[CEI 61508-4, 3.6.4, modifiée et ISO 12100-1:2003, 3.32]

NOTE Les défaillances sont soit aléatoires (pour le matériel) ou systématiques (pour le matériel ou le logiciel).

3.2.40
défaillance dangereuse
défaillance d'un SRECS, d'un sous-système ou d'un élément de sous-système qui a la potentialité de provoquer un phénomène dangereux ou un état de non-fonctionnement

[CEI 61508-4, 3.6.7, modifiée]

NOTE 1 Le fait que cette potentialité se réalise ou non peut dépendre de l’architecture du canal du système; par exemple pour les systèmes ayant plusieurs canaux pour accroître la sécurité, il est moins probable qu’une défaillance dangereuse du matériel conduise à un état dangereux de l’ensemble ou à un état dans lequel la fonction ne peut plus être exécutée.

NOTE 2 Dans un sous-système ayant plusieurs canaux, la probabilité de défaillance dangereuse du sous-système peut être plus faible que le taux de défaillance dangereuse d'un canal qui constitue le sous-système. La probabilité de défaillance dangereuse d'un SRECS ne peut être plus faible que celle d'un quelconque sous-système constituant le SRECS. (Ceci résulte de la définition particulière d'un "sous-système" dans la présente norme.)

NOTE 3 Une défaillance dangereuse se traduit normalement par une défaillance ou une défaillance potentielle dans la réalisation de la SRCF.

3.2.41
défaillance en sécurité
défaillance d'un SRECS, d'un sous-système d'un SRECS ou d'un élément de sous-système d'un SRECS qui n'a pas la potentialité de provoquer un phénomène dangereux

[CEI 61508-4, 3.6.8, modifiée]

3.2.42
proportion de défaillances en sécurité
SFF
proportion du taux global des défaillances d'un sous-système qui n'entraînent pas une défaillance dangereuse

NOTE La proportion de défaillances en sécurité (*SFF*) peut être calculée en utilisant l'équation suivante:

$$(\Sigma\lambda_S + \Sigma\lambda_{DD}) / (\Sigma\lambda_S + \Sigma\lambda_D)$$

où

λ_S est le taux de défaillance en sécurité,

$\Sigma\lambda_S + \Sigma\lambda_D$ est le taux global des défaillances,

λ_{DD} est le taux de défaillance dangereuse qui est détecté par les fonctions de diagnostic, et

λ_D est le taux de défaillance dangereuse.

La couverture du diagnostic (si nécessaire) de chaque sous-système du SRECS est prise en compte dans le calcul de la probabilité de défaillance aléatoire du matériel. La proportion de défaillances en sécurité est prise en compte lors de la détermination des contraintes architecturales sur l'intégrité de sécurité du matériel (voir 6.7.7).

3.3

3.2.43
défaillance de cause commune
CCF
défaillance, résultat d'un ou plusieurs événements, entraînant des défaillances simultanées sur deux ou plusieurs canaux séparés dans un sous-système à canaux multiples (architecture redondante), entraînant la défaillance d'une SRCF

[CEI 61508-4, 3.6.10 modifiée]

NOTE La présente définition diffère de celles données dans l'ISO 12100-1 et dans le VEI 191-04-23.

3.2.44
défaillance aléatoire du matériel
défaillance survenant de manière aléatoire et résultant de divers mécanismes de dégradation au sein du matériel

[CEI 61508-4, 3.6.5]

3.2.45
défaillance systématique
défaillance reliée de façon déterministe à une certaine cause, ne pouvant être éliminée que par une modification de la conception ou du processus de fabrication, des procédures d'exploitation, de la documentation ou d'autres facteurs appropriés

[CEI 61508-4, 3.6.6]

NOTE 1 La maintenance corrective sans modification n'élimine pas, habituellement, la cause de la défaillance.

NOTE 2 Une défaillance systématique peut être induite en simulant la cause de la défaillance.

NOTE 3 Citons comme exemples de causes de défaillances systématiques les erreurs humaines telles que:

- les erreurs de spécification des exigences de sécurité;
- les erreurs de conception, fabrication, installation, exploitation du matériel;
- les erreurs de conception et/ou de mise en œuvre du logiciel.

3.2.46
logiciel d'application
logiciel spécifique à l'application qui a été réalisé par le concepteur du SRECS et qui contient généralement des séquences logiques, des termes et expressions logiques qui commandent l'entrée, la sortie, les calculs appropriés et les décisions nécessaires pour satisfaire aux exigences fonctionnelles du SRECS

3.2.47
logiciel intégré
logiciel embarqué
logiciel, fourni par le constructeur, qui fait partie d'un SRECS et qui n'est pas normalement accessible pour des modifications

NOTE Les micro-logiciel et logiciel système sont des exemples de logiciel intégré.

3.2.48
langage de variabilité totale
FVL
type de langage qui fournit la possibilité de mettre en œuvre une gamme étendue de fonctions et d'applications

[CEI 61511-1, 3.2.81.1.3 modifiée]

NOTE 1 Un exemple typique de système utilisant le FVL est l'ordinateur d'usage général.

NOTE 2 Le FVL se trouve normalement dans les logiciels intégrés et rarement dans les logiciels d'application.

NOTE 3 Parmi les exemples de FVL on peut citer: l'Ada, le C, le Pascal, une liste d'instructions, les langages d'assemblage, le C++, le Java, le SQL.

3.2.49
langage de variabilité limitée
LVL
type de langage qui fournit la possibilité de combiner des fonctions de bibliothèque, prédéfinies, spécifiques à une application, pour mettre en oeuvre les spécifications des exigences concernant la sécurité

[CEI 61511-1, 3.2.81.1.2 modifiée]

NOTE 1 Un LVL fournit une correspondance fonctionnelle étroite avec les fonctions nécessaires pour réaliser l'application.

NOTE 2 Des exemples typiques de LVL sont donnés dans la CEI 61131-3. Ils comprennent: le langage à contacts, le langage en blocs fonctionnels et le diagramme fonctionnel en séquence. Les listes d'instructions et un texte structuré ne sont pas considérés comme LVL.

NOTE 3 Exemple typique des systèmes utilisant le LVL: un automate programmable (PLC) configuré pour la commande d'une machine.

3.2.50
logiciel relatif à la sécurité
logiciel utilisé pour accomplir des fonctions de commande relatives à la sécurité dans un système relatif à la sécurité

3.2.51
vérification
confirmation, par examen (par exemple tests, analyses) que le SRECS, ses sous-systèmes ou éléments de sous-systèmes satisfont aux exigences déterminées par la spécification appropriée

[CEI 61508-4, 3.8.1 modifiée et CEI 61511-1, 3.2.92 modifiée]

NOTE Les résultats de la vérification doivent être documentés afin de fournir des preuves tangibles.

EXEMPLE: Citons comme exemples d'activités de vérification:

- les revues relatives aux sorties d'une phase (documents concernant toutes les phases du cycle de vie de sécurité) destinées à assurer la conformité avec les objectifs et exigences de la phase, et prenant en compte les entrées spécifiques à cette phase;
- les revues de conception;
- les tests réalisés sur les produits mis au point afin de s'assurer que leur fonctionnement est conforme à leur spécification;
- les tests d'intégration réalisés lors de l'assemblage élément par élément de différentes parties d'un système, à partir de tests d'environnement, afin de s'assurer que toutes les parties fonctionnent les unes avec les autres conformément aux spécifications.

3.2.52
validation
confirmation par examen (par exemple tests, analyses) que le SRECS satisfait aux exigences de sécurité fonctionnelle pour une application spécifique

[CEI 61508-4, 3.8.2 modifiée]

3.4 Abréviations

Les abréviations suivantes sont utilisées dans la présente norme.

CCF	défaillance(s) de cause commune
DC	couverture du diagnostic
CEM	compatibilité électromagnétique
FB	bloc fonctionnel
FVL	langage de variabilité totale
I/O	entrée/sortie
LVL	langage de variabilité limitée
PFH_D	probabilité de défaillance dangereuse par heure
MTTF	durée moyenne de fonctionnement avant défaillance
MTTR	durée moyenne de panne
P_{TE}	probabilité d'erreur de transmission dangereuse
SFF	proportion de défaillances en sécurité
SIL	niveau d'intégrité de sécurité
SILCL	limite de revendication de SIL (pour un sous-système)
S-R	relatif à la sécurité
SRECS	système de commande électrique relatif à la sécurité
SRCF	fonction de commande relative à la sécurité
SRS	spécification des exigences de sécurité
SYS	système

4 Gestion de la sécurité fonctionnelle

4.1 Objectifs

Le présent article spécifie les activités techniques et de gestion nécessaires à la réalisation de la sécurité fonctionnelle prescrite du SRECS.

4.2 Exigences

4.2.1 Un plan de sécurité fonctionnelle doit être dressé et documenté pour chaque projet de conception de SRECS, et doit être mis à jour autant que nécessaire. Le plan doit inclure les procédures de contrôle des activités spécifiées dans les Articles 5 à 9.

NOTE 1 Il convient que le sommaire du plan de sécurité fonctionnelle tienne compte des circonstances particulières, qui peuvent comprendre:

– la dimension du projet;
– le degré de complexité;
– le degré d'innovation de la conception et de la technologie;
– le degré de normalisation des particularités de conception;
– la(les) conséquence(s) possible(s) en cas de défaillance.

En particulier, le plan doit:

a) identifier les activités appropriées spécifiées dans les Articles 5 à 9;

b) décrire la politique et la stratégie pour satisfaire aux exigences de sécurité fonctionnelle spécifiées;

c) décrire la stratégie pour réaliser la sécurité fonctionnelle pour le développement, l'intégration, la vérification et la validation du logiciel d'application;

d) identifier les personnes, services ou autres unités ainsi que les ressources responsables de l'exécution et de la revue de chacune des activités spécifiées dans les Articles 5 à 9;

e) identifier ou établir les procédures et les ressources d'enregistrement et d'entretien des informations appropriées à la sécurité fonctionnelle d'un SRECS;

NOTE 2 Il convient de considérer:

- les résultats de l'identification des phénomènes dangereux et de l'appréciation du risque;
- les matériels utilisés pour les fonctions relatives à la sécurité ainsi que leurs exigences concernant la sécurité;
- l'organisation responsable du maintien de la sécurité fonctionnelle;
- les procédures nécessaires pour réaliser et maintenir la sécurité fonctionnelle (y compris les modifications des SRECS).

f) décrire la stratégie de gestion de configuration (voir 9.3) prenant en compte les points d'organisation, tels que les personnes autorisées et les structures internes de l'organisation;

g) établir un plan de vérification qui doit inclure:

– des informations sur le moment auquel la vérification doit être réalisée;
– des informations sur les personnes, services ou unités qui doivent réaliser la vérification;
– les critères de choix entre les stratégies et techniques de vérification;
– les critères de choix et d'utilisation des matériels de test;
– les critères de choix des activités de vérification;
– les critères d'acceptation; et
– les moyens à utiliser pour l'évaluation des résultats de la vérification.

h) établir un plan de validation comprenant:

- des informations sur le moment auquel la validation doit être réalisée;
- l'identification des modes de fonctionnement appropriés de la machine (par exemple fonctionnement normal, réglage);
- les exigences au regard desquelles le SRECS est à valider;
- la stratégie technique de validation, par exemple les méthodes analytiques ou les tests statistiques;
- les critères d'acceptation; et
- les actions à mener en cas de non-satisfaction aux critères d'acceptation.

NOTE 3 Il convient que le plan de validation indique si le SRECS et ses sous-systèmes sont appelés à être soumis à des essais individuels de série, des essais de type et/ou des essais sur prélèvement.

4.2.2 Le plan de sécurité fonctionnelle doit être établi pour garantir un suivi rapide et une résolution satisfaisante des points appropriés à un SRECS liés:

- aux activités spécifiées dans les Articles 5 à 9;
- aux activités de vérification; et
- aux activités de validation.

5 Exigences pour la spécification des fonctions de commande relatives à la sécurité (SRCF)

5.1 Objectifs

Cet article établit les procédures de spécification des exigences de la(des) SRCF à réaliser par les SRECS.

5.2 Spécification des exigences pour les SRCF

5.2.1 Généralités

5.2.1.1 Résultant de la stratégie de réduction de risque, telle qu'énoncée dans l'ISO 12100-1, l'ISO 12100-2 et l'ISO 14121, tous les besoins de fonctions de sécurité seront déterminés.

5.2.1.2 Si des fonctions de sécurité sont choisies pour être réalisées (entièrement ou en partie) par le SRECS, la(les) SRCF associée(s) (voir 3.2.16) doit(doivent) alors être spécifiée(s).

5.2.1.3 Les spécifications pour chaque SRCF doivent comprendre:

- la spécification des exigences fonctionnelles (voir 5.2.3);
- la spécification des exigences d'intégrité de sécurité (voir 5.2.4),

et celles-ci doivent être documentées dans la spécification des exigences de sécurité (SRS).

NOTE 1 Lorsqu'un matériel non électrique, associé à des moyens électriques, contribue au du fonctionnement d'une fonction de sécurité, la(les) valeur(s) cible(s) des défaillances applicables au matériel non électrique n'est (ne sont) pas prise(s) en considération dans la présente norme. On entend par moyens électriques tous les dispositifs ou systèmes fonctionnant à partir de principes électriques, y compris:

- les appareils électromécaniques;
- les appareils électroniques non programmables;
- les appareils électroniques programmables.

NOTE 2 Il est nécessaire de soumettre la SRS à un contrôle de la version dans le cadre des procédures de gestion de la configuration (voir 9.3).

5.2.1.4 La spécification des exigences de sécurité doit être vérifiée pour garantir sa cohérence et son exhaustivité pour l'usage prévu.

NOTE Celle-ci peut être réalisée par exemple par examen, analyse, listes de vérifications. Voir aussi B.2.6 de la CEI 61508-7.

5.2.2 Informations à mettre à disposition

Les informations suivantes doivent être utilisées pour produire à la fois la spécification des exigences fonctionnelles et la spécification des exigences d'intégrité de sécurité de chaque SRCF:

– les résultats de l'appréciation du risque pour la machine y compris toutes les fonctions de sécurité déterminées comme nécessaires au processus de réduction de risque pour chaque phénomène dangereux spécifique;
– les caractéristiques de fonctionnement de la machine, y compris:
  - les modes de fonctionnement,
  - la durée de cycle,
  - le fonctionnement en temps de réponse,
  - les conditions environnementales,
  - l'interaction de personne(s) avec la machine (par exemple, réparation, réglage, nettoyage);
– toute information appropriée aux SRCF qui peut avoir une influence sur la conception du SRECS y compris, par exemple:
  - une description du comportement de la machine que la SRCF est prévue réaliser ou empêcher;
  - toutes les interfaces entre les SRCF, et entre les SRCF et toute autre fonction (qu'elle soit à l'intérieur ou à l'extérieur de la machine);
  - les fonctions réaction à l'anomalie prescrites de la SRCF.

NOTE Il se pourrait qu'une partie des informations ne soit pas disponible ou suffisamment définie avant le démarrage du processus de conception itératif du SRECS; aussi, les spécifications des exigences de sécurité du SRECS peuvent nécessiter une mise à jour lors du processus de conception.

5.2.3 Spécification des exigences fonctionnelles pour les SRCF

5.2.3.1 La spécification des exigences fonctionnelles pour les SRCF doit décrire les informations concernant chaque SRCF à réaliser y compris, pour autant qu'applicable:

– la(les) condition(s) (par exemple mode de fonctionnement) de la machine dans laquelle la SRCF doit être active ou désactivée;
– la priorité associée à ces fonctions, lesquelles peuvent être actives simultanément et provoquer une action conflictuelle;
– la fréquence de fonctionnement pour chaque SRCF;
– le temps de réponse requis de chaque SRCF;
– l'(les)interface(s) des SRCF avec les autres composants de la machine;
– les temps de réponse requis (par exemple des dispositifs d'entrée et de sortie);
– une description de chaque SRCF;
– une description de la (des) fonction(s) réaction(s) à l'anomalie et de toutes les contraintes s'exerçant, par exemple, redémarrage ou fonctionnement continu de la machine, dans les cas où la première réaction à l'anomalie est d'arrêter la machine;

– une description de l'environnement de fonctionnement (par exemple température, humidité, poussière, substances chimiques, vibration mécanique et choc);

– les tests et toutes les facilités associées (par exemple matériel de test, ports d'accès test);

– les taux de cycles de manœuvres, le cycle de fonctionnement, et/ou la catégorie d'utilisation pour les dispositifs électromécaniques prévus dans la SRCF.

5.2.3.2 En complément des exigences de la CEI 61000-6-2, dans le cas où un SRECS est prévu pour un usage en environnement industriel, des niveaux d'immunité électromagnétique (EM) sont donnés en Annexe E. Il convient que les SRECS prévus pour un autre environnement EM (par exemple résidentiel) adoptent des niveaux d'immunité basés sur ceux spécifiés dans les différentes normes CEM (par exemple, pour un environnement résidentiel, selon la CEI 61000-6-1).

NOTE 1 Lors de la spécification des niveaux d'immunité EM, il est nécessaire de considérer si les niveaux employés dans les différentes normes CEM couvrent les cas pouvant se produire dans une application SRECS même avec une faible probabilité d'occurrence.

NOTE 2 Le critère d'aptitude à la fonction sur immunité EM pour la sécurité fonctionnelle d'un SRECS est donné en 6.4.3.

5.2.4 Spécification des exigences d'intégrité de sécurité pour les SRCF

5.2.4.1 Les exigences d'intégrité de sécurité pour chaque SRCF doivent découler de l'appréciation du risque afin de garantir que la réduction nécessaire du risque peut être réalisée. Dans la présente norme, une exigence d'intégrité de sécurité s'exprime en tant que valeur cible des défaillances pour la probabilité de défaillance dangereuse par heure de chaque SRCF.

5.2.4.2 Les exigences d'intégrité de sécurité pour chaque SRCF doivent être spécifiées en termes de SIL selon le Tableau 3 et documentées. Un exemple de méthodologie est donné en Annexe A.

Tableau 3 – Niveaux d'intégrité de sécurité: valeurs cibles des défaillances pour les SRCF

Niveau d'intégrité de sécurité	Probabilité de défaillance dangereuse par heure (PFH_D)
3	$\geq 10^{-8}$ à $< 10^{-7}$
2	$\geq 10^{-7}$ à $< 10^{-6}$
1	$\geq 10^{-6}$ à $< 10^{-5}$

NOTE Lorsque le niveau d'intégrité de sécurité requis d'une SRCF est inférieur au SIL1, il convient au minimum alors que les exigences de la catégorie B de l'ISO 13849-1 soient satisfaites.

