IEC 62566

Edition 1.0 2012-01

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Nuclear power plants – Instrumentation and control important to safety – Development of HDL-programmed integrated circuits for systems performing category A functions

Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Développement des circuits intégrés programmés en HDL pour les systèmes réalisant des fonctions de catégorie A

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

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC

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

About IEC publications

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

Useful links:

IEC publications search - www.iec.ch/searchpub

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It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

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

Electropedia - www.electropedia.org

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

Customer Service Centre - webstore.iec.ch/csc

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

A propos de la CEI

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

A propos des publications CEI

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

Liens utiles:

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

La recherche avancée vous permet de trouver des publications CEI en utilisant différents critères (numéro de référence, texte, comité d'études,…).
Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

Just Published CEI - webstore.iec.ch/justpublished

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

Electropedia - www.electropedia.org

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

Service Clients - webstore.iec.ch/csc

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

IEC 62566

Edition 1.0 2012-01

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Nuclear power plants – Instrumentation and control important to safety – Development of HDL-programmed integrated circuits for systems performing category A functions

Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Développement des circuits intégrés programmés en HDL pour les systèmes réalisant des fonctions de catégorie A

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX XA

ICS 27.120.20

ISBN 978-2-88912-896-9

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

NUCLEAR POWER PLANTS – INSTRUMENTATION AND CONTROL IMPORTANT TO SAFETY – DEVELOPMENT OF HDL-PROGRAMMED INTEGRATED CIRCUITS FOR SYSTEMS PERFORMING CATEGORY A FUNCTIONS

FOREWORD

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

International Standard IEC 62566 has been prepared by subcommittee 45A: Instrumentation and control of nuclear facilities, of IEC technical committee 45: Nuclear instrumentation.

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

FDIS	Report on voting
45A/859/FDIS	45A/865/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 stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

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

INTRODUCTION

a) Technical background, main issues and organisation of the Standard

The electronic systems of class 1 (according to IEC 61513) used in Nuclear Power Plants (NPP) which are required in emergency situations, need to be fully validated and qualified before being used in operation.

In traditional systems that are computer-based, a separation can be drawn between the hardware and software portions. The hardware is mainly designed with standardised components having pre-defined electronic functions such as microprocessors, timers or network controllers, whereas software is used to coordinate the different parts of the hardware and to implement the application functions.

Nowadays, I&C designers may build application functions directly in one integrated circuit using devices such as FPGAs or similar technologies. The function of such an integrated circuit is not defined by the supplier of the physical component or micro-electronic technology but by the I&C designer.

The specific integrated circuits addressed by this Standard are:

1) based on pre-developed micro-electronic resources,
2) developed within an I&C project,
3) developed with Hardware Description Languages (HDL) and related tools used to implement the requirements in a proper assembly of the pre-developed micro-electronic resources.

Therefore these circuits are named “HDL-Programmed Devices”, (HPD). The HDL statements which describe a HPD can include the instantiation of Pre-Developed Blocks (PDB) which are typically provided as libraries, macros, or Intellectual Property cores.

HPDs can be effective solutions to implement functions required by an I&C project. However, the verification and validation may be limited by issues such as high number of internal paths and limited observability, if the HPD has not been developed with verifiability in mind.

In order to achieve the reliability required for safety I&C systems, the development of HPDs shall comply with strict process and technical requirements such as those provided by this Standard, including the specification of requirements, the selection of blank integrated circuits and PDBs, the design and implementation, the verification, and the procedures for operation and maintenance.

It is intended that this Standard be used by hardware designers, operators of NPPs (utilities), and by regulators. Regulatory bodies will find guidance to assess important aspects such as design, implementation, verification and validation of HPDs.

b) Situation of the current Standard in the structure of the IEC SC 45A Standard series

IEC 61513 is a first level IEC SC 45A document and gives guidance applicable to I&C at system level. It is supplemented by guidance at hardware level (IEC 60987) and software level (IEC 60880 and IEC 62138). IEC 62340 gives requirements in order to reduce and overcome the possibility of common cause failure of category A functions.

IEC 62566 is a second level IEC SC 45A document which focuses on the activities when HPDs are developed. It complements IEC 60987 which deals with the generic issues of hardware design of computer based systems. It refers to IEC 60880 when issues identical to that of software development are addressed.