5.2.4.3 Lorsqu'une norme produit spécifie un SIL pour une SRCF, alors celui-ci doit avoir priorité par rapport à l'Annexe A.

6 Conception et intégration des systèmes de commande électrique relatifs à la sécurité (SRECS)

6.1 Objectifs

L'objectif des exigences de cet article est de choisir ou concevoir un SRECS qui satisfasse aux exigences fonctionnelles et aux exigences d'intégrité de sécurité listées dans la spécification des exigences de sécurité (voir 5.2).

6.2 Exigences générales

6.2.1 Les SRECS doivent être choisis ou conçus pour satisfaire à la spécification des exigences de sécurité (voir 5.2) et lorsque c'est approprié, à la spécification des exigences de sécurité du logiciel (voir 6.10) prenant en compte les exigences appropriées de la présente norme.

6.2.2 Le choix ou la conception des SRECS (y compris l'architecture d'ensemble du matériel et du logiciel, les capteurs, les organes de commande, l'électronique programmable, le logiciel intégré, le logiciel d'application, etc.) doivent être conforme soit à 6.5 ou à 6.6. Quelle soit la méthode utilisée, les SRECS doivent satisfaire aux exigences suivantes:

a) les exigences pour l'intégrité de sécurité du matériel comprenant:
   – les contraintes architecturales sur l'intégrité de sécurité du matériel (voir 6.6.3.3); et
   – les exigences pour la probabilité de défaillance dangereuse aléatoire du matériel (voir 6.6.3.2);
b) les exigences pour l'intégrité de sécurité systématique (voir 6.4) comprenant:
   – les exigences pour l'évitement des défaillances, et
   – les exigences pour la maîtrise des anomalies systématiques;
c) les exigences relatives au comportement du SRECS sur détection d'anomalie (voir 6.3);
d) les exigences pour la conception et le développement du logiciel relatif à la sécurité (voir 6.10 et 6.11).

6.2.3 La conception des SRECS doit tenir compte des aptitudes et des limites humaines (y compris le mauvais usage raisonnablement prévisible) et doit convenir aux actions attribuées aux opérateurs, au personnel chargé de la maintenance et à quiconque pourrait interagir avec les SRECS. La conception de toutes les interfaces opérateurs doit être fondée sur de bonnes pratiques en termes de facteur humain (voir la série CEI 61310) et doit s'accommoder du niveau probable de formation et de connaissances des opérateurs comme, en particulier, pour les sous-systèmes de série destinés au grand public, où l'opérateur peut être un membre du public.

NOTE Il convient que l'objectif de la conception soit de prévenir ou d'éliminer, dans toute la mesure du possible, les erreurs humaines critiques prévisibles imputables aux opérateurs ou au personnel de maintenance. Dans le cas où cela n'est pas possible, il convient que d'autres moyens soient appliqués (par exemple une action manuelle avec confirmation secondaire avant finalisation) afin de réduire la possibilité d'erreurs opérateur et garantir que des erreurs prévisibles ne conduisent pas à un accroissement du risque.

6.2.4 La maintenabilité et la testabilité doivent être prises en compte lors des activités de conception et d'intégration afin de faciliter la mise en oeuvre de ces propriétés dans les SRECS.

6.2.5 La conception des SRECS, y compris leurs fonctions de diagnostic et réaction à l'anomalie, doit être documentée. Cette documentation doit:

– être précise, complète et concise;
– être adaptée pour son usage prévu;
– être accessible et actualisable;
– avoir été soumise à un contrôle de version.

6.2.6 Les résultats des activités réalisées pendant la conception, le développement et la réalisation des SRECS doivent être vérifiés à des stades appropriés.

6.3 Exigences comportementales (d'un SRECS) lors de la détection d'une anomalie dans le SRECS

6.3.1 La détection d'une anomalie dangereuse dans un quelconque sous-système ayant une tolérance aux anomalies du matériel supérieure à zéro doit déclencher l'exécution de la fonction réaction à l'anomalie spécifiée.

La spécification peut autoriser l'isolement de la partie du sous-système présentant l'anomalie afin de permettre la poursuite en sécurité de l'exploitation de la machine, pendant que la partie présentant une anomalie est réparée. Dans ce cas, si la partie présentant une anomalie n'est pas réparée durant le temps estimé maximal, pris comme hypothèse dans le calcul de la probabilité de défaillance aléatoire du matériel (voir 6.7.8), alors une seconde réaction à l'anomalie doit avoir lieu afin de maintenir un état sûr.

Si le SRECS est conçu pour être réparé en ligne, l'isolement de la partie présentant une anomalie ne doit être appliqué que si cela n'accroît pas la probabilité de défaillance dangereuse aléatoire du matériel du SRECS au-dessus de celle spécifiée dans la SRS.

Après l'apparition d'anomalies ayant réduit la tolérance aux anomalies du matériel à zéro, les exigences de 6.3.2 s'appliquent.

NOTE La durée moyenne de panne (voir VEI 191-13-08) prise en compte dans le modèle de fiabilité nécessite de tenir compte de l'intervalle des tests de diagnostic, de la durée de réparation et de tous les autres délais préalables au rétablissement.

6.3.2 Lorsqu'une(des) fonction(s) de diagnostic est(sont) nécessaire(s) pour réaliser la probabilité de défaillance dangereuse aléatoire du matériel requise et que le sous-système a une tolérance aux anomalies de zéro, alors la détection d'anomalie ainsi que la réaction à l'anomalie spécifiée doivent être exécutées avant que la situation dangereuse traitée par la SRCF puisse se produire.

EXCEPTION à 6.3.2: Dans le cas d'un sous-système mettant en oeuvre une SRCF particulière lorsque le rapport du taux de test de diagnostic au taux de sollicitation dépasse 100, alors l'intervalle des tests de diagnostic de ce sous-système doit permettre au sous-système de satisfaire aux exigences de la probabilité de défaillance dangereuse aléatoire.

6.3.3 Lorsque l'exécution d'une fonction réaction à l'anomalie, partie d'une SRCF spécifiée SIL3, a entraîné l'arrêt de la machine, le fonctionnement normal ultérieur de la machine via le SRECS (par exemple permettant le redémarrage de la machine) ne doit pas être possible tant que l'anomalie n'a pas été réparée ou corrigée. Pour les SRCF de niveau d'intégrité de sécurité spécifié inférieur à SIL3, le comportement de la machine après l'exécution d'une fonction réaction à l'anomalie (par exemple redémarrant le fonctionnement normal) doit dépendre de la spécification des fonctions réactions aux anomalies appropriées (voir 5.2.3).

6.4 Exigences pour l'intégrité de sécurité systématique des SRECS

NOTE Ces exigences s'appliquent au "niveau système" lorsque les sous-systèmes sont interconnectés pour réaliser un SRECS. Pour les exigences appropriées à la réalisation d'un sous-système, voir 6.7.8.

6.4.1 Exigences pour l'évitement des défaillances systématiques du matériel

6.4.1.1 Les mesures suivantes doivent s'appliquer:

a) le SRECS doit être conçu et réalisé selon le plan de sécurité fonctionnelle (voir 4.2);

b) choix, combinaison, arrangements, assemblage et montage corrects des sous-systèmes, y compris les câbles, les conducteurs et toutes les interconnexions;

c) utilisation des SRECS dans le cadre de la spécification du fabricant;

d) utilisation de notes d'applications du fabricant, par exemple feuillets catalogue, instructions de montage et emploi des règles de l'art (voir aussi ISO 13849-2, Annexe D.1);

e) utilisation de sous-systèmes ayant des caractéristiques de fonctionnement compatibles (voir aussi ISO 13849-2, Annexe D.1);

f) le SREC S doit être protégé selon la CEI 60204-1;

g) mesures pour empêcher la perte de la(des) connexion(s) de terre fonctionnelle(s) selon la CEI 60204-1;

h) les modes de fonctionnement non documentés de composants ne doivent pas être utilisés (par exemple les registres "réservés" d'un matériel programmable); et

i) prise en compte du mauvais usage prévisible, des changements ou modification(s) de l'environnement.

6.4.1.2 De plus, au moins une des techniques et/ou mesures suivantes doivent s'appliquer prenant en compte la complexité du SRECS et le(s) SIL des fonctions à réaliser par le SRECS:

a) la revue de conception du matériel du SRECS (par exemple par inspection ou lecture croisée); l'établissement de revues et/ou d'analyses de toutes les divergences entre la spécification et la réalisation;

NOTE 1 Afin de faire apparaître les divergences entre la spécification et la réalisation, tous les points de doute ou pouvant présenter une faiblesse concernant la réalisation, la mise en œuvre et l'utilisation du produit sont documentés de façon à ce que qu'ils puissent être résolus, prenant en compte que lors d'une procédure d'inspection, l'auteur est passif et l'inspecteur est actif tandis que lors d'une procédure de lecture croisée, l'auteur est actif et l'inspecteur est passif.

b) des outils d'aide tels que des ensembles de conception assistée par ordinateur capables de simulations ou d'analyses, et/ou l'utilisation d'outils de conception assistée par ordinateur afin d'exécuter des procédures de conception systématiques utilisant des éléments types déjà disponibles et testés.

NOTE 2 L'intégrité de ces outils peut être prouvée par des tests spécifiques, ou par un rapport détaillé sur une utilisation satisfaisante ou par une vérification indépendante de leurs résultats pour le SRECS particulier en cours de conception. Voir 6.11.3.4.

c) simulation: effectuer une assimilation complète et systématique de la conception d'un SRECS en termes de caractéristiques fonctionnelles, de dimensionnement et d'interaction corrects de ses sous-systèmes.

EXEMPLE La fonction du SRECS peut être simulée sur ordinateur par un modèle comportemental de logiciel (voir 6.11.3.4) dans lequel les sous-systèmes particuliers ou éléments de sous-systèmes ont chacun leur propre comportement simulé et la réponse du circuit dans lequel ils sont assemblés est étudiée en observant les spécifications aux limites de chaque sous-système ou élément de sous-système.

6.4.2 Exigences pour la maîtrise des anomalies systématiques

Les mesures suivantes doivent s'appliquer:

a) utilisation de l'absence de tension: le SRECS doit être conçu de façon qu'en cas de perte de son alimentation électrique, puisse être réalisé ou maintenu un état sûr de la machine.

b) mesures pour maîtriser l'effet des défaillances temporaires d'un sous-système: le SRECS doit être conçu par exemple de façon que:

   – les effets d'une variation de la tension (par exemple coupures, creux de tension) sur un sous-système particulier ou une partie d'un sous-système n'entraînent pas un phénomène dangereux (par exemple une interruption de tension qui affecte un circuit de moteur ne doit pas provoquer une mise en marche intempestive lorsque l'alimentation est rétablie); et

   NOTE 1 Voir aussi les exigences appropriées de la CEI 60204-1. En particulier:

   Il convient que la surtension ou la sous-tension soit détectée suffisamment tôt de façon que toutes les sorties puissent être commutées en position de sécurité par le programme de mise hors tension ou basculées sur une seconde unité d'alimentation; et/ou

   si nécessaire, il convient aussi que la surtension ou la sous-tension soit détectée suffisamment tôt pour que l'état interne puisse être sauvegardé dans une mémoire non volatile, de façon que toutes les sorties puissent être commutées en position de sécurité par le programme de mise hors tension ou basculées sur une seconde unité d'alimentation.

   – les effets des interférences électromagnétiques de l'environnement physique ou d'un(de) sous-système(s) n'entraînent pas un phénomène dangereux;

c) mesures pour maîtriser les effets des erreurs et autres effets liés à un quelconque processus de communication, y compris les erreurs de transmission, les répétitions, les suppressions, les insertions, les modifications du séquencement, la corruption, le retard et le masquage;

   NOTE 2 Des informations complémentaires peuvent être trouvées dans la CEI 60870-5-1, dans la EN 50159-1, la EN 50159-2 ainsi que dans la CEI 61508-2.

   NOTE 3 Le terme "masquage" signifie que le contenu exact d'un message n'est pas correctement identifié. Par exemple, un message provenant d'un composant qui n'est pas de sécurité est identifié incorrectement comme un message provenant d'un composant de sécurité.

d) lorsqu'une défaillance dangereuse se produit sur une interface, la fonction réaction à l'anomalie doit être exécutée avant que le phénomène dangereux dû à cette anomalie ne puisse se produire. Lorsqu'une anomalie réduisant la tolérance aux anomalies du matériel à zéro se produit, cette réaction à l'anomalie doit être exécutée avant que la MTTR estimée ne soit dépassée (voir le point g) de 6.7.4.4).

Les exigences du point d) s'appliquent aux interfaces qui sont les entrées et les sorties des sous-systèmes et toutes les autres parties de sous-systèmes qui comprennent du câblage ou nécessitent des opérations de câblage en phase d'intégration (par exemple les appareils de coupure des signaux de sortie d'une barrière immatérielle, d'une sortie d'un capteur de position de fin de course).

NOTE 4 Il n'est pas demandé qu'un sous-système ou élément de sous-système détecte de lui-même une anomalie au niveau de sa(ses) sortie(s). La fonction réaction à l'anomalie peut aussi être initiée par tout sous-système subséquent à l'issue de l'exécution d'un test de diagnostic.

6.4.3 Compatibilité électromagnétique (CEM) – Immunité

En complément des exigences de la CEI 61000-6-2 et des phénomènes EM donnés en Annexe E, les critères d'aptitude à la fonction suivants pour la sécurité fonctionnelle doivent être satisfaits par un SRECS:

– des conditions non sûres ou des phénomènes dangereux ne doivent pas être introduits; et

– pas de perte de la(des) SRCF; ou

– la(les) SRCF réalisée(s) par le SRECS peuvent être perturbées temporairement ou de façon permanente sous réserve que soit maintenu ou réalisé un état sûr de la machine avant qu'un phénomène dangereux ne puisse se produire. Lorsqu'un phénomène EM peut entraîner la destruction de composants, on doit garantir (par exemple par analyse) que la sécurité fonctionnelle n'est pas affectée (y compris par une(des) valeur(s) plus faible(s) de phénomènes EM pouvant provoquer une destruction partielle).

NOTE Il convient de prendre en considération le comportement du SRECS en réponse à un phénomène électromagnétique pour toute(s) valeur(s) jusqu'à celles données en Annexe E.

6.5 Choix du système de commande électrique relatif à la sécurité

Lorsqu'un fournisseur délivre un SRECS pour une fonction spécifique référencée dans la spécification des exigences de sécurité, on peut choisir un SRECS de conception type au lieu d'en concevoir un sur mesure sous réserve qu'il satisfasse aux exigences de la spécification des exigences de sécurité ainsi qu'à 6.3, 6.4 et 6.6.1.

NOTE Le choix d'un SRECS type est une alternative à la conception et au développement d'un SRECS spécifique conformément à 6.6.

6.6 Conception et développement d'un système de commande électrique relatif à la sécurité (SRECS)

6.6.1 Exigences générales

6.6.1.1 Le SRECS doit être conçu et développé selon la spécification des exigences de sécurité du SRECS (voir 5.2).

6.6.1.2 On doit suivre un processus de conception clairement structuré et documenté (voir 6.6.2).

6.6.1.3 Si l'utilisation de diagnostics est nécessaire pour réaliser l'intégrité de sécurité requise et en cas de détection d'anomalie, le SRECS doit exécuter la fonction réaction à l'anomalie spécifiée (voir 5.2 et 6.3).

6.6.1.4 Si un SRECS ou une partie d'un SRECS (c'est-à-dire un ou plusieurs de ses sous-systèmes) a à réaliser à la fois des SRCF et d'autres fonctions, alors tout leur matériel et leur logiciel doivent être considérés relatifs à la sécurité sauf s'il peut être prouvé que les SRCF et les autres fonctions sont suffisamment indépendantes dans leur réalisation (c'est-à-dire que le fonctionnement normal ou la défaillance de n'importe laquelle des autres fonctions n'affecte pas les SRCF).

NOTE L'indépendance suffisante dans la réalisation peut être prouvée en démontrant que la probabilité d'une défaillance dépendante entre des parties non relatives à la sécurité et des parties relatives à la sécurité est équivalente à celle du niveau d'intégrité de sécurité du SRECS.

6.6.1.5 Dans le cas d'un SRECS ou de ses sous-systèmes réalisant des fonctions de commande relatives à la sécurité de différents niveaux d'intégrité de sécurité, leur matériel et leur logiciel doivent être considérés comme nécessitant le niveau d'intégrité de sécurité le plus élevé sauf s'il peut être prouvé que les fonctions de commande relatives à la sécurité des différents niveaux d'intégrité de sécurité sont suffisamment indépendantes dans leur réalisation.

NOTE L'indépendance suffisante dans la réalisation peut être prouvée en démontrant que la probabilité d'une défaillance dépendante entre des parties réalisant des SRCF de différents niveaux d'intégrité de sécurité est équivalente à celle du niveau d'intégrité de sécurité réalisé par le SRECS.

6.6.1.6 Les interconnexions (c'est-à-dire les conducteurs, les câbles) autres que celles dédiées à la communication de données numériques doivent être considérées comme faisant partie d'un des sous-systèmes auquel elles sont raccordées (voir aussi 6.4.2 d).

6.6.1.7 Lorsque la communication de données numériques est employée comme partie de la réalisation d'un SRECS, elle doit satisfaire les exigences appropriées de la CEI 61508-2 selon le(s) niveau(x) SIL cible(s) de la(des) SRCF.

6.6.1.8 Les informations relatives à l'utilisation du SRECS doivent spécifier les techniques et mesures nécessaires pour maintenir le niveau d'intégrité de sécurité du SRECS pendant la durée de vie prévue à la conception.

6.6.2 Processus de conception et développement

La conception et le développement doivent suivre un processus clairement défini qui doit prendre en compte tous les aspects couverts par le processus indiqué en Figure 2.

NOTE L'approche de la présente norme est d'appliquer un processus de conception structuré au SRECS, commençant par les exigences qui sont spécifiées dans la spécification des exigences de sécurité. La Figure 3 montre le diagramme du processus de conception et la terminologie utilisée aux différents niveaux.

6.6.2.1 Conception de l'architecture système

6.6.2.1.1 Chaque SRCF, comme spécifié dans la spécification des exigences de sécurité du SRECS, doit être décomposée en une structure de blocs fonctionnels, par exemple comme indiqué à la Figure 3. Cette structure doit être complétée des informations comprenant:

– la description de la structure;

– les exigences de sécurité (fonctionnelles, d'intégrité) pour chaque bloc fonctionnel;

– les définitions des entrées et sorties de chaque bloc fonctionnel.

NOTE 1 Il convient que le processus de décomposition mène à une structure de blocs fonctionnels qui décrive complètement les exigences fonctionnelles et les exigences d'intégrité de la SRCF. Il convient que ce processus soit appliqué à ce niveau, ce qui permet aux exigences fonctionnelles et aux exigences d'intégrité déterminées pour chaque bloc fonctionnel d'être attribuées aux sous-systèmes, lorsque l'attribution de la totalité des exigences fonctionnelles à un sous-système d'un bloc fonctionnel est possible. Toutefois, il est possible d'attribuer plus d'un bloc fonctionnel à un sous-système unique, mais il n'est pas possible d'attribuer un bloc fonctionnel à plusieurs sous-systèmes lorsqu'il est prévu que ces sous-systèmes ont des exigences fonctionnelles et des exigences d'intégrité séparées. Lorsqu'il est prévu d'attribuer les exigences fonctionnelles d'un bloc fonctionnel à des éléments de sous-systèmes redondants, se reporter à 6.7.4.

NOTE 2 Les entrées et sorties de chaque bloc fonctionnel sont les informations qui y sont transférées, par exemple la vitesse, la position, le mode de fonctionnement, etc.

NOTE 3 Les blocs fonctionnels sont une représentation des fonctions de la SRCF (voir 3.2.16) et ne comprennent pas les fonctions de diagnostic du SRECS (voir 3.2.17). Pour les besoins de la présente norme, les fonctions de diagnostic sont considérées comme des fonctions séparées qui peuvent avoir une structure différente de celle de la SRCF (voir 6.8).

6.6.2.1.2 La première conception de l'architecture d'un SRECS doit être créée selon la structure des blocs fonctionnels.

NOTE Il devrait exister une collaboration permanente entre le développeur de l'architecture de commande relative à la sécurité, l'organisation responsable de la configuration des appareils et le développeur du logiciel. Au fur et à mesure que les exigences de sécurité du logiciel et l'architecture envisagée du logiciel se font plus précises, il peut y avoir un impact sur l'architecture matérielle du SRECS, et pour cette raison, une coopération étroite entre le concepteur de l'architecture du SRECS, le(s) fournisseur(s) de sous-systèmes, le développeur du logiciel et, si nécessaire, le concepteur de la machine ou l'utilisateur, peut aider à réduire la possibilité d'une(de) défaillance(s) systématique(s).

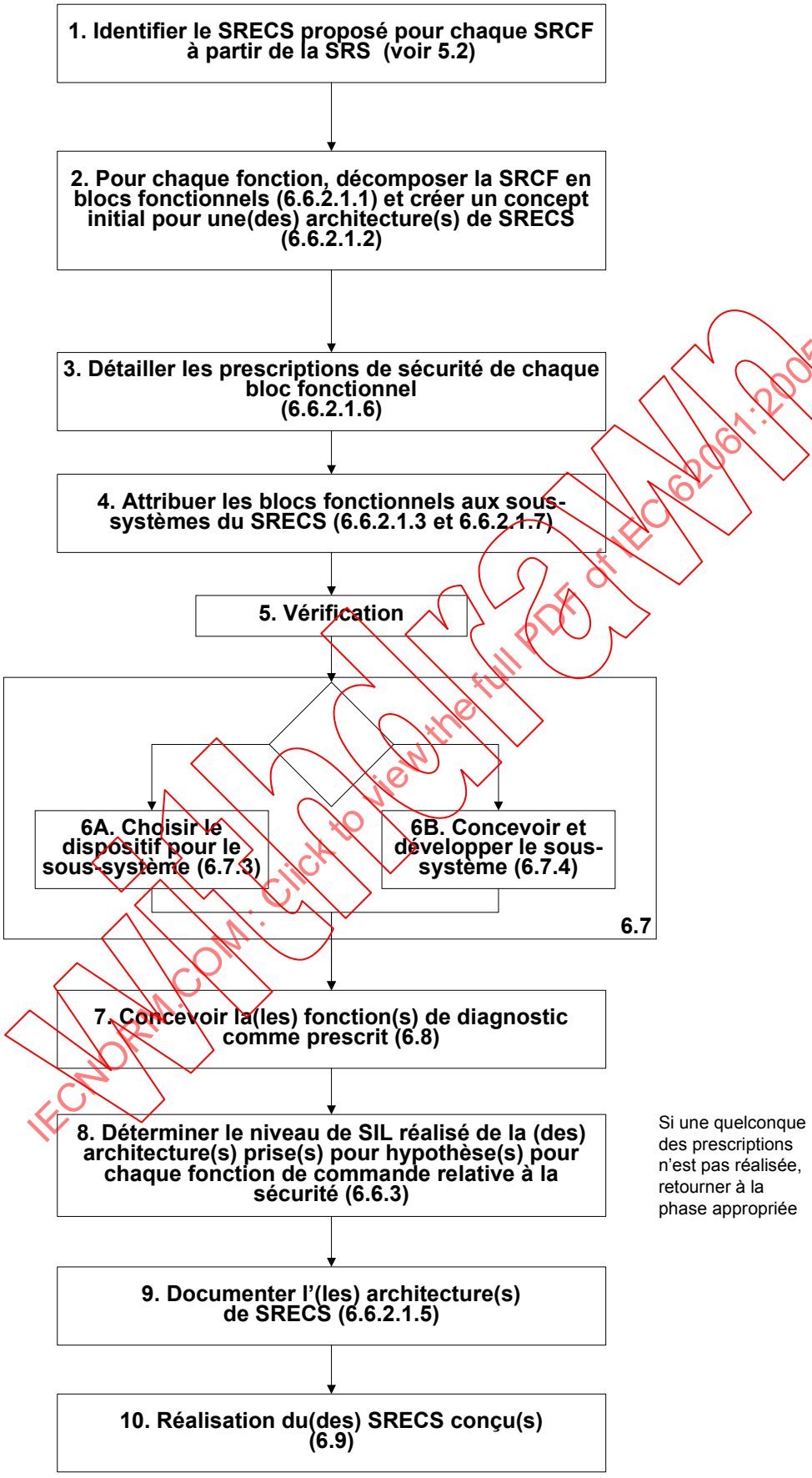


Figure 2 – Diagramme du processus de conception et de développement d'un SRECS

6.6.2.1.3 Chaque bloc fonctionnel doit être attribué à un sous-système de l'architecture du SRECS. Un ou plusieurs blocs fonctionnels peuvent être attribués à un sous-système.

6.6.2.1.4 Chaque sous-système et les blocs fonctionnels qui lui sont attribués doivent être clairement identifiés

6.6.2.1.5 L'architecture doit être documentée et décrire ses sous-systèmes et les relations qui les lient.

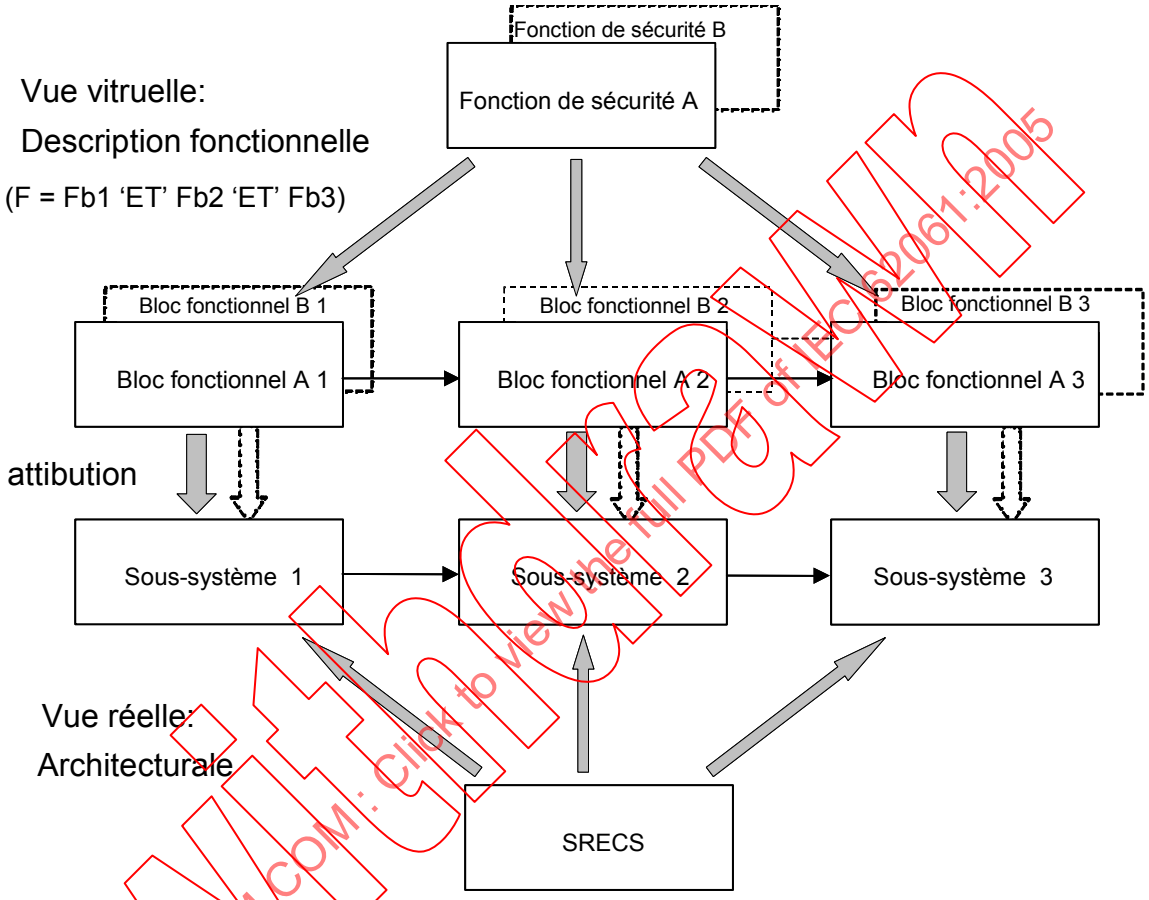


Figure 3 – Attribution des exigences de sécurité des blocs fonctionnels aux sous-systèmes (voir 6.6.2.1.1)

6.6.2.1.6 Les exigences de sécurité de chaque bloc fonctionnel doivent être comme spécifiées dans la spécification des exigences de sécurité de la SRCF correspondante en termes de:

– exigences fonctionnelles (par exemple les informations d'entrée, le fonctionnement interne (logique) et la sortie du bloc fonctionnel);

– les exigences d'intégrité de sécurité.

6.6.2.1.7 Les exigences de sécurité pour un sous-système doivent être celles du(des) bloc(s) fonctionnel(s) qui lui est(sont) attribué(s). Si plus d'un bloc fonctionnel est attribué à un sous-système, alors les exigences d'intégrité les plus élevées s'appliquent (voir 6.6.3). Ces exigences doivent être documentées au même titre que la spécification des exigences de sécurité du sous-système.

6.6.3 Exigences pour l'estimation de l'intégrité de sécurité réalisée par un SRECS

6.6.3.1 Généralités

Le niveau SIL pouvant être réalisé par le SRECS doit être considéré séparément pour chaque SRCF à exécuter par le SRECS.

Le niveau SIL pouvant être réalisé par le SRECS doit être déterminé à partir de la probabilité de défaillance dangereuse aléatoire du matériel, des contraintes architecturales et de l'intégrité de sécurité systématique des sous-systèmes que comprend le SRECS. Le niveau SIL réalisé est inférieur ou égal à la valeur la plus faible des SILCL de tous les sous-systèmes pour l'intégrité de sécurité systématique et les contraintes architecturales données.

6.6.3.2 Intégrité de sécurité du matériel

6.6.3.2.1 La probabilité de défaillance dangereuse pour chaque SRCF due aux défaillances dangereuses aléatoires du matériel doit être égale ou inférieure à la valeur cible des défaillances telle que spécifiée dans la spécification des exigences de sécurité.

NOTE Les valeurs cibles associées aux SIL sont indiquées dans le Tableau 3.

6.6.3.2.2 La probabilité de défaillance dangereuse pour chaque SRCF, due aux défaillances dangereuses aléatoires du matériel doit être estimée en prenant en compte:

a) l'architecture du SRECS au regard de chaque SRCF à l'étude;

NOTE Ceci implique de décider lesquels des modes de défaillances des sous-systèmes sont dans une configuration série (c'est-à-dire qu'une quelconque défaillance entraîne la défaillance de la SRCF à effectuer appropriée) et lesquels sont dans une configuration parallèle (redondante) (c'est-à-dire que des défaillances coïncidentes sont nécessaires pour entraîner la défaillance de la SRCF correspondante).

b) le taux de défaillance estimé de chaque sous-système lors de l'exécution de ses blocs fonctionnels attribués dans tous les modes et qui entraîneraient une défaillance dangereuse du SRECS.

6.6.3.2.3 L'estimation de la probabilité de défaillance dangereuse doit être basée sur la probabilité de défaillance dangereuse aléatoire du matériel de chaque sous-système correspondant et déduite des informations requises en 6.7.2.2 y compris, si applicable 6.7.2.2 (k), pour les processus de communication de données numériques entre les sous-systèmes. La probabilité de défaillance dangereuse aléatoire du matériel du SRECS est la somme des probabilités de défaillance dangereuse aléatoire de tous les sous-systèmes impliqués dans le fonctionnement de la SRCF et doit comprendre, si applicable, la probabilité d'erreur de transmission dangereuse dans le cas de processus de communication de données numériques:

$$PFH_{D} = PFH_{D1} + \ldots + PFH_{Dn} + P_{TE}$$

NOTE 1 Cette approche est basée sur la définition d'un bloc fonctionnel qui établit que la défaillance de tout bloc fonctionnel entraînera la défaillance de la SRCF (voir 3.2.16).

NOTE 2 Les interconnexions autres que celles dédiées à la communication de données numériques sont considérées comme étant une partie des sous-systèmes.

6.6.3.3 Contraintes architecturales

Le niveau SIL réalisé par le SRECS selon les contraintes architecturales est inférieur ou égal au SILCL le plus faible de tout sous-système (voir 6.7.6) participant au fonctionnement du SRECS.

NOTE Par exemple, lorsqu'un SRECS comprend deux sous-systèmes raccordés en série (sous-système 1 et sous-système 2), la SFF et la tolérance aux anomalies de chaque sous-système sont considérées être comme indiqué au Tableau 4. La PFH_{D} estimée pour un SRECS est de 8×10^{-8}, ce qui correspond au niveau SIL 3. Toutefois, selon le Tableau 5, la contrainte architecturale du sous-système 2 limite le niveau SIL pouvant être réalisé par le SRECS au niveau SIL 2.

Tableau 4 – Caractéristiques des sous-systèmes 1 et 2 utilisés dans cet exemple (voir Note ci-dessus)

Sous-système	Tolérance aux anomalies du matériel	SFF	Limite de revendication de SIL selon les contraintes architecturales (voir Tableau 5)
1	1	95 %	SIL3
2	1	80 %	SIL2

6.6.3.4 Intégrité de sécurité systématique

Le niveau SIL réalisé par le SRECS est inférieur ou égal au SILCL le plus faible d'un quelconque sous-système participant au fonctionnement de la SRCF.

NOTE Les mesures décrites en 6.7.9 donnent un SILCL jusqu'à SIL 3 pour l'intégrité de sécurité systématique d'un sous-système conçu selon 6.7.4.

6.7 Réalisation des sous-systèmes

6.7.1 Objectifs

L'objectif est de réaliser un sous-système qui satisfasse à toutes les exigences de sécurité des blocs fonctionnels attribués (voir Figure 3). Deux approches sont considérées.

– le choix d'un appareil suffisant pour satisfaire aux exigences de ce sous-système, c'est-à-dire devant satisfaire à la spécification des exigences de sécurité de chacun des blocs fonctionnels qui lui sont attribués ainsi qu'aux exigences de la présente norme; ou
– la conception et le développement d'un sous-système en combinant les éléments de blocs fonctionnels et en spécifiant la façon de les arranger et dont ils interagissent.

6.7.2 Exigences générales pour la réalisation d'un sous-système

6.7.2.1 Le sous-système doit être réalisé soit par le choix (voir 6.7.3) soit par la conception (voir 6.7.4) conformément à sa spécification des exigences de sécurité (voir 6.6.2.1.7), en prenant en compte toutes les exigences de 6.2. Le(s) sous-système(s) incorporant des composants complexes doivent se conformer à la CEI 61508-2 et à la CEI 61508-3 pour autant qu'applicables, pour le niveau SIL prescrit.

EXCEPTION : Lorsqu'un sous-système comprend un composant complexe en tant qu'élément de sous-système, 6.7.4.2.3 s'applique.

6.7.2.2 Les informations suivantes doivent être disponibles pour chaque sous-système:

a) la spécification fonctionnelle des fonctions et des interfaces du sous-système pouvant être utilisées par les SRCF;

b) le taux de défaillance estimé (dû aux défaillances aléatoires du matériel) déclaré dans tous les modes susceptibles d'entraîner une défaillance dangereuse du SRECS;

NOTE 1 Pour les sous-systèmes électromécaniques, il convient que la probabilité de défaillance soit estimée en prenant en compte le nombre de cycles de manœuvres déclaré par le fabricant et le cycle de fonctionnement (voir 5.2.3). Il convient que la présente information soit basée sur une valeur B10 (c'est-à-dire le temps attendu pour lequel 10 % de la population aura eu une défaillance). Voir aussi la CEI 61810-2[1].

c) les contraintes sur le sous-système liées

– à l'environnement et aux conditions de fonctionnement qu'il convient d'observer afin de maintenir la validité des taux de défaillance estimés dus aux défaillances aléatoires du matériel; et

[1] A publier.

– à la durée de vie du sous-système qu'il convient de ne pas dépasser afin de maintenir la validité des taux de défaillance estimés dus à des défaillances aléatoires du matériel;

d) les tests et/ou les exigences de maintenance;

e) la couverture de diagnostic et l'intervalle des tests de diagnostic (lorsque prescrits, voir la note 2);

NOTE 2 Le point e) se rapporte aux fonctions de diagnostic qui sont externes au sous-système. Les informations correspondantes sont prescrites uniquement lorsque dans le modèle de fiabilité du SRECS, des fonctions de diagnostic sont annoncées comme étant exécutées dans le sous-système considéré.

f) les informations supplémentaires (par exemple, les durées de réparation) qui sont nécessaires pour déduire une durée moyenne de panne (MTTR) à la suite de la détection d'une anomalie par les diagnostics;

NOTE 3 Les points b) à f) sont nécessaires pour permettre l'estimation de la probabilité de défaillance par heure de la SRCF.

g) le SILCL dû aux contraintes architecturales (voir 6.7.6) ou:

– les informations nécessaires pour permettre de déduire la proportion de défaillances en sécurité (SFF) du sous-système telle qu'appliquée au SRECS; et

NOTE 4 Parmi les informations nécessaires figurent les modes de défaillance possibles du sous-système. Sur la base des modes de défaillance du sous-système, il peut être décidé que la défaillance du sous-système entraîne soit une défaillance en sécurité soit une défaillance dangereuse du SRECS.