For more details on the structure of the IEC SC 45A Standard series, see item d) of this introduction.

c) Recommendations and limitations regarding the application of the Standard

It is important to note that this Standard establishes no additional functional requirements for safety systems.

Aspects for which special requirements and recommendations have been produced are:

1) an approach to specify the requirements of, to design, to implement and to verify “HDL-Programmed Devices” (HPD, 3.7), and to handle the corresponding aspects of system integration and validation;
2) an approach to analyse and select the blank integrated circuits, micro-electronic technologies and Pre-Developed Blocks (PDB, 3.11) used to develop HPDs;
3) procedures for the modification and configuration control of HPDs;
4) requirements for selection and use of software tools used to develop HPDs.

It is recognized that digital technology is continuing to develop at a rapid pace and that it is not possible for a Standard such as this one to include references to all modern design technologies and techniques.

To ensure that the Standard will continue to be relevant in future years the emphasis has been placed on issues of principle, rather than specific technologies. If new techniques are developed then it should be possible to assess the suitability of such techniques by applying the safety principles contained within this Standard.

d) Description of the structure of the IEC SC 45A Standard series and relationships with other IEC documents and other bodies documents (IAEA, ISO)

The top-level document of the IEC SC 45A Standard series is IEC 61513. It provides general requirements for I&C systems and equipment that are used to perform functions important to safety in NPPs. IEC 61513 structures the IEC SC 45A Standard series.

IEC 61513 refers directly to other IEC SC 45A Standards for general topics related to categorization of functions and classification of systems, qualification, separation of systems, defence against common cause failure, software aspects of computer-based systems, hardware aspects of computer-based systems, and control room design. The Standards referenced directly at this second level should be considered together with IEC 61513 as a consistent document set.

At a third level, IEC SC 45A Standards not directly referenced by IEC 61513 are Standards related to specific equipment, technical methods, or specific activities. Usually these documents, which make reference to second-level documents for general topics, can be used on their own.

A fourth level extending the IEC SC 45 Standard series, corresponds to the Technical Reports which are not normative.

IEC 61513 has adopted a presentation format similar to the basic safety publication IEC 61508 with an overall safety life-cycle framework and a system life-cycle framework and provides an interpretation of the general requirements of IEC 61508-1, IEC 61508-2 and IEC 61508-4, for the nuclear application sector. Compliance with IEC 61513 will facilitate consistency with the requirements of IEC 61508 as they have been interpreted for the nuclear industry. In this framework IEC 60880 and IEC 62138 correspond to IEC 61508-3 for the nuclear application sector.

IEC 61513 refers to ISO as well as to IAEA GS-R-3 and IAEA GS-G-3.1 for topics related to quality assurance.

The IEC SC 45A Standards series consistently implements and details the principles and basic safety aspects provided in the IAEA code on the safety of NPPs and in the IAEA safety series, in particular the Requirements NS-R-1, establishing safety requirements related to the design of Nuclear Power Plants, and the Safety Guide NS-G-1.3 dealing with instrumentation and control systems important to safety in Nuclear Power Plants. The terminology and definitions used by SC 45A Standards are consistent with those used by the IAEA.

NUCLEAR POWER PLANTS – INSTRUMENTATION AND CONTROL IMPORTANT TO SAFETY – DEVELOPMENT OF HDL-PROGRAMMED INTEGRATED CIRCUITS FOR SYSTEMS PERFORMING CATEGORY A FUNCTIONS

1 Scope and object

1.1 General

This International Standard provides requirements for achieving highly reliable “HDL-Programmed Devices” (HPD), for use in I&C systems of nuclear power plants performing functions of safety category A as defined by IEC 61226.

The programming of HPDs relies on Hardware Description Languages (HDL) and related software tools. They are typically based on blank FPGAs or similar micro-electronic technologies. General purpose integrated circuits such as microprocessors are not HPDs.

This Standard provides requirements on:

a) a dedicated development life-cycle addressing each phase of the development of HPDs, including specification of requirements, design, implementation, verification, integration and validation,

b) planning and complementary activities such as modification and production,

c) selection of pre-developed components. This includes micro-electronic resources (such as a blank FPGA or CPLD) and HDL statements representing Pre-Developed Blocks (PDB),

d) use of simplicity and deterministic principles, recognized to be of primary importance to achieve “fault free” implementation of category A functions,

e) tools used to design, implement and verify HPDs.