NOTE 5 Pour des informations complémentaires sur l'estimation de la SFF, voir 6.7.7.

– la tolérance aux anomalies du matériel pour le sous-système;

h) les limitations concernant l'application du sous-système qu'il convient d'observer afin d'éviter les défaillances systématiques;

i) le niveau d'intégrité de sécurité le plus élevé qui peut être revendiqué pour une SRCF qui utilise le sous-système, sur la base:

– de mesures et techniques utilisées pour empêcher l'introduction d'anomalies systématiques lors de la conception et de la réalisation du matériel et du logiciel du sous-système;

– des caractéristiques de conception qui rendent le sous-système tolérant aux anomalies systématiques.

NOTE 6 Les points h) et i) sont nécessaires pour déterminer le niveau d'intégrité de sécurité le plus élevé qui peut être revendiqué pour une SRCF, en prenant en compte les contraintes architecturales. Aussi, ces points peuvent être utilisés pour fournir un lien (voir Tableaux 4 et 5) avec les exigences de catégorie de l'ISO 13849-1 à la fois en termes de détection d'anomalie et de tolérance aux anomalies du matériel.

j) les informations nécessaires à l'identification de la configuration du matériel et du logiciel du sous-système, afin de permettre la gestion de configuration d'un SRECS, conformément à 6.11.3.2;

k) la probabilité d'erreurs de transmission dangereuses dans le cas de processus de communication de données numériques, si applicable.

6.7.3 Exigences pour le choix de sous-systèmes (types) existants

6.7.3.1 Lorsqu'un fournisseur délivre un sous-système pour une SRCF particulière référencée dans la spécification des exigences de sécurité, on peut choisir un sous-système type au lieu de concevoir un sous-système particulier sous réserve qu'il satisfasse à la spécification des exigences de sécurité du sous-système et à 6.4.3 et 6.7.3.2 ou 6.7.3.3.

6.7.3.2 Le(s) sous-système(s) incorporant des composants complexes doivent se conformer à la CEI 61508-2 et à la CEI 61508-3 pour autant qu'applicables, pour le niveau SIL prescrit.

EXCEPTION : Lorsqu'une conception de sous-système comprend un composant complexe en tant qu'élément de sous-système, 6.7.4.2.3 s'applique.

6.7.3.3 Les sous-systèmes comprenant uniquement des composants de faible complexité doivent se conformer à 6.7.4.4, 6.7.6.2, 6.7.6.3, 6.7.7, 6.7.8 et 6.8 de la présente norme.

6.7.4 Conception et développement des sous-systèmes

6.7.4.1 Objectifs

6.7.4.1.1 Le premier objectif est de concevoir un sous-système qui satisfasse aux exigences de sécurité du(des) bloc(s) fonctionnel(s) attribué(s).

6.7.4.1.2 Le second objectif est de créer une architecture en termes d'éléments de sous-système qui travaillent ensemble en combinaison pour satisfaire aux exigences fonctionnelles et aux exigences d'intégrité de sécurité de tous les blocs fonctionnels attribués au sous-système.

6.7.4.2 Exigences générales

6.7.4.2.1 Le sous-système doit être conçu en prenant en compte sa spécification des exigences de sécurité.

6.7.4.2.2 Le sous-système doit être tel qu'il satisfasse à toutes les exigences de a) à c) qui suivent:

a) les exigences pour l'intégrité de sécurité du matériel comprenant:
   - les contraintes architecturales sur l'intégrité de sécurité du matériel (voir 6.7.6);
   - et les exigences pour la probabilité de défaillance dangereuse aléatoire du matériel (voir 6.7.8);

b) les exigences pour l'intégrité de sécurité systématique comprenant:
   - les exigences pour l'évitement des défaillances (voir 6.7.9.1), et les exigences pour la maîtrise des anomalies systématiques (voir 6.7.9.2), ou
   - la preuve que le matériel est "validé en utilisation". Dans ce cas, le sous-système doit satisfaire aux exigences appropriées de la CEI 61508-2 (voir la CEI 61508-2, 7.4.7.5 à 7.4.7.12).

c) les exigences relatives au comportement du sous-système sur détection d'anomalie (voir 6.3).

6.7.4.2.3 Lorsque la conception d'un sous-système incorpore un composant complexe (en tant qu'élément de sous-système) qui satisfait à toutes les exigences appropriées de la CEI 61508-2 et de la CEI 61508-3 en prenant en compte le SILCL, ce composant peut être considéré comme un composant de faible complexité dans le contexte de la conception d'un sous-système dès lors que ses modes de défaillance, comportement sur détection d'anomalie, taux de défaillance et autres informations relatives à la sécurité sont connues. De tels composants doivent être utilisés uniquement selon leur spécification et les informations appropriées relatives à leur utilisation, données par leur fournisseur.

6.7.4.3 Processus de conception et développement d'un sous-système

La conception et le développement d'un sous-système doivent suivre un processus clairement défini qui doit prendre en compte tous les aspects couverts par le processus indiqué en Figure 4.

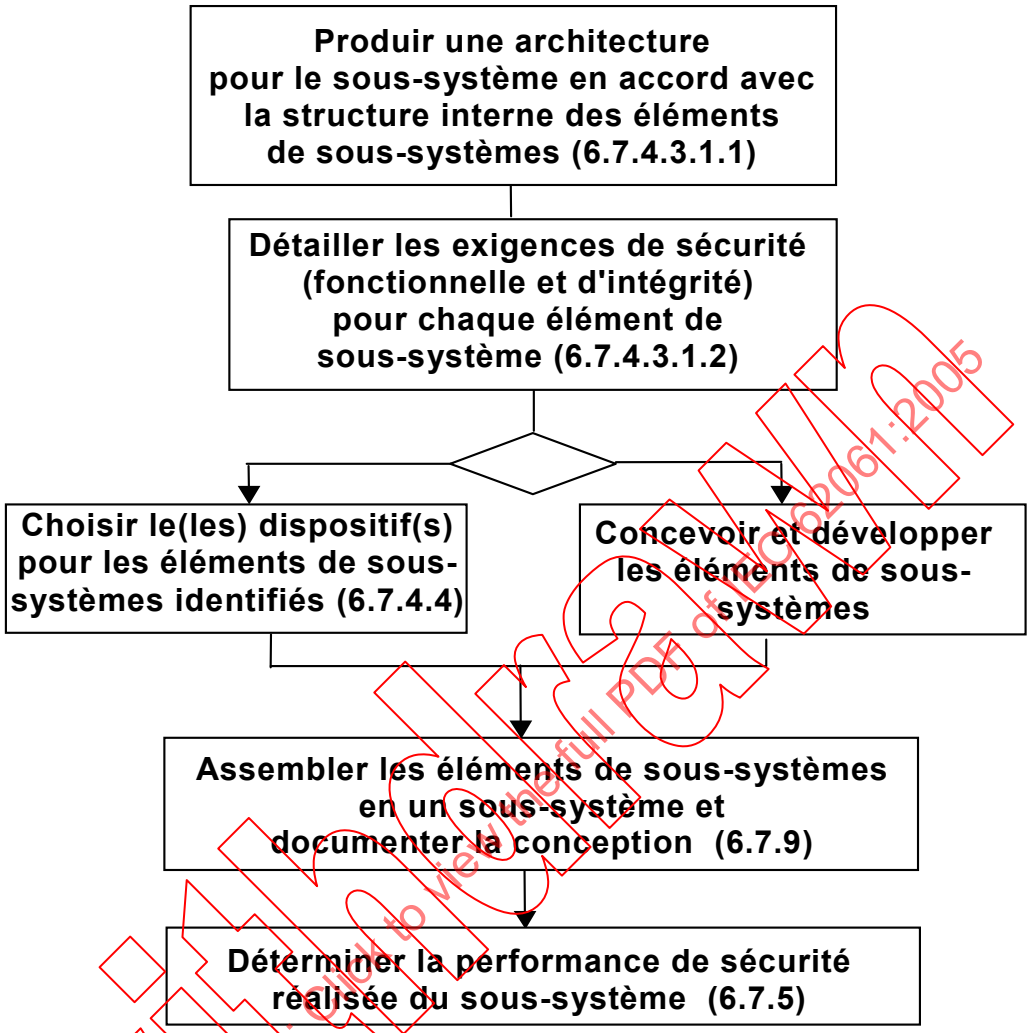


Figure 4 – Diagramme de conception et développement d'un sous-système (voir case 6B de la Figure 2)

6.7.4.3.1 Conception de l'architecture d'un sous-système

6.7.4.3.1.1 Pendant la conception de l'architecture d'un sous-système, il convient que le processus de décomposition mène à une structure d'éléments de blocs fonctionnels représentant la totalité des exigences fonctionnelles du bloc fonctionnel. Il convient que ce processus soit appliqué à ce niveau, ce qui permet aux exigences fonctionnelles déterminées pour chaque bloc fonctionnel d'être attribuées aux éléments de sous-systèmes (voir Figure 5).

NOTE Le diagramme du processus de conception est indiqué en Figure 4.

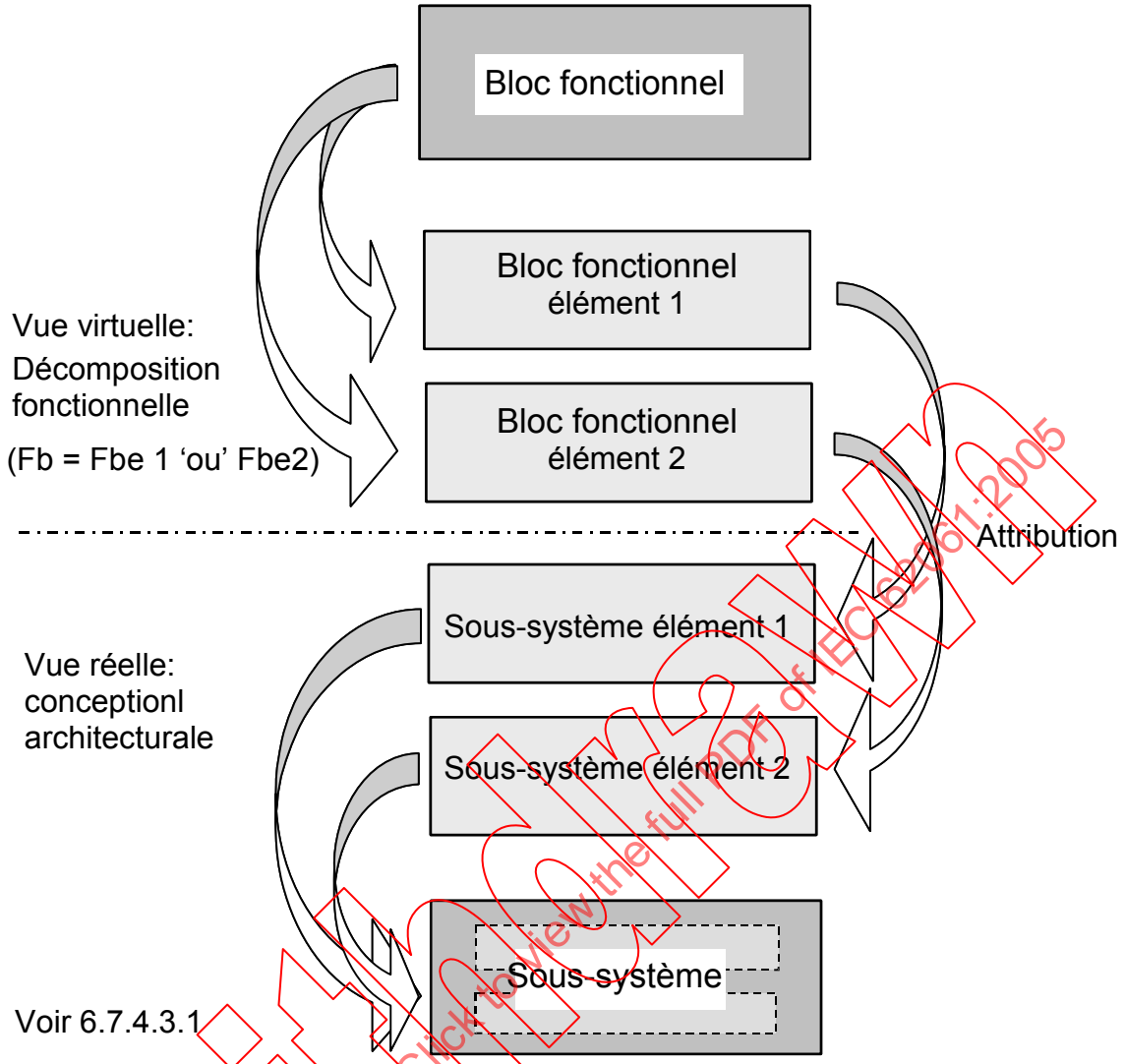


Figure 5 – Décomposition d'un bloc fonctionnel en éléments de blocs fonctionnels redondants et leurs éléments de sous-systèmes associés

6.7.4.3.1.2 L'architecture du sous-système doit être documentée en termes d'éléments qui la composent et des relations qui les lient. Lorsque c'est nécessaire, elle doit aussi comprendre les informations relatives aux éléments de blocs fonctionnels qui sont attribués aux éléments de sous-systèmes.

6.7.4.4 Exigences pour le choix et la conception des éléments de sous-systèmes

6.7.4.4.1 Les éléments d'un sous-système doivent convenir à l'usage auquel ils sont destinés et doivent être conformes aux normes internationales appropriées lorsqu'elles existent.

6.7.4.4.2 Les informations suivantes doivent être disponibles pour chaque élément de sous-système:

a) une spécification fonctionnelle de l'élément de sous-système;

b) la spécification des interfaces de l'élément de sous-système (par exemple les caractéristiques électriques);

c) chaque mode de défaillance et sa probabilité d'occurrence, et, si approprié (par exemple pour des composants complexes utilisés selon 6.7.4.2.3), la couverture de diagnostic et la probabilité de défaillance dangereuse.

NOTE 1 Pour les sous-systèmes électromécaniques, il convient que la probabilité de défaillance soit estimée en prenant en compte le nombre de cycles de manœuvres déclaré par le fabricant et le cycle de fonctionnement (voir 5.2.3). Il convient que la présente information soit basée sur une valeur B10 (c'est-à-dire le temps attendu pour lequel 10 % de la population aura eu une défaillance). Voir aussi la CEI 61810-2[2].

d) les contraintes sur l'élément de sous-système liées:
   – à l'environnement et aux conditions de fonctionnement qu'il convient d'observer afin de maintenir la validité des informations données en c); et
   – à la durée de vie de l'élément de sous-système qu'il convient de ne pas dépasser afin de maintenir la validité des informations données en c);

e) les tests et/ou les exigences de maintenance;

f) les caractéristiques qui peuvent contribuer aux diagnostics (par exemple des contacts mécaniquement liés);

g) les informations supplémentaires (par exemple, les durées de réparation) qui sont nécessaires pour déduire une durée moyenne de panne (MTTR) à la suite de la détection d'une anomalie par les diagnostics;

h) les limitations concernant l'application de l'élément de sous-système qu'il convient d'observer afin d'éviter les défaillances systématiques;

i) la tolérance aux anomalies du matériel.

6.7.5 Détermination du fonctionnement de la sécurité d'un sous-système

Le fonctionnement de la sécurité d'un sous-système est caractérisé par le SILCL déterminé par ses contraintes architecturales (6.7.6), son SILCL dû à l'intégrité systématique (6.7.9) et sa probabilité de défaillance dangereuse aléatoire du matériel (6.7.8).

NOTE 1 Le SILCL d'un sous-système établit une limite pour le niveau d'intégrité de sécurité maximal qui peut être revendiqué pour une fonction de commande relative à la sécurité utilisant ce sous-système.

NOTE 2 Les informations concernant l'ensemble des trois aspects sont nécessaires pour déterminer le niveau SIL atteint par le système de commande relatif à la sécurité réalisant la SRCF attribuée.

6.7.6 Contraintes architecturales sur l'intégrité de sécurité du matériel des sous-systèmes

6.7.6.1 Dans le contexte de l'intégrité de sécurité du matériel, le niveau d'intégrité de sécurité le plus élevé pouvant être revendiqué pour une SRCF est limité par les tolérances aux anomalies du matériel et les proportions de défaillances en sécurité des sous-systèmes qui réalisent cette SRCF. Le Tableau 5 spécifie le niveau d'intégrité de sécurité le plus élevé pouvant être revendiqué pour une SRCF qui utilise un sous-système prenant en compte la tolérance aux anomalies du matériel et la proportion de défaillances en sécurité de ce sous-système. Les contraintes architecturales données dans le Tableau 5 doivent s'appliquer à chaque sous-système. Prenant en compte ces contraintes architecturales:

a) une tolérance aux anomalies du matériel N signifie que N + 1 anomalies sont susceptibles de provoquer la perte de la SRCF. Lors de la détermination de la tolérance aux anomalies du matériel, aucune autre mesure pouvant maîtriser l'effet des anomalies, telles que les diagnostics, n'est prise en compte; et

b) lorsqu'une anomalie donne directement lieu à l'apparition d'une ou de plusieurs anomalies subséquentes, celles-ci doivent être considérées comme une anomalie unique;

c) lors de la détermination de la tolérance aux anomalies du matériel, certaines anomalies peuvent être exclues pourvu que leur probabilité d'occurrence soit très faible par rapport aux exigences d'intégrité de sécurité du sous-système. De telles exclusions d'anomalies doivent être justifiées et documentées (voir aussi 6.7.7).

---

[2] A publier.

6.7.6.2 Les contraintes architecturales du Tableau 5 doivent s'appliquer à chaque sous-système réalisant un bloc fonctionnel d'une SRCF.

6.7.6.3 Un sous-système comprenant uniquement un seul élément de sous-système doit satisfaire aux exigences du Tableau 5. En particulier, pour un tel sous-système ayant une tolérance aux anomalies du matériel de zéro (c'est-à-dire N = 0), une SFF supérieure à 99 % doit alors être réalisée par la(les) fonction(s) de diagnostic d'un SRECS.

NOTE La présente exigence est nécessaire afin de s'assurer qu'une forme appropriée des contraintes architecturales est appliquée aux sous-systèmes composés uniquement d'un seul élément de sous-système en vue de justifier d'un SILCL de SIL3.

Tableau 5 – Contraintes architecturales sur les sous-systèmes: SIL maximal pouvant être revendiqué pour une SRCF utilisant ce sous-système

Proportion de défaillances en sécurité	Tolérance aux anomalies du matériel (voir Note 1)		
	0	1	2
< 60 %	Non autorisé (pour les exceptions, voir Note 3)	SIL1	SIL2
60 % - < 90 %	SIL1	SIL2	SIL3
90 % - < 99 %	SIL2	SIL3	SIL3 (voir Note 2)
≥ 99 %	SIL3	SIL3 (voir Note 2)	SIL3 (voir Note 2)

NOTE 1 Une tolérance aux anomalies du matériel N signifie que N + 1 anomalies sont susceptibles de provoquer la perte de la fonction de commande relative à la sécurité.

NOTE 2 Une limite de revendication SIL 4 n'est pas prise en compte dans la présente norme. Pour le SIL4, voir la CEI 61508-1.

NOTE 3 Voir 6.7.6.4 ou pour les sous-systèmes dans le cas où des exclusions d'anomalies ont été appliquées à des défauts pouvant conduire à une défaillance dangereuse, voir 6.7.7.

6.7.6.4 Pour les sous-sytèmes électromécaniques qui disposent d'une proportion de défaillances en sécurité de moins de 60 % et une tolérance aux anomalies du matériel de zéro, et qui utilisent des composants correctement essayés (voir Note) selon l'ISO 13849-1:2006, un PLC de Catégorie 1 doit être considéré pour réaliser une SILCL de SIL1.

NOTE Un composant correctement essayé pour une application liée à la sécurité est un composant qui a été:

a) largement utilisé dans le passé avec des résultats satisfaisants dans des application similaires, ou

b) réalisé et vérifié selon des principes qui démontrent son aptitude et sa fiabilité pour des applications liées à la sécurité.

6.7.6.5 Lorsqu'un sous-système est conçu selon l'ISO 13849-1:1999 et validé selon l'ISO 13849-2:2003, la relation suivante dans le contexte des contraintes architecturales seules, peut s'appliquer selon le Tableau 6. Il est admis qu'un sous-système d'une catégorie particulière conforme à l'ISO 13849-1:1999 a la tolérance aux anomalies du matériel et la proportion de défaillances en sécurité associées telles qu'indiquées au Tableau 6.

NOTE Pour accomplir le niveau SIL prescrit, il est aussi nécessaire de satisfaire aux exigences tenant compte de la probabilité de défaillance dangereuse et de l'intégrité de sécurité systématique.

Tableau 6 – Contraintes architecturales: SILCL en relation avec les catégories

Catégorie	Tolérance aux anomalies du matériel	SFF	Limite de revendication SIL maximale selon les contraintes architecturales
	Il est admis que les sous-systèmes de catégorie spécifiée ont les caractéristiques indiquées ci-dessous		
1	0	< 60 %	Voir Note 1
2	0	60 % - 90 %	SIL1 (voir Note 2)
3	1	< 60 %	SIL1
	1	60 % - 90 %	SIL2
4	>1	60 % - 90 %	SIL3 (voir Note 3)
	1	> 90 %	SIL3 (voir Note 4)

NOTE 1 Les sous-systèmes ayant une SFF < 60 % mais qui sont conçus selon la Catégorie 1 de l'ISO 13849-1:1999 et validés selon l'ISO 13849-2:2003 sont admis comme réalisant un SILCL de SIL1.

NOTE 2 Le cas de la catégorie 2 avec une SFF > 90 % est admis comme n'étant pas réalisé par les exigences de conception de l'ISO 13849-1:1999.

NOTE 3 La couverture de diagnostic est admise être inférieure à 90 % pour les sous-systèmes de catégorie 4 lorsque le cas où elle est plus grande que la tolérance aux anomalies du matériel seule (c'est-à-dire les fautes accumulées) est considéré.

NOTE 4 La catégorie 4 requiert une SFF supérieure à 90 % mais inférieure à 99 % lorsque n'est considérée que la tolérance aux anomalies du matériel.

NOTE 5 La catégorie B selon l'ISO 13849-1:1999 n'est pas considérée suffisante pour réaliser le niveau SIL1.

6.7.7 Estimation de la proportion de défaillances en sécurité (SFF)

6.7.7.1 On doit estimer la SFF lorsqu'il est prescrit de déterminer le SILCL consécutif aux contraintes architecturales.

6.7.7.2 Pour estimer la SFF, une analyse (par exemple analyse par arbre de panne, analyse des modes de défaillance et de leurs effets) de chaque sous-système doit être réalisée afin de déterminer les anomalies appropriées et leurs modes de défaillance correspondants. Qu'une défaillance soit une défaillance en sécurité ou une défaillance dangereuse dépend du SRECS et des fonctions de commande relatives à la sécurité prévues, y compris la fonction réaction à l'anomalie. La probabilité de chaque mode de défaillance doit être déterminée sur la base de la probabilité des anomalies associées en prenant en compte l'usage prévu, et peut être déduite de sources telles que:

a) les données de taux de défaillance suffisamment fiables issues du retour d'expérience du fabricant et appropriées à l'usage prévu;
b) les données de défaillance de composants provenant d'une origine industrielle reconnue (voir références en Annexe D) et appropriées à l'usage prévu;
c) les données des modes de défaillance indiquées en Annexe D;
d) les données de taux de défaillance déduits des résultats des tests et des analyses.

EXCEPTION: Pour un sous-système ayant une tolérance aux anomalies du matériel de zéro et lorsque les exclusions d'anomalies ont été appliquées aux anomalies qui pourraient entraîner une défaillance dangereuse, alors le SILCL consécutif aux contraintes architecturales de ce sous-système est limité au maximum à SIL2.

6.7.7.3 L'usage des exclusions d'anomalies doit être justifié (par exemple par analyse) et documenté.

NOTE Il est admis d'exclure les anomalies selon 3.3 et le Tableau D.5 de l'ISO 13849-2:2003.

6.7.8 Exigences pour la probabilité de défaillance dangereuse aléatoire du matériel des sous-systèmes

6.7.8.1 Exigences générales

6.7.8.1.1 La probabilité de défaillance dangereuse aléatoire du matériel doit être égale ou inférieure à la valeur cible des défaillances comme spécifié dans la spécification des exigences de sécurité du sous-système (voir 6.6.2.1.7).

6.7.8.1.2 La probabilité de défaillance dangereuse pour chaque sous-système, consécutive aux défaillances aléatoires du matériel, pour réaliser les blocs fonctionnels attribués doit être estimée en prenant en compte:

a) l'architecture du sous-système dès lors qu'elle traite des blocs fonctionnels attribués en considération;

NOTE 1 Ceci implique de décider s'il y a une tolérance aux anomalies du matériel ou non.

b) le taux de défaillance de chaque élément de sous-système dans tous les modes qui pourraient entraîner une défaillance dangereuse du sous-système mais qui sont détectés par les tests de diagnostic (voir 6.3);

c) le taux de défaillance de chaque élément de sous-système dans tous les modes qui pourraient entraîner une défaillance dangereuse du sous-système mais qui ne sont pas détectés par les tests de diagnostic (voir 6.3);

d) la sensibilité du sous-système aux défaillances de cause commune qui pourraient entraîner une défaillance dangereuse du sous-système (voir Notes 2 et 3);

NOTE 2 Lorsque la comparaison de composants redondants est utilisée pour la détection d'anomalie, une défaillance des moyens de détection d'anomalie peut se produire lorsque les composants redondants subissent une anomalie au même moment et dans le même mode. Elle peut se produire à partir d'une cause commune et est appelée défaillance de cause commune (CCF) qui s'exprime en facteur bêta (ß).

Une approche simplifiée pour estimer la sensibilité aux défaillances de cause commune est donnée en 6.7.8.3. Pour de plus amples conseils sur la quantification de l'effet des défaillances de cause commune relatives au matériel, voir aussi la CEI 61508-6, Annexe D.

e) la couverture de diagnostic des tests de diagnostic (voir 3.2.38) et l'intervalle des tests de diagnostic associé;

f) les intervalles auxquels les tests périodiques sont réalisés pour révéler les anomalies dangereuses qui ne sont pas détectées par les tests de diagnostic et/ou la durée de mission du(des) élément(s) de sous-système qu'il convient de ne pas dépasser afin de maintenir la validité des informations données en b) et c);

g) les durées de réparation pour les anomalies détectées lorsque le sous-système est conçu pour la réparation en ligne.

NOTE 3 La durée de réparation maximale constituera une partie de la durée de panne (voir VEI 191-10-05), y compris aussi la durée consacrée à la détection d'une anomalie et toute durée pendant laquelle la réparation n'est pas possible (voir la CEI 61508-6, Annexe B, pour un exemple de la façon dont la durée moyenne de panne peut être utilisée pour le calcul de la probabilité de défaillance). Pour les situations dans lesquelles la réparation ne peut être effectuée que dans un intervalle de temps spécifique, pendant que la machine est mise à l'arrêt et dans un état sûr, il est particulièrement important que l'intervalle de temps pendant lequel aucune réparation ne peut être effectuée soit totalement pris en compte, spécialement lorsque celui-ci est particulièrement long.

NOTE 4 Une approche simplifiée pour l'estimation de la probabilité de défaillance dangereuse aléatoire du matériel des sous-systèmes est donnée en 6.7.8.2. D'autres méthodes existent et le choix de la méthode la plus appropriée dépendra des circonstances. Les autres méthodes comprennent:

a) analyse par arbre de panne (voir B.6.6.5 de la CEI 61508-7 et la CEI 61025);

b) les modèles de Markov (voir C.6.4 de la CEI 61508-7 et la CEI 61165-13);

c) les diagrammes de fiabilité (voir C.6.5 de la CEI 61508-7).

NOTE 5 Les défaillances consécutives à des effets de cause commune et des processus de communication de données peuvent être le résultat d'effets autres que ceux des défaillances réelles de composants du matériel (par exemple les interférences électromagnétiques, les erreurs logicielles, etc.). Voir 6.7.9.

6.7.8.1.3 Pour les sous-systèmes ou éléments de sous-systèmes dont la probabilité de défaillance est donnée en relation avec un nombre de cycles de manœuvre, ces valeurs doivent être transformées en valeurs fonction du temps en utilisant le cycle de fonctionnement pour les SRCF appropriées (voir 5.2.3).

6.7.8.1.4 L'intervalle des tests de diagnostic de tout sous-système ayant une tolérance aux anomalies du matériel supérieure à zéro doit être tel qu'il permette au sous-système de satisfaire à l'exigence concernant la probabilité de défaillance aléatoire du matériel (voir 6.3.1).

NOTE Il convient que cet intervalle des tests de diagnostic soit d'une durée telle qu'une anomalie soit détectée avant l'apparition d'une anomalie subséquente pouvant entraîner une défaillance dangereuse du sous-système et qui dépasse la mesure de la défaillance cible.

6.7.8.1.5 L'intervalle des tests de diagnostic de tout sous-système ayant une tolérance aux anomalies du matériel de zéro doit être tel que les exigences de 6.3.2 soient satisfaites.

6.7.8.1.6 Lorsqu'un sous-système de faible complexité est conçu selon l'ISO 13849-1:1999 et validé selon l'ISO 13849-2:2003, et satisfait aussi aux exigences pour les contraintes architecturales (voir 6.7.6) et l'intégrité de sécurité systématique (voir 6.7.9), les valeurs seuil de probabilité de défaillance dangereuse (PFH_D) données dans le Tableau 7 peuvent être utilisées pour estimer l'intégrité de sécurité du matériel (voir 6.6.3.2).

Tableau 7 – Probabilité de défaillance dangereuse

Catégorie	**Tolérance aux anomalies du matériel**	**DC**	**Valeur seuil PFH_D (par heure) pouvant être revendiquée pour le sous-système**
	Il est admis que les sous-systèmes de catégorie spécifiée ont les caractéristiques indiquées ci-dessous		**PFH_D (MTTF$_{sous\text{-}système}$, T$_{test}$, DC) (voir Note 1)**
1	0	0 %	A indiquer par le fournisseur ou utiliser des données génériques (voir Annexe D)
2	0	60 % - 90 %	$\geq 10^{-6}$
3	1	60 % - 90 %	$\geq 2 \times 10^{-7}$
4	>1	60 % - 90 %	$\geq 3 \times 10^{-8}$
	1	> 90 %	$\geq 3 \times 10^{-8}$

NOTE 1 La valeur seuil PFH_D est une fonction de la MTTF du sous-système (à dériver par le fabricant du sous-système ou à partir des manuels de données appropriés des composants), de la durée de cycle du test/vérification comme spécifiée dans la spécification des exigences de sécurité (cette information est aussi prescrite pour la validation du sous-système selon l'ISO 13849-2:2003, 3.5) et de la couverture du diagnostic comme indiquée dans ce Tableau (ces valeurs sont basées sur les exigences des catégories décrites dans l'ISO 13849-1:1999).

NOTE 2 La catégorie B selon l'ISO 13849-1:1999 ne peut pas être considérée suffisante pour réaliser le niveau SIL1.

6.7.8.2 Approche simplifiée pour l'estimation de la probabilité de défaillance dangereuse aléatoire du matériel des sous-systèmes

6.7.8.2.1 Généralités

Ce paragraphe décrit une approche simplifiée de l'estimation de la probabilité de défaillance dangereuse aléatoire du matériel pour un nombre d'architectures de sous-systèmes simples et donne les formules qui peuvent être utilisées pour les sous-systèmes assemblés à partir d'éléments de sous-systèmes de faible complexité ou d'éléments de sous-systèmes complexes. Les formules sont en elles-mêmes une simplification de la théorie de l'analyse de fiabilité et sont prévues pour fournir des estimations qui sont en faveur de la sécurité. La condition première de la validité de toutes les formules données dans ce paragraphe est que $1 >> \lambda \times T_1$, où T_1 est la plus petite valeur de l'intervalle de test périodique ou la durée de vie, et que le sous-système fonctionne dans le "mode forte sollicitation ou continu" (voir 3.2.27).

NOTE 1 Les résultats obtenus représentent une limitation de la probabilité de défaillance dangereuse aléatoire du matériel des sous-systèmes et lorsque celle-ci se révèle inacceptable, il est possible d'appliquer des techniques de modélisation plus précises (voir 6.7.8.1.1).

NOTE 2 Pour les équations (A) à (D) données en 6.7.8.2, on prend pour hypothèses que les taux de défaillance (λ) des éléments de sous-systèmes sont constants et suffisamment faibles ($1>> \lambda \times T$) (cela veut dire qu'il faut que la durée moyenne de fonctionnement avant défaillance dangereuse soit beaucoup plus grande que l'intervalle de test périodique ou que la durée de vie du sous-système). De ce fait, les équations de base suivantes peuvent être utilisées:

- λ = 1/MTTF, où MTTF est exprimé en heures.

Pour les appareils électromécaniques, le taux de défaillance est déterminé en utilisant la valeur B_{10} et le nombre de cycles de manoeuvre C (exprimé comme le nombre de cycles de manoeuvre par heure) de l'application comme spécifié (voir 5.2.3).

- $\lambda = 0{,}1 \times C/B_{10}$.

NOTE 3 Les termes utilisés sont les suivants:

- $\lambda = \lambda_S + \lambda_D$; où λ_S est le taux de défaillance en sécurité et λ_D est le taux de défaillance dangereuse.
- $PFH_D = \lambda_D \times 1h$; probabilité moyenne de défaillance dangereuse en une heure.
- T_2: Intervalle de test de diagnostic.
- T_1: Intervalle de test périodique ou durée de vie selon la valeur la plus faible.

6.7.8.2.2 Architecture de type A d'un sous-système simple: tolérance aux anomalies de zéro sans fonction de diagnostic

Dans cette architecture, toute défaillance dangereuse d'un élément de sous-système entraîne une défaillance de la SRCF. Pour l'architecture de type A, la probabilité de défaillance dangereuse du sous-système est égale à la somme des probabilités de défaillance dangereuse de tous les éléments de sous-systèmes.

$$\lambda_{DssA} = \lambda_{De1} + \ldots + \lambda_{Den} \qquad \text{(A)}$$

$$PFH_{DssA} = \lambda_{DssA} \times 1h$$

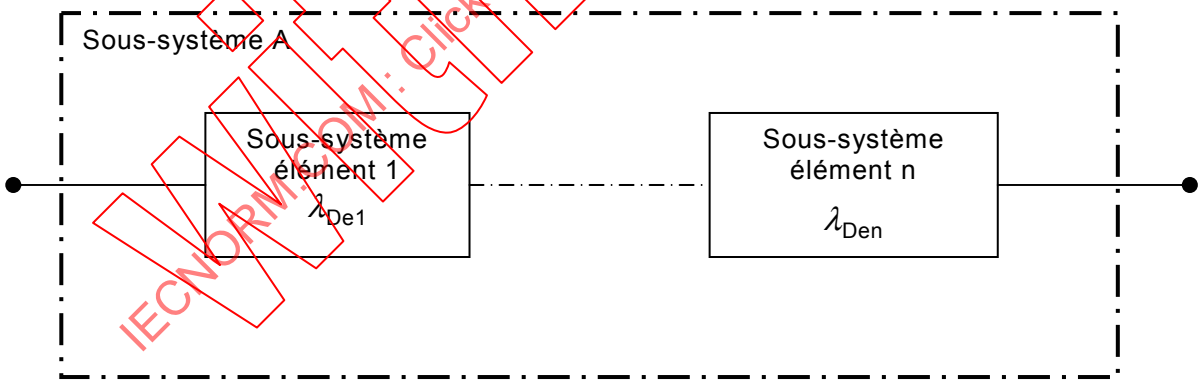


Figure 6 – Représentation logique d'un sous-système de type A

NOTE La Figure 6 est une représentation logique de l'architecture d'un sous-système de type A et il convient qu'elle ne soit pas interprétée comme sa réalisation physique.