This Standard does not put requirements on the development of the micro-electronic resources, which are usually available as "commercial off-the-shelf" items and are not developed under nuclear quality assurance Standards. It addresses the developments made with these micro-electronic resources in an I&C project with HDLs and related tools.

This Standard provides guidance to avoid as far as possible latent faults remaining in HPDs, and to reduce the susceptibility to single failures as well as to potential Common Cause Failures (CCF). The requirements within this Standard for clear and comprehensive documentation should facilitate the effective application of IEC 62340.

Reliability aspects related to environmental qualification and failures due to ageing or physical degradation are not handled in this Standard. Other Standards, especially IEC 60987, IEC 60780 and IEC 62342, address these topics.

Subclause 5.7 of IEC 60880:2006 provides security requirements that apply to the development of HPDs as applicable.

1.2 Use of this Standard

This Standard provides guidance and requirements to produce verifiable designs and implementations where justification is necessary due for example to the function performed or to the importance to safety of its behaviour. Class 1 I&C systems may use HPDs for which full demonstration of compliance with the requirements of this Standard is not mandatory, e.g.

when they do not implement the logic of a safety function. However, deviations from this Standard should be justified.

This Standard describes the activities to develop HPDs, organized in the framework of a dedicated life-cycle. It also describes activities and guidelines to be used in addition to the requirements of IEC 61513 for system integration and validation when HPDs are included.

Those requirements of IEC 60987 that relate to programmable logic device development are applicable, in addition to those of this Standard, where HPDs are part of class 1 I&C systems.

NOTE In case of conflicting requirements, this Standard supersedes those in IEC 60987 about class 1 HPDs.

2 Normative references

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

IEC 60671, *Nuclear power plants – Instrumentation and control systems important to safety – Surveillance testing*

IEC 60880:2006, *Nuclear power plants – Instrumentation and control systems important to safety – Software aspects for computer-based systems performing category A functions*

IEC 60987:2007, *Nuclear power plants – Instrumentation and control important to safety – Hardware design requirements for computer-based systems*

IEC 61513:2011, *Nuclear power plants – Instrumentation and control important to safety – General requirements for systems*

IEC 62138, *Nuclear power plants – Instrumentation and control important for safety – Software aspects for computer-based systems performing category B or C functions*

IEC 62340, *Nuclear power plants – Instrumentation and control systems important to safety – Requirements for coping with common cause failure (CCF)*

IAEA guide NS-G-1.3:2002, *Instrumentation and Control Systems Important to Safety in Nuclear Power Plants*

3 Terms and definitions

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

3.1
Application Specific Integrated Circuit , ASIC
integrated circuit designed for specific applications

[IEC 60050-521:2002, 521-11-18]

NOTE Specialized integrated circuit designed for the purpose of one company. It embeds bespoke functions defined by this company.

3.2
block
one of the parts that make up a design; a block may be subdivided into other blocks

NOTE A block is either a Pre-Developed Block or a Native Block or a block developed during the considered project.

3.3
Common Cause Failure, CCF
failure of two or more structures, systems or components due to a single specific event or cause

[IAEA Safety Glossary 2007 Edition]

NOTE Common causes may be internal or external to an I&C system.

[IEC 61513]

3.4
Electronic System Level, ESL
high-level description of an electronic system, based on a set of processes representing functionalities of components such as microprocessors, memories, specialized computing units, or communication channels

NOTE This description allows the designer to partition the system into components, to assess its performance under different mapping of functions to the components, and to establish the requirements for the components.

It is typically performed with languages such as SystemC (IEEE 1666), SystemVerilog (IEEE 1800), or Matlab (R).

3.5
Field Programmable Gate Array, FPGA
integrated circuit that can be programmed in the field by the I&C producer. It includes programmable logic blocks (combinatorial and sequential), programmable interconnections between them and programmable blocks for input and/or outputs. The function is then defined by the I&C designer, not by the integrated circuit supplier.

NOTE While FPGAs are essentially digital devices, some of them may integrate analog input/outputs and analog to digital converters. FPGAs may include advanced digital functions such as hardware multipliers, dedicated memory and embedded processor cores.