6.7.8.2.3 Architecture B d'un sous-système simple: tolérance aux anomalies uniques sans fonction de diagnostic

Cette architecture est telle qu'une défaillance unique de tout élément de sous-système n'entraîne pas une perte de la SRCF. Ainsi, il pourrait se produire une défaillance dangereuse dans plus d'un élément avant qu'une défaillance de la SRCF ne puisse se produire. Pour une architecture de type B, la probabilité de défaillance dangereuse du sous-système est:

$$\lambda_{DssB} = (1 - ß)^2 \times \lambda_{De1} \times \lambda_{De2} \times T_1 + ß \times (\lambda_{De1} + \lambda_{De2})/2 \qquad (B)$$

$$PFH_{DssB} = \lambda_{DssB} \times 1h$$

où
T_1 est l'intervalle de test périodique ou la durée de vie selon la valeur la plus faible.
ß est la sensibilité aux défaillances de cause commune.

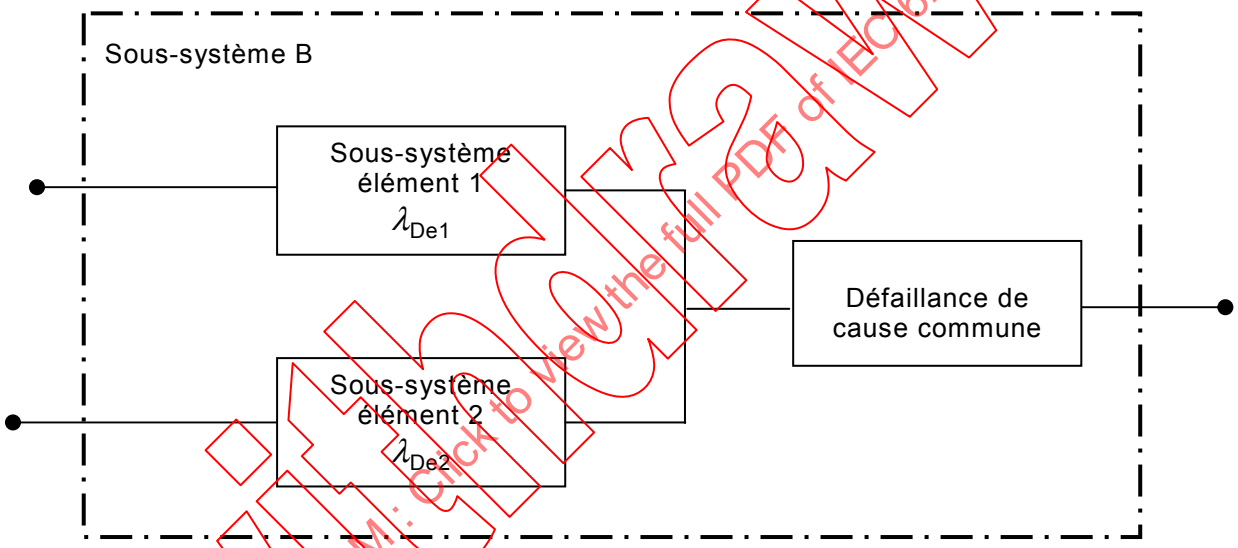


Figure 7 – Représentation logique d'un sous-système de type B

NOTE La Figure 7 est une représentation logique de l'architecture d'un sous-système de type B et il convient qu'elle ne soit pas interprétée comme sa réalisation physique.

6.7.8.2.4 Architecture de type C d'un sous-système simple: tolérance aux anomalies de zéro avec fonction de diagnostic

Toute anomalie dangereuse non détectée d'un élément de sous-système conduit à une défaillance dangereuse de la SRCF. Lorsqu'une anomalie d'un élément de sous-système est détectée, la(les) fonction(s) de diagnostic déclenche(nt) une fonction réaction à l'anomalie (voir 6.3.2). Pour une architecture de type C, la probabilité de défaillance dangereuse du sous-système est:

$$\lambda_{DssC} = \lambda_{De1}\ (1 - DC_1) + \ldots. + \lambda_{Den}(1 - DC_n) \quad (C)$$

$$PFH_{DssC} = \lambda_{DssC} \times 1h$$

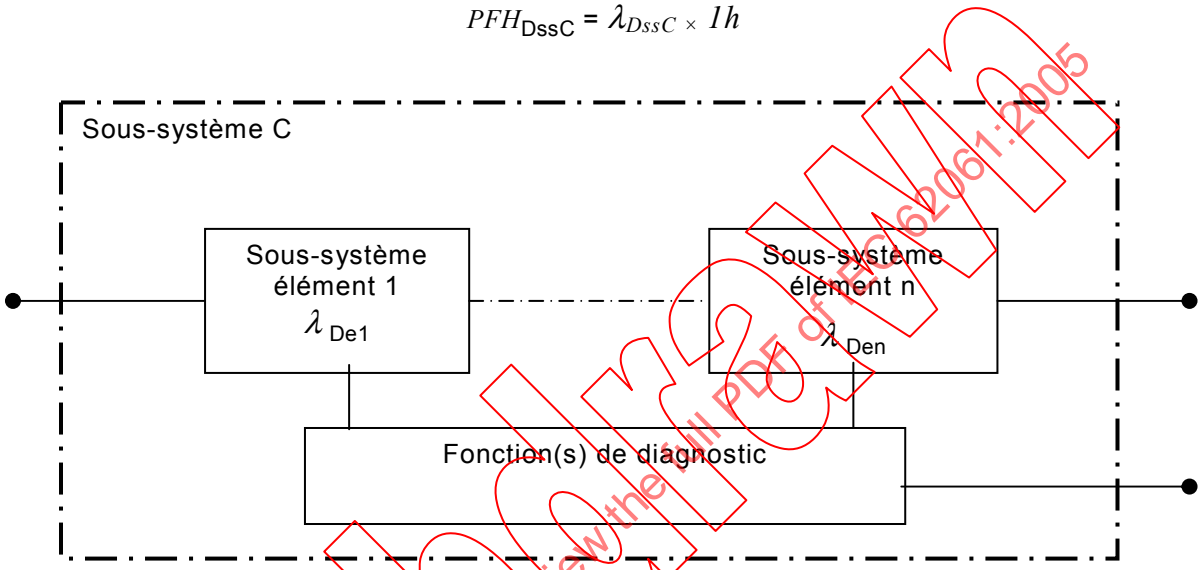


Figure 8 – Représentation logique d'un sous-système de type C

NOTE La Figure 8 est une représentation logique de l'architecture d'un sous-système de type C et il convient qu'elle ne soit pas interprétée comme sa réalisation physique. La fonction de diagnostic indiquée peut être réalisée par:

- le sous-système qui requiert des diagnostics; ou
- d'autres sous-systèmes du SRECS; ou
- des sous-systèmes non impliqués dans le fonctionnement de la fonction de commande relative à la sécurité.

6.7.8.2.5 Architecture de type D d'un sous-système simple: tolérance aux anomalies uniques avec fonction(s) de diagnostic

Cette architecture est telle qu'une défaillance unique de tout élément de sous-système n'entraîne pas une perte de la SRCF, où

T_2 est l'intervalle de test périodique;

T_1 est l'intervalle de test périodique ou la durée de vie selon la valeur la plus faible.

ß est la sensibilité aux défaillances de cause commune; $\lambda_D = \lambda_{DD} + \lambda_{DU}$; où λ_{DD} est le taux de défaillance dangereuse détectable et λ_{DU} est le taux de défaillance dangereuse non détectable.

$\lambda_{DD} = \lambda_D \times DC$

$\lambda_{DU} = \lambda_D \times (1\text{-}DC)$

Pour les éléments de sous-systèmes de conception différente:

λ_{De1} est le taux de défaillance dangereuse de l'élément de sous-système 1;

DC_1 est la couverture de diagnostic de l'élément de sous-système 1;

λ_{De2} est le taux de défaillance dangereuse de l'élément de sous-système 2;

DC_2 est la couverture de diagnostic de l'élément de sous-système 2.

$$\lambda_{DssD} = (1 - \beta)^2 \{[\lambda_{De1} \times \lambda_{De2} \times (DC_1 + DC_2)] \times T_2/2 + [\lambda_{De1} \times \lambda_{De2} \times (2 - DC_1 - DC_2)] \times T_1/2\} + \beta \times (\lambda_{De1} + \lambda_{De2})/2 \quad \text{(D.1)}$$

$$PFH_{DssD} = \lambda_{DssD} \times 1h$$

Pour les éléments de sous-systèmes de conception identique:

λ_{De} est le taux de défaillance dangereuse de l'élément de sous-système 1 ou 2;

DC est la couverture de diagnostic de l'élément de sous-système 1 ou 2.

$$\lambda_{DssD} = (1 - \beta)^2 \{[\lambda_{De}^2 \times 2 \times DC] \times T_2/2 + [\lambda_{De}^2 \times (1 - DC)] \times T_1\} + \beta \times \lambda_{De} \quad \text{(D.2)}$$

$$PFH_{DssD} = \lambda_{DssD} \times 1h$$

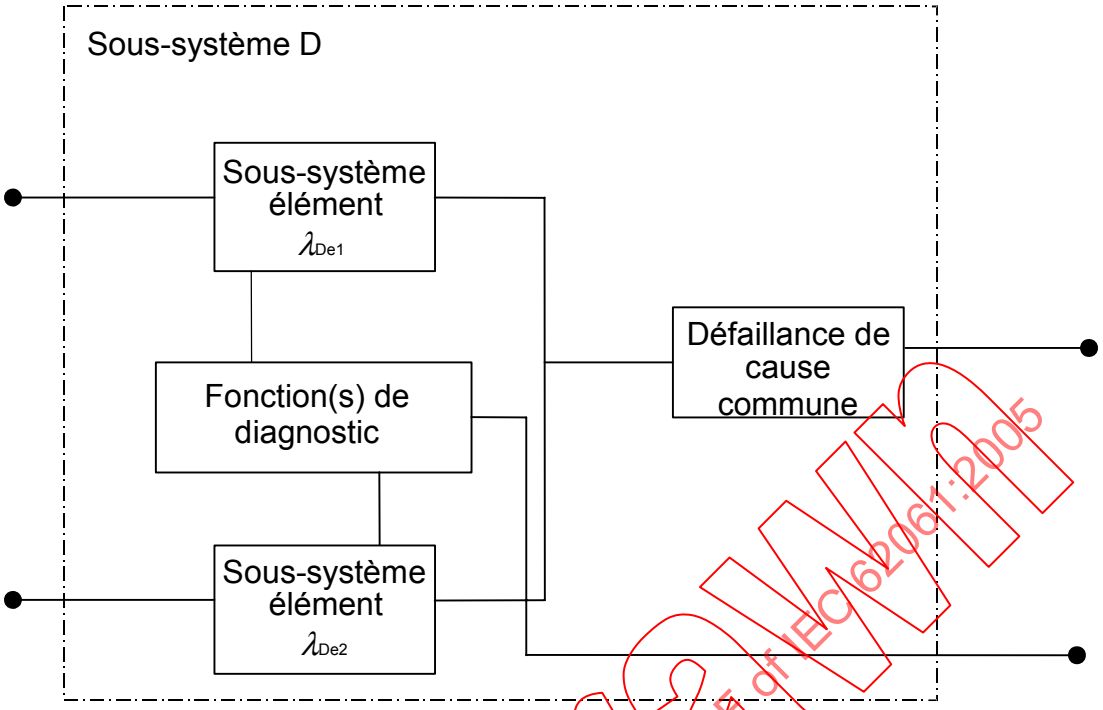


Figure 9 – Représentation logique d'un sous-système de type D

NOTE 1 La Figure 9 est une représentation logique de l'architecture d'un sous-système de type D et il convient qu'elle ne soit pas interprétée comme sa réalisation physique. La(les) fonction(s) de diagnostic indiquée(s) peut(vent) être réalisée(s) par:

– le sous-système qui requiert des diagnostics; ou

– d'autres sous-systèmes du SRECS; ou

– des sous-systèmes non impliqués dans le fonctionnement de la fonction de commande relative à la sécurité.

NOTE 2 Il est pris pour hypothèse que la réaction à l'anomalie pour ce sous-système est l'arrêt du fonctionnement approprié comme prescrit en 6.3.1. Lorsqu'une réparation en ligne est incorporée dans la conception, si la réaction à l'anomalie consiste à faire état de l'anomalie mais sans terminer le fonctionnement concerné, il convient de déterminer pour l'architecture restante un nouveau PFH_D pour le sous-système après apparition d'une première anomalie.

6.7.8.3 Approche simplifiée pour l'estimation de la contribution des défaillances de cause commune (CCF)

6.7.8.3.1 La connaissance de la sensibilité d'un sous-système à la CCF est nécessaire pour contribuer à l'estimation de la probabilité de défaillance dangereuse aléatoire du matériel d'un sous-système (voir 6.7.8.1).

6.7.8.3.2 Lorsqu'une architecture redondante est employée pour réaliser la probabilité requise de défaillance dangereuse aléatoire du matériel d'un sous-système et qu'une(des) CCF peut(peuvent) annuler l'effet de cette redondance, la probabilité de défaillance dangereuse aléatoire du matériel basée sur la probabilité d'apparition de la cause commune doit être ajoutée à la probabilité de défaillance dangereuse aléatoire du matériel d'un sous-système basé sur l'emploi de la redondance.

6.7.8.3.3 La probabilité d'apparition de la CCF dépendra habituellement d'une combinaison de technologie, d'architecture, d'application et d'environnement. L'utilisation de l'Annexe F sera efficace pour éviter plusieurs types de CCF.

6.7.8.3.4 L'Annexe F contient un tableau de points et une méthodologie associée pouvant être utilisée pour estimer l'efficacité des mesures développées lors de la conception d'un sous-système pour réduire la sensibilité aux CCF.

6.7.9 Exigences pour l'intégrité de sécurité systématique des sous-systèmes

Le SILCL dû à l'intégrité de sécurité systématique d'un sous-système va jusqu'à SIL3 lorsque les exigences de 6.7.9.1 et 6.7.9.2 sont satisfaites.

NOTE Ces exigences s'appliquent au "niveau sous-système" lorsque les éléments de sous-systèmes sont interconnectés pour réaliser un sous-système. Pour les autres exigences appropriées à la réalisation d'un SRECS, voir 6.4.

6.7.9.1 Exigences pour l'évitement des défaillances systématiques

6.7.9.1.1 Les mesures suivantes doivent s'appliquer:

a) choix, combinaison, arrangements, assemblage et montage corrects des composants, y compris les câbles, les conducteurs et toutes les interconnexions: utilisation de notes d'applications du fabricant et emploi des règles de l'art;

b) utilisation du sous-système et des éléments de sous-système dans le cadre de la spécification du fabricant et selon les instructions d'installation;

c) compatibilité: utilisation de composants ayant des caractéristiques de fonctionnement compatibles;

d) résistance à des conditions environnementales spécifiées: concevoir le sous-système de façon qu'il soit capable de travailler dans tous les environnements prévus et dans toutes les conditions adverses prévisibles, par exemple, température, humidité, vibrations et interférences électromagnétiques (EMI) (voir ISO 13849-2, Article D.1);

e) utilisation de composants selon une norme appropriée et ayant leurs modes de défaillance bien définis: pour réduire le risque d'anomalies non détectées par l'utilisation de composants ayant des caractéristiques spécifiées;

f) utilisation de matériaux appropriés et de fabrication adéquate: choix des matériaux, méthodes et procédés de fabrication en relation avec par exemple les contraintes, la durabilité, l'élasticité, la friction, l'usure, la corrosion, la température, la conductivité, la rigidité diélectrique;

g) dimensions et formes corrects: considérer les effets, par exemple, des contraintes, de la traction, de la fatigue, de la température, de la rugosité de surface, des tolérances de fabrication.

6.7.9.1.2 En complément, une ou plusieurs des mesures suivantes doivent s'appliquer en prenant en compte la complexité du sous-système:

a) la revue de conception du matériel (par exemple par inspection ou lecture croisée): pour révéler par les revues et/ou les analyses les divergences entre la spécification et la réalisation;

NOTE 1 Afin de faire apparaître les divergences entre la spécification et la réalisation, tous les points de doute ou pouvant présenter une faiblesse concernant la réalisation, la mise en œuvre et l'utilisation du produit sont documentés de façon à ce que qu'ils puissent être résolus, prenant en compte que lors d'une procédure d'inspection, l'auteur est passif et l'inspecteur est actif tandis que lors d'une procédure de lecture croisée, l'auteur est actif et l'inspecteur est passif.

b) des outils de conception assistée par ordinateur capables de simulation ou d'analyse: effectuer la procédure systématique de conception et inclure les éléments de construction automatiques appropriés déjà disponibles et testés;

NOTE 2 L'intégrité de ces outils peut être prouvée par des tests spécifiques, ou par un rapport détaillé sur une utilisation satisfaisante ou par une vérification indépendante de leurs résultats pour le sous-système particulier en cours de conception. Voir 6.11.3.4.

c) simulation: effectuer une simulation systématique de la conception d'un sous-système en termes de caractéristiques fonctionnelles, de dimensionnement et d'interaction corrects de ses composants.

NOTE 3 La fonction du sous-système peut être simulée sur ordinateur par un modèle comportemental de logiciel (voir 6.11.3.4) dans lequel les composants particuliers du circuit ont chacun leur propre comportement simulé et la réponse du sous-système dans lequel ils sont connectés est étudiée en observant les spécifications aux limites de chaque composant.

6.7.9.2 Exigences pour la maîtrise des anomalies systématiques

6.7.9.2.1 Les mesures suivantes doivent s'appliquer:

a) mesures pour maîtriser les effets d'un claquage de l'isolant, de variations et interruptions de tension, d'une surtension, d'une baisse de tension: le comportement du sous-système en réponse à un claquage de l'isolant, des variations et interruptions de tension, des conditions de surtension et de sous-tension doit être prédéterminé de façon que le sous-système puisse réaliser ou maintenir un état sûr du SRECS;

NOTE 1 Voir aussi les exigences appropriées de la CEI 60204-1. En particulier:

– il convient que la surtension soit détectée suffisamment tôt de façon que toutes les sorties puissent être commutées en position de sécurité par le programme de mise hors tension ou basculées sur une seconde unité d'alimentation; et/ou

– il convient que la tension du circuit de commande soit surveillée et qu'une mise hors tension soit initiée, ou qu'un basculement sur une seconde unité d'alimentation soit réalisé si la tension ne se trouve pas à l'intérieur de sa plage spécifiée; et/ou

– il convient aussi que la surtension ou la sous-tension soit détectée suffisamment tôt pour que l'état interne puisse être sauvegardé dans une mémoire non volatile (si nécessaire), de façon que toutes les sorties puissent être commutées en position de sécurité par le programme de mise hors tension ou basculées sur une seconde unité d'alimentation.

b) mesures pour maîtriser et éviter les effets de l'environnement physique (par exemple, la température, l'humidité, l'eau, les vibrations, la poussière, les substances corrosives, les interférences électromagnétiques et leurs effets): le comportement du sous-système en réponse aux effets de l'environnement physique doit être prédéterminé de façon que le SRECS puisse réaliser ou maintenir un état sûr de la machine. Voir aussi la CEI 60204-1;

c) mesures pour maîtriser ou éviter les effets d'un accroissement ou d'une diminution de la température si des variations de température sont susceptibles de se produire: il convient que le sous-système soit conçu de façon que, par exemple, une température anormale puisse être détectée avant que le sous-système ne commence à fonctionner en dehors de sa spécification.

NOTE 2 On peut trouver de plus amples informations dans la CEI 61508-7, Article A.10.

6.7.9.2.2 En complément, les mesures suivantes, si appropriées, doivent s'appliquer pour la maîtrise des défaillances systématiques:

– détection de défaillance par surveillance en ligne;

– tests comparatifs du matériel redondant;

– différentier les matériels;

– fonctionnement en mode positif (par exemple, un interrupteur de fin de course est actionné si un protecteur est ouvert);

– mode de défaillance orienté;

– surdimensionnement par un facteur approprié, lorsque le fabricant peut démontrer que la réduction améliorera la fiabilité.

NOTE 1 Lorsque le surdimensionnement est approprié, il convient d'utiliser un facteur de surdimensionnement d'au moins 1,5.

NOTE 2 On peut trouver de plus amples informations dans l'ISO 13849-2, Article D.3.

6.7.10 Ensemble sous-système

Les éléments de sous-systèmes doivent être combinés pour former un sous-système selon 6.7.4.3.1.2 et la conception détaillée documentée.

6.8 Réalisation des fonctions de diagnostic

6.8.1 Chaque sous-système doit être équipé des fonctions de diagnostic associées nécessaires pour satisfaire aux exigences pour les contraintes architecturales (6.7.6) et à la probabilité de défaillance dangereuse aléatoire du matériel (6.7.8).

6.8.2 Les fonctions de diagnostic sont considérées comme des fonctions séparées qui peuvent avoir une structure différente de celle de la SRCF et être réalisées par:

– le même sous-système qui requiert des diagnostics; ou

– d'autres sous-systèmes du SRECS; ou

– des sous-systèmes du SRECS n'effectuant pas la SRCF.

NOTE Voir aussi la Note 3 de 6.6.2.1.

6.8.3 Les fonctions de diagnostic doivent satisfaire ce qui suit, pour autant qu'applicable à leurs SRCF associées:

– les exigences pour l'évitement des défaillances (voir 6.7.9.1); et

– les exigences pour la maîtrise des anomalies systématiques (voir 6.7.9.2).

6.8.4 La probabilité de défaillance de la(des) fonction(s) de diagnostic du SRECS doit être prise en compte lors de l'estimation de la probabilité de défaillance dangereuse de la SRCF.

NOTE 1 Voir aussi la Note 3 de 6.6.2.1.

NOTE 2 Les contraintes temporelles applicables lors des tests du sous-système réalisant une fonction de diagnostic peuvent différer de celles applicables aux SRCF et, en général, il convient que l'intervalle de test satisfasse aux exigences applicables à un sous-système ayant une tolérance aux anomalies du matériel égale à 1.

NOTE 3 Il convient qu'une défaillance d'une (des) fonction(s) de diagnostic soit détectée et qu'une réaction appropriée soit déclenchée afin d'assurer que la contribution de la fonction de diagnostic à l'intégrité de sécurité de la SRCF soit maintenue. La défaillance d'une (des) fonction(s) de diagnostic peut être détectée par des tests en ligne, un contrôle croisé par le matériel redondant, etc.

6.8.5 Une description claire de la (des) fonction(s) de diagnostic du SRECS, de leur(s) détection(s)/réaction(s) vis-à-vis des défaillances ainsi qu'une analyse de leur contribution à l'intégrité de sécurité des SRCF associées doivent être fournies.

6.8.6 Pour appliquer l'approche simplifiée pour l'estimation de la probabilité de défaillance dangereuse aléatoire du matériel des sous-systèmes (6.7.8.2), les dispositions suivantes doivent s'appliquer:

- lorsqu'une(des) fonction(s) de diagnostic d'un SRECS est(sont) nécessaire(s) pour réaliser la probabilité de défaillance dangereuse aléatoire du matériel requise et que le sous-système a une tolérance aux anomalies du matériel de zéro, alors la détection d'anomalie ainsi que la réaction à l'anomalie spécifiée doivent être exécutées avant que le phénomène dangereux dû à cette anomalie ne puisse se produire; et
- la(les) fonction(s) de diagnostic du SRECS doit(doivent) au minimum être réalisée(s) de sorte que la probabilité de défaillance aléatoire du matériel et l'intégrité de sécurité systématique soient identiques à celles spécifiées pour la(les) SRCF correspondante(s); ou

  NOTE 1 Il n'est pas nécessaire que les contraintes architecturales sur l'intégrité de sécurité du matériel s'appliquent à la réalisation de la(des) fonction(s) de diagnostic.

- lorsque la probabilité de défaillance dangereuse aléatoire du matériel est d'un ordre de grandeur plus important que celui spécifié pour la SRCF, alors un test doit être effectué afin de déterminer lequel de l'(des) appareil(s) de diagnostic ou de la(des) fonction(s) de diagnostic continue à fonctionner. Il est admis qu'un tel test de la(des) fonction(s) de diagnostic ou de l'(des) appareil(s) de diagnostic est effectué avec un intervalle de test 10 fois supérieur à l'intervalle des tests périodiques appliqués au sous-système.

NOTE 2 Il convient qu'un test de la(des) fonction(s) de diagnostic couvre autant que possible 100 % des parties réalisant la(les) fonction(s) de diagnostic.

NOTE 3 Lorsqu'une fonction de diagnostic est réalisée par l'unité logique du SRECS, il peut ne pas être nécessaire de réaliser un test particulier de la fonction de diagnostic puisque sa défaillance peut être révélée par une défaillance de la SRCF.

NOTE 4 Un test peut être réalisé soit par des moyens externes (par exemple, matériel de test) ou par des vérifications dynamiques internes (par exemple intégrées dans l'unité logique) du SRECS.

6.9 Réalisation du matériel d'un SRECS

Le SRECS doit être réalisé conformément à la conception documentée du SRECS.

6.9.1 Interconnexion d'un SRECS

6.9.1.1 Le SRECS doit être interconnecté de façon à satisfaire aux parties appropriées de la spécification des exigences de sécurité du SRECS ainsi qu'aux exigences concernant les conducteurs, les pratiques de raccordement et de câblage de la CEI 60204-1.

6.9.1.2 Les mesures pour l'évitement et la maîtrise des défaillances des conducteurs et des câbles d'interconnexion doivent être réalisées conformément à 6.4.1 et 6.4.2.

6.10 Spécification des exigences de sécurité du logiciel

6.10.1 Généralités

Si on est appelé à utiliser du logiciel dans une quelconque partie d'un SRECS réalisant une(des) fonction(s) de commande relative(s) à la sécurité, une spécification des exigences de sécurité du logiciel doit être développée et documentée.

6.10.2 Exigences

6.10.2.1 Une spécification des exigences de sécurité du logiciel doit être développée pour chaque sous-système à partir de la spécification et de l'architecture du SRECS.

6.10.2.2 La spécification des exigences pour la sécurité du logiciel pour chaque sous-système doit être déduite (1) des exigences de sécurité spécifiées de la SRCF, (2) des exigences résultant de l'architecture du SRECS et (3) de toute exigence du plan de sécurité fonctionnel (voir 4.2). Ces informations doivent être rendues disponibles pour le développeur du logiciel d'application.

6.10.2.3 La spécification des exigences de sécurité du logiciel d'application doit être suffisamment détaillée pour permettre la conception et la réalisation d'un SRECS accomplissant l'intégrité de sécurité prescrite, ainsi que la vérification.

6.10.2.4 Le développeur du logiciel d'application doit passer en revue les informations contenues dans la spécification afin de s'assurer que les exigences sont établies de façon adéquate. En particulier, le développeur du logiciel doit se conformer à la présente norme sur les points suivants:

– les SRCF;
– la configuration de l'architecture du système;
– le fonctionnement en termes d'aptitude et de temps de réponse;
– les interfaces avec les matériels et l'opérateur;
– tous les modes appropriés de fonctionnement de la machine tels qu'établis dans la spécification des exigences de sécurité;
– les tests de diagnostic des appareils externes (par exemple les capteurs et les éléments finaux).

6.10.2.5 Les exigences spécifiées pour l'intégrité du logiciel doivent être exprimées et structurées de façon qu'elles soient:

– claires, vérifiables, testables, actualisables et manoeuvrables, en rapport avec le niveau d'intégrité de sécurité;
– traçables par rapport à la spécification des exigences de sécurité du SRECS;
– exemptes de toute terminologie et descriptions ambiguës.

6.10.2.6 La spécification des exigences de sécurité du logiciel doit spécifier les propriétés requises de chaque sous-système en fournissant les informations permettant le choix adapté des matériels. Pour les SRCF liées au logiciel, les exigences suivantes doivent être spécifiées:

– la logique (c'est-à-dire la fonctionnalité) de tous les blocs fonctionnels attribuée à chaque sous-système;
– les interfaces des entrées et des sorties assignées à chaque bloc fonctionnel;
– les formats et les plages de valeurs des données d'entrée et de sortie et leur relation au bloc fonctionnel;
– les données appropriées pour décrire toutes les limites de chaque bloc fonctionnel, par exemple le temps de réponse maximal, les valeurs limites pour les contrôles de vraisemblance;
– les fonctions de diagnostic des autres appareils compris dans le SRECS (par exemple les capteurs et les éléments finaux) à réaliser par chaque sous-système;
– les fonctions qui permettent à la machine d'atteindre ou de maintenir un état sûr;
– les fonctions liées à la détection, la signalisation et le traitement des anomalies;
– les fonctions liées au test périodique des SRCF en ligne et hors ligne;
– les fonctions qui empêchent une modification non autorisée du SRECS;
– les interfaces en général excepté avec les SRCF; et
– le fonctionnement en termes d'aptitude et de temps de réponse.

NOTE Les interfaces comprennent les facilités de programmation en ligne et hors ligne.

6.10.2.7 Lorsque c'est approprié, des méthodes semi-formelles telles que diagrammes de blocs logiques/fonctionnels, ou diagrammes de séquence doivent être utilisés dans la documentation.

NOTE Des recommandations sur la documentation du logiciel sont données dans la CEI 61506, l'ISO/CEI 15910 et l'ISO/CEI 9254.

6.11 Conception et développement du logiciel

6.11.1 Conception et développement du logiciel intégré

Un logiciel intégré, incorporé dans des sous-systèmes doit être conforme à la CEI 61508-3 pour autant qu'applicable pour le niveau de SIL prescrit.

NOTE 1 Voir aussi 6.7.3.2.

NOTE 2 L'Annexe C est destinée à fournir une aide à la conception et au développement d'un logiciel intégré utilisé pour la réalisation des SRCF dans un SRECS.

6.11.2 Paramétrisation liée au logiciel

6.11.2.1 La paramétrisation liée au logiciel des paramètres relatifs à la sécurité doit être considérée comme un aspect relatif à la sécurité de la conception d'un SRECS, laquelle est décrite dans la spécification des exigences de sécurité du logiciel (voir 6.10). La paramétrisation doit être effectuée à l'aide d'un outil dédié fourni par le fabricant du SRECS ou du(des) sous-système(s) lié(s). Cet outil doit posséder sa propre identification (nom, version, etc.). Cet outil de paramétrisation doit empêcher une modification non autorisée, par exemple par l'utilisation d'un mot de passe.

6.11.2.2 L'intégrité de toutes les données utilisées pour la paramétrisation doit être actualisée. Ceci doit être réalisé en mettant en œuvre des mesures pour:

– maîtriser la plage des entrées valides;
– maîtriser l'altération des données avant transmission;
– maîtriser les effets des erreurs du processus de transmission des paramètres;
– maîtriser les effets d'une transmission de paramètres incomplète; et
– maîtriser les effets des anomalies et des défaillances des matériel et logiciel de l'outil utilisé pour la paramétrisation.

6.11.2.3 L'outil utilisé pour la paramétrisation doit satisfaire aux exigences suivantes:

– toutes les exigences appropriées pour un sous-système au sens de la présente norme afin d'assurer une paramétrisation correcte; ou
– une procédure particulière doit être utilisée pour le réglage des paramètres relatifs à la sécurité. Cette procédure doit comprendre la confirmation des paramètres d'entrée du SRECS par soit:
  - la retransmission des paramètres modifiés vers l'outil de paramétrisation; ou
  - d'autres moyens de confirmation de l'intégrité des paramètres;

  et la confirmation ultérieure (par exemple par une personne qualifiée convenablement et une vérification automatique par un outil de paramétrisation);

  NOTE Ceci revêt une importance particulière lorsque la paramétrisation est réalisée en utilisant un dispositif qui n'est pas particulièrement destiné à cet usage (par exemple ordinateur personnel ou équivalent).
– les modules de logiciel utilisés pour le codage/décodage dans les processus de transmission/retransmission ainsi que les modules de logiciel utilisés pour la visualisation des paramètres relatifs à la sécurité destinée à l'utilisateur doivent au minimum utiliser la diversité dans la(les) fonction(s) destinée(s) à éviter les défaillances systématiques.

6.11.2.4 La documentation concernant la paramétrisation liée au logiciel doit indiquer les données utilisées (par exemple les réglages de paramètres prédéfinis), les informations nécessaires à l'identification des paramètres associés au SRECS, la(les) personne(s) effectuant (ensemble) la paramétrisation ainsi que les autres informations appropriées telles que la date de la paramétrisation.

6.11.2.5 Les activités de vérification suivantes doivent s'appliquer à la paramétrisation liée au logiciel:

– vérification du réglage correct pour chaque paramètre relatif à la sécurité (valeurs minimale, maximale et valeurs représentatives);
– vérification que les paramètres relatifs à la sécurité sont soumis à un contrôle de vraisemblance, par exemple par la détection de valeurs invalides, etc.;
– vérification que les modifications non autorisées des paramètres relatifs à la sécurité sont empêchées;

– vérification que les signaux/données pour la paramétrisation sont générés et traités de telle façon que les anomalies ne peuvent entraîner une perte de la(des) SRCF.

NOTE Ceci revêt une importance particulière lorsque la paramétrisation est réalisée en utilisant un dispositif qui n'est pas particulièrement destiné à cet usage (par exemple ordinateur personnel ou équivalent).

6.11.3 Conception et développement du logiciel d'application

NOTE Ce paragraphe est basé sur la CEI 61508-3.

6.11.3.1 Exigences générales

6.11.3.1.1 Les exigences de la CEI 61508-3 s'appliquent aux langages de variabilité totale (FVL). Les exigences suivantes doivent s'appliquer aux logiciels d'applications basés sur les langages de variabilité limitée (LVL).

6.11.3.1.2 Les résultats des activités réalisées pendant le développement du logiciel d'application doivent être vérifiés à des stades appropriés.

6.11.3.1.3 La méthode de conception et le langage de l'application choisis pour satisfaire au niveau de SIL prescrit de la SRCF doivent posséder des caractéristiques appropriées à l'application qui facilitent:

a) l'abstraction, la modularité et autres caractéristiques pour maîtriser la complexité; à chaque fois que possible, le logiciel doit être basé sur des fonctions logiques éprouvées comprenant des fonctions bibliothèque utilisateur et des règles bien définies pour la liaison des fonctions logiques;

b) l'expression
   - de la fonctionnalité, idéalement comme une description logique ou des fonctions algorithmiques;
   - du flot d'informations entre les éléments modulaires;
   - du séquencement et des exigences liées au temps;
   - des contraintes temporelles;
   - des structures de données et de leurs propriétés, y compris les types de données, la validité des plages de données;

c) la compréhension par les développeurs et tous ceux qui ont besoin de comprendre la conception, à la fois à partir d'une compréhension fonctionnelle de l'application et d'une connaissance des contraintes liées à la technologie du SRECS;

d) la vérification et la validation, y compris des tests structurels (boîte blanche) du logiciel d'application, des tests fonctionnels (boîte noire), du programme d'application intégré et de l'essai de l'interface (boîte grise), de l'interaction avec le SRECS et sa configuration spécifique du matériel d'application;

e) la modification sûre.

6.11.3.1.4 Les tests doivent constituer la méthode principale de vérification utilisée pour le logiciel d'application. Le programme de test doit couvrir ce qui suit:

– la politique de vérification de l'intégration du logiciel et du matériel;

– les tests élémentaires et les résultats des tests;

– les types de tests à réaliser;

– les outils y compris le matériel de test, le logiciel support et la description de la configuration;

– les critères de test sur lesquels l'achèvement du test doit être jugé;

– l'(les) emplacement(s) physique(s) (par exemple usine ou site);
– la dépendance d'une fonctionnalité externe;
– le nombre de tests élémentaires nécessaires; et
– l'exhaustivité au regard des fonctions ou des exigences concernées.

6.11.3.1.5 Lorsque le logiciel d'application a à accomplir à la fois des fonctions de commande relatives à la sécurité et des fonctions de commande non relatives à la sécurité, tout le logiciel d'application doit alors être traité comme relatif à la sécurité, excepté si une indépendance appropriée entre les fonctions peut être démontrée dans la conception.

6.11.3.1.6 La conception doit comprendre des vérifications d'intégrité des données et des contrôles de vraisemblance de la couche application (par exemple des vérifications dans les liaisons de communication, des vérifications des limites sur les entrées des capteurs et des limites sur les paramètres de données).

6.11.3.1.7 La conception du logiciel d'application doit comprendre une auto-surveillance des flot de commande et flot de données excepté si de telles fonctions sont incluses dans le logiciel intégré. En cas de détection de défaillance, des actions appropriées doivent être accomplies pour réaliser ou maintenir un état sûr.

6.11.3.1.8 Si des fonctions bibliothèque logiciel développées précédemment doivent être utilisées pour la conception, leur capacité à satisfaire la spécification des exigences de sécurité du logiciel doit être justifiée. Cette capacité du logiciel doit être établie à partir de la constatation d'un fonctionnement satisfaisant dans des applications similaires pour lesquelles a été démontrée l'existence d'une fonctionnalité similaire, ou doit être soumise aux mêmes procédures de vérification et de validation que celles auxquelles sont normalement soumis les logiciels relatifs à la sécurité nouvellement développés. Les contraintes imposées par l'environnement du logiciel précédent (par exemple les dépendances au système d'exploitation et au compilateur) doivent être évaluées.