3.6
Hardware Description Language, HDL
language used to formally describe the functions and/or the structure of an electronic component for documentation, simulation or synthesis

NOTE The most widely used HDLs are VHDL (IEEE 1076) and Verilog (IEEE 1364).

3.7
HDL-Programmed Device, HPD
integrated circuit configured (for NPP I&C systems), with Hardware Description Languages and related software tools

NOTE 1 HDLs and related tools (e.g. simulator, synthesizer) are used to implement the requirements in a proper assembly of pre-developed micro-electronic resources.

NOTE 2 The development of HPDs can use Pre-Developed Blocks.

NOTE 3 HPDs are typically based on blank FPGAs, PLDs or similar micro-electronic technologies.

3.8
module
one of the parts that make up a design; a module may be subdivided into other modules

NOTE Module" is a synonym of "Block"; "Block" is often used in the context of electronic design. "Module" is the term used by IEC 60880 and is needed in this Standard for references to IEC 60880.

3.9
native block
a Block which represents a pre-existing resource in the integrated circuit, e.g. an OR gate or a more complex block such as a multiplier or a serial transmission controller. By programming the HPD, the Native Blocks are configured and connected to provide the required function.

3.10
netlist
description of an electronic component in terms of interconnections between its terminal elements (e.g. Native Blocks)

3.11
Pre-Developed Block, PDB
pre-developed functional block usable in a HDL description

NOTE 1 PDBs are typically provided as libraries, macros, or Intellectual Property cores. They are used in the development of a HPD and incorporated in this HPD.

NOTE 2 A PDB may need significant work before incorporation in a HPD, e.g. synthesizing an electronic circuit from the HDL statements, mapping the notional components of this circuit on the hardware structures of the physical integrated circuit and routing the interconnections.

3.12
Pre-Developed Software, PDS
software part that already exists, is available as a commercial or proprietary product, and is being considered for use

[IEC 60880]

3.13
Programmable Logic Device, PLD
integrated circuit that consists of logic elements with an interconnection pattern, parts of which are user programmable.

[IEC 60050-521:2002, 521-11-01]

NOTE 1 Different kinds of PLD exist, e.g. Erasable PLD or Complex PLD (CPLD).

NOTE 2 The differences between “FPGA” and “PLD” are not well defined, but “PLD” usually refers to a simpler device than “FPGA”.

3.14
Register Transfer Level, RTL
synchronous parallel model of an electronic circuit, describing its behaviour by means of signals processed according to a combinatorial logic and transferred between registers on clock pulses. The RTL model is typically written in HDL or generated out of HDL source code.

4 Symbols and abbreviations

ASIC: Application Specific Integrated Circuit
CCF: Common Cause Failure
CPLD: Complex Programmable Logic Device
DRC: Design Rule Check
ESL: Electronic System Level
FPGA: Field Programmable Gate Array
HDL: Hardware Description Language
HPD: HDL-Programmed Device
IP: Intellectual Property

I&C: Instrumentation and Control
PAL: Programmable Array Logic
PDB: Pre-Developed Block
PDS: Pre-Developed Software
PLD: Programmable Logic Device
RAM: Random Access Memory
RTL: Register Transfer Level
SEU: Single Event Upset
SRAM: Static RAM
STA: Static Timing Analysis
VHDL: Very High Speed Integrated Circuit Hardware Description Language
V&V: Verification and Validation

5 General requirements for HPD projects

5.1 General

This clause first locates the HPD within the I&C system described by IEC 61513. Then it describes the HPD development life-cycle which structures the HPD project.

Finally it provides requirements for HPD projects, for quality assurance and for configuration management. As these issues are common with those of software development processes, the requirements are defined by reference to relevant sections of IEC 60880, supplemented by HPD specific requirements if needed.

With reference to Clause 1, the scope of this Standard excludes the development of micro-electronic technologies or blank integrated circuits. Therefore wordings such as “HPD development”, “HPD life-cycle”, “HPD design” or “HPD verification” refer to what is done within the I&C project, starting from these technologies or these blank integrated circuits, to produce the specific integrated circuit for use in the I&C system.