6.11.3.1.9 Toute modification ou correction du logiciel d'application doit être soumise à une analyse d'impact qui identifie tous les modules affectés du logiciel et les nécessaires activités de revérification pour confirmer que la spécification des exigences de sécurité du logiciel est encore satisfaite.

6.11.3.2 Gestion de la configuration du logiciel

6.11.3.2.1 Le plan de sécurité fonctionnelle doit définir la stratégie pour le développement, l'intégration, la vérification et la validation du logiciel.

6.11.3.2.2 La gestion de la configuration du logiciel doit:

– assurer que toutes les opérations nécessaires ont été effectuées en vue de démontrer que le niveau d'intégrité de sécurité requis du logiciel a été atteint;
– maintenir de manière précise et au moyen d'une identification unique tous les documents relatifs aux éléments de configuration qui sont nécessaires au maintien de l'intégrité du SRECS. Les éléments de configuration doivent comprendre au moins ce qui suit:
  - des exigences de sécurité et une analyse de sécurité;
  - la spécification du logiciel et les documents de conception;
  - les modules de code source du logiciel;
  - les programmes de tests et les résultats;
  - les progiciels et composants logiciels préexistants à incorporer dans le SRECS;
  - tous les outils et environnements de développement qui sont utilisés pour créer, tester ou effectuer une action sur le logiciel d'application;

– appliquer des procédures de maîtrise des modifications pour:
  - empêcher toute modification non autorisée;
  - documenter les demandes de modification;
  - analyser l'impact des modifications proposées, et approuver ou rejeter la(les) demande(s) de modification;
  - documenter les détails et les autorisations pour toutes les modifications approuvées;
  - documenter la configuration du logiciel à des points-clés appropriés lors du développement du logiciel;
– documenter les informations suivantes pour permettre un audit ultérieur: état des versions, justification et approbation de toutes les modifications, et détails des modifications;
– documenter de manière formelle la version du logiciel d'application. Conserver obligatoirement les copies originales du logiciel et toute la documentation associée afin de permettre la maintenance et la modification au cours de la vie opérationnelle de la version du logiciel.

6.11.3.3 Exigences pour l'architecture du logiciel

NOTE 1 L'architecture du logiciel définit les composants majeurs et les sous-systèmes d'un système ainsi que le logiciel d'application, la manière dont ils sont interconnectés et comment il convient de réaliser leurs attributions prescrites. Certains des modules logiciels d'application comprennent des fonctions d'application reproduites à travers toute la machine, les entrées/sorties de la machine, les composants prioritaires et d'inhibition, la vérification de la validité des données ainsi que les vérifications des limites, etc.

NOTE 2 L'architecture du logiciel est aussi affectée par l'architecture de base du sous-système délivré par le fournisseur.

6.11.3.3.1 La conception de l'architecture du logiciel doit être basée sur la spécification de sécurité du SRECS prescrite prenant en compte les contraintes de l'architecture système du SRECS et la conception du sous-système.

6.11.3.3.2 La conception de l'architecture du logiciel doit:

a) fournir une description compréhensible de la structure interne et du fonctionnement du SRECS et de ses composants (voir Note);

b) inclure la spécification de tous les composants logiciels identifiés, et la description des connexions et interactions entre ces composants identifiés (logiciel et matériel);

c) inclure la conception interne et l'architecture de tous les composants identifiés qui ne sont pas des boîtes noires;

d) identifier les modules logiciels contenus dans le SRECS mais non utilisés dans l'un quelconque des modes de fonctionnement relatifs à la sécurité.

NOTE Il est particulièrement important que la documentation sur l'architecture soit à jour et complète vis-à-vis du SRECS.

6.11.3.3.3 Le jeu de techniques et de mesures nécessaires lors de la conception du logiciel d'application pour satisfaire à la spécification doit être décrit et justifié. Ces techniques et mesures doivent avoir pour but de garantir la prédictibilité du comportement du SRECS et doivent être cohérentes avec toutes les contraintes identifiées dans la documentation du SRECS.

6.11.3.3.4 Les mesures pour maintenir l'intégrité de toutes les données doivent être décrites et justifiées. Ces données peuvent comprendre les données d'entrée/sortie de la machine, les données de communication, les données d'interfaces de fonctionnement, les données d'entretien et les données de la base de données interne.

6.11.3.4 Exigences pour les outils supports, le manuel utilisateur et les langages d'application

6.11.3.4.1 Un jeu adapté d'outils, comprenant la gestion de la configuration, la simulation et les outils de simulation de tests, doit être choisi. On doit prendre en considération la capacité de ces outils adaptés (pas nécessairement ceux utilisés pendant le développement initial du système) à fournir les services appropriés pendant la durée de vie du SRECS. La pertinence de ces outils doit être expliquée et documentée.

NOTE Le choix des outils de développement dépend de la nature des activités de développement du logiciel, du logiciel intégré et de l'architecture du logiciel. Des outils de vérification et de validation tels que des analyseurs de codes et des simulateurs peuvent être nécessaires.

6.11.3.4.2 Lorsque c'est nécessaire, un sous-ensemble du langage de programmation de l'application doit être défini.

6.11.3.4.3 Le logiciel d'application doit être défini en prenant en compte les contraintes et les faiblesses connues listées dans les manuels utilisateurs du SRECS et du(des) sous-système(s).

6.11.3.4.4 Le langage d'application choisi doit soit:

– être traité par un traducteur/compilateur qui doit être évalué en vue d'établir son aptitude à l'utilisation prévue;

– être totalement défini de façon non ambiguë, ou limité à des caractéristiques définies de façon non ambiguë;

– correspondre aux caractéristiques de l'application;

NOTE Les caractéristiques d'une application font référence par exemple à l'une quelconque des contraintes de fonctionnement.

– comporter des caractéristiques qui facilitent la détection des erreurs de programmation; et

– supporter des caractéristiques adaptées à la méthode de conception;

ou les insuffisances du langage utilisé doivent alors être documentées dans la description de la conception de l'architecture du logiciel et l'aptitude à l'utilisation prévue du langage doit être expliquée, y compris les mesures supplémentaires nécessaires pour couvrir les défauts identifiés du langage.

6.11.3.4.5 Les procédures d'utilisation du langage d'application doivent spécifier les règles de l'art en ce qui concerne la configuration, proscrire les caractéristiques logicielles génériques non sûres (par exemple, les caractéristiques langage non définies, les conceptions non structurées, etc.), identifier les vérifications pouvant être utilisées pour détecter les erreurs de configuration et spécifier les procédures de documentation du programme d'application. Les informations minimales suivantes doivent être contenues dans la documentation du programme d'application:

a) l'entité légale (exemple: la société, le ou les auteurs, etc.);

b) la description;

c) la traçabilité des exigences fonctionnelles d'application;

d) la traçabilité de la fonction bibliothèque normale;

e) les entrées et sorties; et

f) la gestion de la configuration.

6.11.3.5 Exigences pour la conception du logiciel d'application

6.11.3.5.1 Les informations suivantes doivent être disponibles avant le début de la conception détaillée du logiciel d'application:

– la spécification des exigences de sécurité du logiciel;

– la description de la conception de l'architecture logicielle y compris l'identification de la logique d'application et la fonctionnalité de tolérance aux anomalies, une liste des données d'entrée et de sortie, les modules logiciels génériques et les outils supports devant être utilisés ainsi que les procédures pour configurer le logiciel d'application avec les matériels disponibles, afin de fournir la fonctionnalité d'application pour l'E/S définie; et

– le plan destiné à valider la sécurité du logiciel.

6.11.3.5.2 Le logiciel d'application doit être produit d'une façon structurée pour réaliser:

– la modularité de la fonctionnalité d'application et les données de maîtrise des E/S;

– la testabilité de la fonctionnalité (y compris les caractéristiques de tolérance aux anomalies) et de la structure interne;

– la capacité à être modifié de façon sûre au travers de dispositions pour assurer la traçabilité adéquate, l'explication des fonctions d'application et des contraintes associées.

6.11.3.5.3 Pour chaque composant/sous-système majeur dans la description de la conception de l'architecture du logiciel d'application (voir 6.11.3.5.1), le perfectionnement de la conception doit être basé sur:

– les fonctions utilisées de façon cyclique lors de la conception;

– la mise en correspondance des informations d'entrée/sortie des modules du logiciel d'application;

– la réalisation des fonctions d'application à partir des fonctions logicielles génériques et la mise en correspondance des E/S.

6.11.3.5.4 La conception de chaque module du logiciel d'application et les tests structurels devant être effectués pour chacun d'entre eux doivent être spécifiés.

6.11.3.5.5 Les tests d'intégration du SRECS et du logiciel approprié doivent être spécifiés pour garantir que le programme d'application satisfait aux exigences spécifiées pour la sécurité du logiciel d'application. Doivent être considérés ce qui suit:

– la division du logiciel d'application en ensembles d'intégration faciles à gérer;

– les tests élémentaires;

– les types de tests à réaliser;

– l'environnement de test, les outils, la configuration et les programmes;

– les critères de test sur lesquels l'achèvement du test doit être jugé; et

– les procédures d'actions correctives en cas de défaillance d'un test.

6.11.3.6 Exigences pour le développement du code d'application

6.11.3.6.1 Le logiciel d'application doit:

– être lisible, compréhensible et testable;

– satisfaire aux principes de conception appropriés;

– satisfaire aux exigences appropriées spécifiées pendant la planification de la sécurité.

6.11.3.6.2 Le logiciel d'application doit être revu pour garantir la conformité à la conception spécifiée, aux règles de codage et aux exigences de la planification de sécurité.

NOTE La revue du logiciel d'application comprend les techniques telles que les inspections du logiciel ou les lectures croisées, l'analyse de code ou la preuve mathématique. Il convient que ces techniques soient utilisées conjointement avec les tests et/ou la simulation pour donner l'assurance que le logiciel d'application satisfait à sa spécification associée.

6.11.3.7 Exigences pour le test du module d'application

NOTE Tester que le logiciel d'application satisfait correctement à sa spécification de test est une activité de vérification. Elle est la combinaison de la revue de code et du test structurel et donne l'assurance qu'un module logiciel d'application satisfait à sa spécification associée, c'est-à-dire qu'il est vérifié.

6.11.3.7.1 La configuration de chaque point d'entrée et chaque point de sortie doit être vérifiée au travers de la revue, des tests ou de la simulation pour confirmer que les données d'E/S sont mises en correspondance avec la logique d'application correcte.

6.11.3.7.2 Chaque module logiciel doit être vérifié au travers d'un processus de revue, de simulation et de tests afin de déterminer que la fonction prévue est correctement exécutée et que les fonctions non prévues ne sont pas exécutées.

6.11.3.7.3 Les tests doivent être adaptés au module particulier soumis au test et doivent:

– assurer que chaque branche de l'un quelconque des modules logiciels d'application soit testée;
– assurer que les données aux limites sont testées;
– assurer que les séquences sont correctement réalisées, y compris les conditions de synchronisation appropriées.

6.11.3.7.4 Les résultats des tests du module logiciel d'application doivent être documentés.

6.11.3.7.5 Lorsque le logiciel a déjà été évalué ou si une part significative d'expérience satisfaisante de fonctionnement est disponible, la quantité des essais peut être réduite.

6.11.3.8 Exigences pour les tests d'intégration du logiciel d'application

NOTE Vérifier que le logiciel est intégré de façon satisfaisante constitue une activité de vérification.

6.11.3.8.1 Les tests du logiciel d'application doivent vérifier que tous les modules du logiciel d'application et les composants/sous-systèmes interagissent correctement entre eux ainsi qu'avec le logiciel intégré de base pour réaliser les fonctions prévues et ne pas réaliser les fonctions non prévues qui pourraient compromettre une fonction de sécurité.

6.11.3.8.2 Les résultats des tests d'intégration du logiciel d'application doivent être documentés, établissant ainsi:

– les résultats des tests; et
– si les objectifs des critères des tests sont satisfaits.

6.11.3.8.3 En cas de défaillance, les raisons de cette défaillance et l'action corrective entreprise doivent être indiquées dans la documentation des résultats du test.

6.11.3.8.4 Pendant l'intégration du logiciel d'application, toute modification ou changement du logiciel doit être soumis à une analyse d'impact sur la sécurité, laquelle doit déterminer:

– tous les modules logiciels touchés; et
– toutes les activités nécessaires de reconception et revérification.

6.12 Intégration et test du système de commande électrique relatif à la sécurité

NOTE L'intégration d'un SRECS est habituellement effectuée préalablement à l'installation mais, dans certains cas, cette intégration ne peut être effectuée qu'après installation (par exemple, lorsque le développement du logiciel d'application n'est finalisé qu'après installation).

6.12.1 Exigences générales

6.12.1.1 Le SRECS doit être intégré conformément à la conception SRECS spécifiée. Dans le cadre de l'intégration de tous les sous-systèmes et éléments de sous-systèmes dans le SRECS, le SRECS doit être soumis aux tests conformément aux tests d'intégration spécifiés. Ces tests doivent vérifier que tous les modules interagissent de manière correcte pour remplir leurs fonctions prévues et ne pas réaliser de fonctions non prévues.

6.12.1.2 L'intégration du logiciel d'application relatif à la sécurité dans le SRECS doit comprendre des tests qui sont spécifiés lors de la phase de conception et développement afin de s'assurer de la compatibilité du logiciel d'application avec le matériel et la plateforme logicielle intégrée de façon que les exigences de fonctionnement fonctionnelles et de sécurité soient satisfaites.

NOTE 1 Ceci n'implique pas de soumettre aux tests toutes les combinaisons d'entrée. Il peut être suffisant de soumettre aux tests toutes les classes d'équivalence (voir B.5 et C.5.7 de la CEI 61508-7). Il est admis que le nombre de cas de tests puisse être réduit à un niveau acceptable par une analyse statique, analyse dynamique ou une analyse des défaillances. L'utilisation d'une conception ou de méthodes semi-formelles structurées peut faciliter les tests et la vérification.

NOTE 2 L'utilisation d'une conception ou de méthodes semi-formelles structurées peut permettre de réduire le nombre et l'ampleur des tests élémentaires.

NOTE 3 Il est également admis d'utiliser des preuves statistiques pour réduire le nombre et l'ampleur des tests élémentaires.

6.12.1.3 La documentation appropriée des tests d'intégration du SRECS doit être produite, en indiquant les résultats des tests et en précisant la conformité aux objectifs et critères spécifiés pendant la phase de conception et de développement. En cas de défaillance, les raisons de cette défaillance doivent être documentées, des actions correctives doivent être entreprises et un nouveau test effectué.

6.12.1.4 Pendant les tests d'intégration, toute modification ou changement effectué sur le SRECS doit faire l'objet d'une analyse d'impact qui doit identifier tous les composants touchés ainsi que les activités de revérification nécessaires.

6.12.1.5 Lors des tests d'intégration d'un SRECS, doivent être documentés ce qui suit:

a) la version de la spécification de test utilisée;

b) les critères d'acceptation des tests d'intégration;

c) la version du SRECS soumis aux tests;

d) les outils et les équipements utilisés ainsi que les données d'étalonnage;

e) les résultats de chaque test;

f) toute divergence entre les résultats prévus et réels;

g) l'analyse effectuée ainsi que les décisions prises quant à la poursuite du test ou l'émission d'une demande de modification dans le cas où des divergences seraient observées.

6.12.2 Tests pour déterminer l'intégrité de sécurité systématique lors de l'intégration d'un SRECS

6.12.2.1 Des tests pour révéler les anomalies et éviter les défaillances lors de l'intégration du logiciel d'application et du matériel doivent être réalisés. Lors des tests, des revues doivent être effectuées afin de déterminer si les caractéristiques du SRECS sont réalisées.

6.12.2.2 Les tests suivants doivent s'appliquer:

a) des tests fonctionnels dans lesquels des données caractérisant le fonctionnement sont appliquées au SRECS. Les sorties doivent être observées et leurs réponses sont comparées avec celles indiquées dans la spécification. Les divergences par rapport à la spécification et les indices d'une spécification incomplète doivent être documentés; et

b) des tests dynamiques afin de vérifier le comportement dynamique en conditions réelles de fonctionnement, révéler les défaillances pour satisfaire à la spécification fonctionnelle du SRECS, et évaluer l'utilité et la robustesse du SRECS.

NOTE Les fonctions d'un système ou d'un programme sont exécutées dans un environnement spécifié avec des données de test spécifiées, déduites systématiquement de la SRS du SRECS conformément aux critères établis. Cela met en évidence le comportement du SRECS et permet une comparaison avec la spécification. Le but est de déterminer si le SRECS et/ou ses sous-systèmes effectuent correctement toutes les fonctions exigées par la spécification. La technique consistant à former des classes d'équivalence est un exemple des critères pour définir des données de test «boîte noire». L'ensemble des données d'entrée est subdivisé en des fourchettes de valeurs d'entrée (classes d'équivalence) à l'aide de la spécification. Les tests élémentaires sont donc formés à partir:

- des données dans les fourchettes permises;

- des données en dehors des fourchettes permises;

- des données aux limites des fourchettes;

- des valeurs extrêmes;

- et des combinaisons des classes ci-dessus.

D'autres critères peuvent être efficaces pour choisir les tests élémentaires dans les différentes activités de test (test du module, test d'intégration et test du système).

6.13 Installation d'un SRECS

6.13.1 Objectifs

Les objectifs des exigences de cet article concernent l'installation d'un SRECS afin de s'assurer qu'il est approprié à l'usage prévu et qu'il est prêt pour la validation.

6.13.2 Exigences

6.13.2.1 Un SRECS doit être installé conformément au plan de sécurité fonctionnel en vue de la dernière validation du système (voir 4.2.1h).

6.13.2.2 Des enregistrements appropriés de l'installation du SRECS doivent être produits, établissant tous les résultats de tests. En cas de défaillance, on doit enregistrer les raisons de cette défaillance.

7 Informations pour l'utilisation du SRECS

7.1 Objectifs

Des informations concernant le SRECS doivent être fournies à l'utilisateur afin de lui permettre de développer des procédures pour s'assurer que la sécurité fonctionnelle prescrite du SRECS est maintenue pendant l'utilisation et l'entretien de la machine.

7.2 Documentation pour l'installation, l'utilisation et l'entretien

NOTE 1 Voir aussi l'Article 6 de l'ISO 12100-2 qui donne des informations générales qu'il convient de prendre en considération lors de la rédaction des documents d'accompagnement.

NOTE 2 Un ou plusieurs éléments de la documentation décrite dans le présent paragraphe peuvent avoir été développés afin de satisfaire à d'autres aspects de la présente norme.