5.2 Life-cycle

The process of producing I&C systems for use in nuclear power plants is given in IEC 61513 that introduces the concept of system life-cycle. This is a vehicle by which the development process can be controlled and whose adoption should also result in the evidence necessary to justify the correct operation of safety systems. It includes and places requirements on, but does not dictate the project arrangements to be used for, production of systems (see Figure 1).

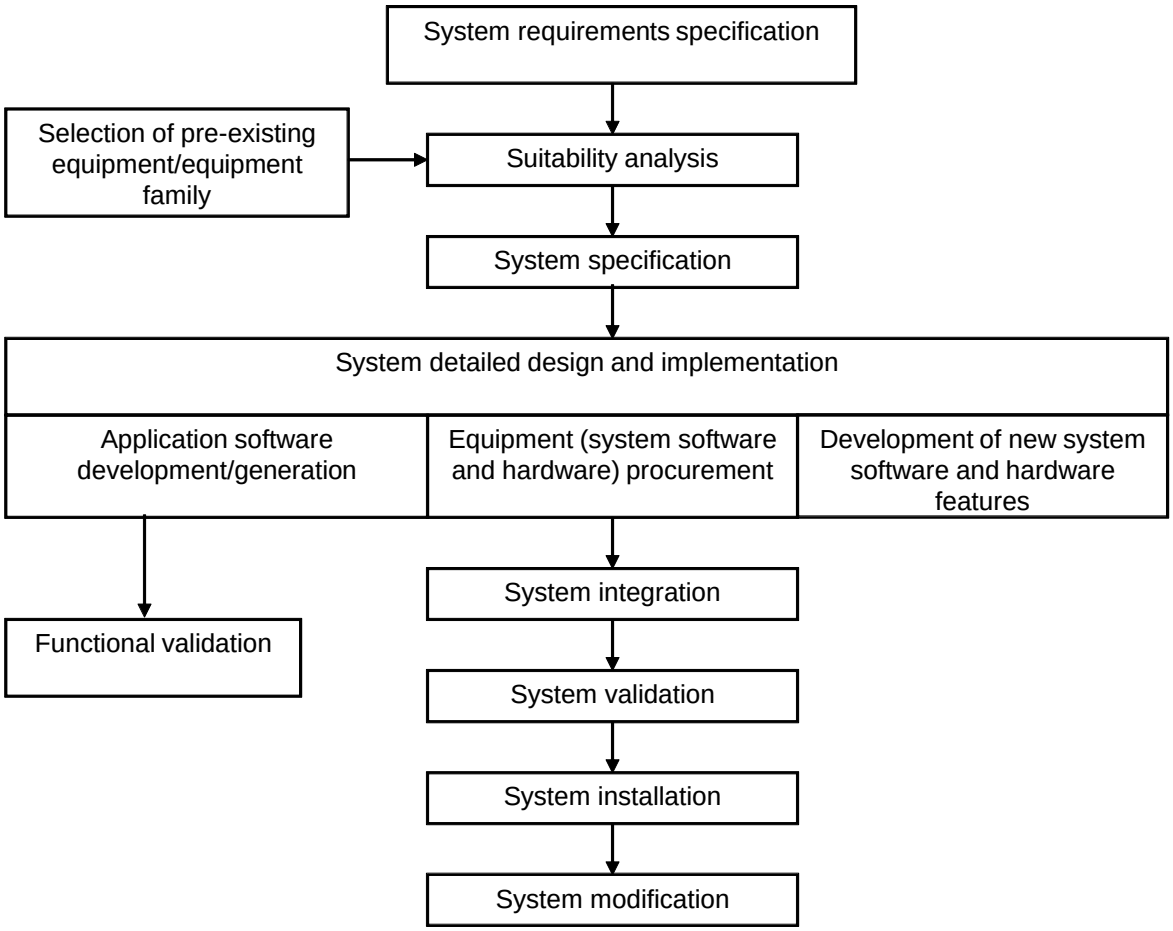

Figure 1 – System life-cycle (informative, as defined by IEC 61513)

The system life-cycle of IEC 61513 is complemented in IEC 60880 (for category A functions) and IEC 62138 (for category B and C functions) for software development and in IEC 60987 for hardware development of computer-based systems. The requirements of this Standard apply to the development of HPDs in class 1 systems, in addition to the requirements of IEC 60987.

NOTE In case of conflicting requirements, this Standard supersedes those in IEC 60987 about class 1 HPDs.

HPDs are developed by means of computer tools which tend to structure the development according to a cycle that includes activities dedicated to requirement capture, design and implementation, integration and validation, together with verification and test activities.

The system design and implementation phases of IEC 61513 shown in Figure 1 (particularly the “Equipment (system software and hardware) procurement” and “Development of new system software and hardware features”) are essential parts of the system life-cycle of IEC 61513. These phases are expanded in Figure 2 to illustrate in more detail the phases between the specification of requirements and the validation for the system components which are HPDs.

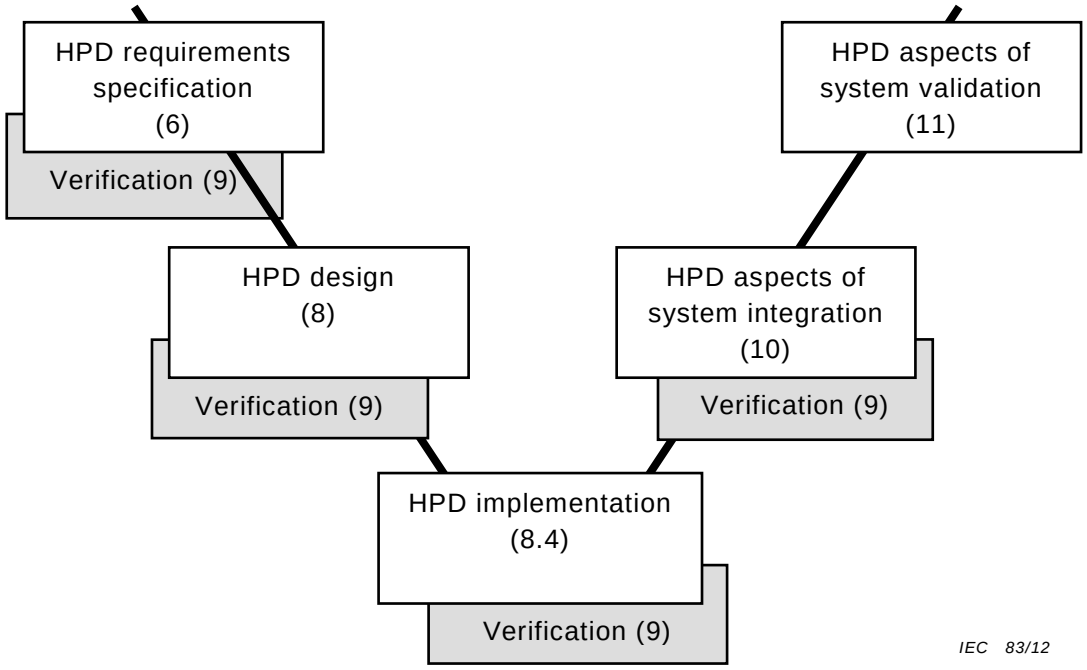

Figure 2 – Development life-cycle of HPD

Designers generally use pre-developed items such as programmable blank integrated circuits or Pre-Developed Blocks (PDB) to build integrated circuits specifically tailored to the needs of the project. The activities dedicated to the selection of these pre-developed items are addressed in Clause 7; they may be performed in parallel with the first phases of the life-cycle described in Figure 2, provided all dependencies are formally managed and documented.

The life-cycle described in Figure 2 shows the development life-cycle of one HPD that may be undertaken in parallel with the development of other components (software or hardware) of the system as shown in Figure 1, but coming together at the integration and validation phases of the system life-cycle.

The approach proposed to development is based on the traditional “V cycle” model as this approach has been reflected in other Standards and is also recommended in IAEA NS-G 1.3, but allowing necessary adjustments recognizing that some phases of the development can be done automatically by tools and that development may be iterative.

There is often no clear separation and well-identified boundary between the integration of a given component and the system integration. Therefore, in this Standard, the integration of a HPD is considered to be part of the system integration. Similarly, the validation of the HPD is considered to be part of the system validation.

Depending on the function achieved by the HPD, the system or subsystem to consider during integration may range:

1) from the I&C system when the HPD implements a safety function logic,
2) to an electronic board or cabinet when it implements a function (internal to the board or cabinet) that has been demonstrated, by suitable analysis, to be incapable of affecting the outputs of any safety function in the wider system.

The situation usually most critical from a safety standpoint is when the HPD directly implements the safety function logic.

The following activities support the development process of HPDs:

a) project management (5.3),

b) quality assurance and quality control (5.4),

c) configuration management (5.5),

d) verification (Clause 9).

There are also activities involving selection of tools to support the development (Clause 15), the production of documentation (Annex A) and modification (Clause 12).

5.3 HPD project management

5.3.1 General

5.3.1.1 Each HPD shall be developed within a dedicated HPD project.

5.3.1.2 The HPD project shall comply with the requirements of 5.4 of IEC 60880:2006 (by replacing “software” with “HPD”).

NOTE 1 A typical list of the documents required through the life-cycle is given in Annex A of this Standard.

NOTE 2 The documented inputs addressed by 5.4.6 of IEC 60880:2006 include parameters for the automated activities of the software tools (e.g.: optimize timing, optimize density, etc.).

5.3.1.3 The development process may be iterative; a phase may start before the activities of the preceding phase are complete; however, a phase shall only be terminated if the preceding phases have been completed and if its outputs are consistent with the inputs provided by these preceding activities.

5.3.1.4 The phases of the HPD project shall include the specification of requirements, the design and the implementation of the HPD.

5.3.2 Additional requirements

5.3.2.1 The selection of the pre-developed items used by the project shall be performed according to the requirements of Clause 7 of this Standard.

5.3.2.2 Transition criteria between phases shall be defined.

5.3.2.3 The criteria for phase termination shall have methodological and technical content, involving enough detail such that their evaluation requires an in-depth analysis of the phase outputs.

5.3.2.4 The documentation (5.4.11 of IEC 60880:2006) shall include the description of the functions performed by the HPD and its interface.

5.4 HPD quality assurance plan

A quality assurance plan for the HPD shall exist and shall comply with the requirements of 5.5 of IEC 60880:2006 (by replacing “software” with “HPD”).

NOTE In this context, “language” means “computer language”.

5.5 Configuration management

5.5.1 Configuration management of the HPD shall be performed according to the requirements of 5.6 of IEC 60880:2006 (by replacing “software” with “HPD”).

NOTE The segregation required by 5.6.6 of IEC 60880:2006 applies to the documentation and computer files used or produced by the HPD project.

5.5.2 The configuration management shall record the following items:

a) documentation of modules (blocks) developed within the project and of PDBs,

b) identification marking of integrated circuits,

c) computer files used for simulation, verification and production,

d) parameters used for the automated activities of the software tools (see Clause 15), such as “optimize timing, optimize density” for the Place and Route activity,

e) identification of the versions of all software tools (see Clause 15), including any “software patch” applied, as well as general purpose libraries and technology dependent libraries.

6 HPD requirements specification

6.1 General

6.1.1 A requirement specification shall document the requirements of the HPD, either in the document itself or by referencing sets of requirements stated at system or subsystem level (e.g. the functional behaviour to be implemented).

6.1.2 The requirement specification shall be understandable for all participants, including hardware engineers and people mentioned in 6.6.

6.1.3 The requirements specification shall be unequivocal, verifiable and achievable, including for temporal aspects.

6.1.4 When the HPD implements a safety function, its requirement specification shall be derived from the requirements of the I&C system implementing this safety function and shall be part of the specification of the subsystem which uses the HPD.

6.1.5 The requirement specification shall describe what is to be done and not how it is to be done.

6.1.6 A documented, formal and auditable process shall be defined and implemented for the establishment of the requirements specification.

6.1.7 The requirement specification shall be such that compliance with the requirement specification of the I&C system can be verified. If the HPD is used by a subsystem of the I&C system, it shall also be possible to verify the compliance with the system design specifications.

6.1.8 The requirement specification shall consider all plant operating conditions down to the HPD level for the functions that are impacted.

6.1.9 Interface requirements with other systems or components shall be addressed according to IEC 61513.

6.1.10 Interface requirements with other systems or components shall be documented.

6.1.11 When they are not part of the HPD requirements but result from HPD design decisions, the following interface requirements shall be documented:

a) electrical and temporal performance (e.g. input load, setup and hold time of inputs, operating frequency, fan-out, propagation time from any input to the associated outputs),

b) profiles of interfaced signal and power supplies,

c) power dissipation, operating temperature and cooling requirements.

6.2 Functional aspects of the requirement specification

This subclause describes the content of the requirements specification directly related to the functional needs. Subclauses 6.3 and 6.4 address additional aspects to be included in the Requirements Specification.

The requirement specification shall specify:

a) the functions to be provided by the HPD,

b) the HPD’s different modes, and the corresponding conditions of transition, including power-on and initialization,

c) the HPD's interfaces and interactions with its environment (operators and other I&C components), including the roles, protocols, types, data formats, bit numbering, ranges and constraints of inputs and outputs,

d) any HPD parameters which can be modified manually during operation, and their roles,

e) the HPD's performance, in particular response time,

f) what the HPD must not do or must avoid, when appropriate,

g) any assumptions regarding the HPD's environment (e.g. electrical and temporal characteristics of inputs-outputs, power supplies, specific profiles during power-on, cooling).

6.3 Deterministic design

The requirement specification shall specify that the function of the HPD is deterministic by design. This means that any given input sequence fulfilling the electrical and temporal specification always produces the same outputs.

NOTE Modern FPGA and other integrated circuits covered by this Standard can contain analogue functional blocks (e.g. analogue to digital converter) that are subject to electronic noise, digitisation error, etc. Variations in the response of these analogue functional blocks due to these causes, as well as their impacts on the response of the HPD, are not breaches in the deterministic design.

6.4 Fault detection and fault tolerance

The requirements of IEC 60987 subclauses 5.3 and 5.4 addressing the reliability with respect to random failures and the environmental withstand apply. This includes the faults due to SEU (single event upset) and neutron/alpha radiation when relevant.

Defensive design is typically based on a combination of techniques (e.g. redundancy, vote, parity and cyclic redundancy checks, watch-dog, range and plausibility checks).

6.4.1 The requirement specification shall specify requirements for defensive design to address fault detection and fault tolerance.

6.4.2 The benefit from defensive design measures should be balanced with their induced additional complexity. The overall objective is to take into account the testability of the HPD during design and implementation, using internal and external detection means to achieve high fault coverage.

6.4.3 The requirement specification shall describe the provisions to detect HPD malfunctions, taking into account the provisions already taken at subsystem or system level.

6.4.4 These provisions may need the HPD to provide additional outputs, either to be used by an external mechanism such as a watch-dog or to achieve the coverage of the supervision made by an external testing device.

6.4.5 The defensive design should allow the detection of erroneous behaviour (such as data corruption or deviation from specified processing algorithm, or deviation from specified operating conditions), erroneous data transmission between processing units, unintended modification of memories or configuration data.

6.4.6 The defensive design shall not have adverse influence on the I&C system functions, nor prevent the HPD from meeting its response time specification.

6.4.7 The requirement specification shall describe the expected logical and temporal behaviour (such as output values and specific information issued) when a fault is detected.

6.4.8 This behaviour shall comply with the system behaviour required by the system specification and with IEC 61513 system design requirements.

6.4.9 The requirement specification shall specify and justify the target coverage of the fault detection to be achieved by defensive design.

6.5 Requirements capture using Electronic System Level tools

6.5.1 General

This Standard does not prescribe a specific method to capture the HPD requirements. If they are captured using tools at Electronic System Level (ESL, see Clause B.1), then the requirements of 6.5.2 and 6.5.3 apply to these tools and to their use.

In the case of ESL, as the requirements specification language may be similar to implementation languages, it may be less practical to fulfil 6.1.5. (separation between what has to be done (the requirement) and how it is done (the design)). Provisions may be needed to fulfil it, e.g. comments to specify inputs, outputs and algorithms.

6.5.2 Requirements on the formalism of tools used at ESL level

6.5.2.1 When the HPD requirements are captured using an ESL tool:

a) this tool shall offer a formalism with a rigorous semantics and clarity (standardization of structure and presentation, modularity, sound comments);

b) the formalism used in the ESL tool shall be understandable for all participants;

c) if the tool offers flexible mechanisms to redefine functions and operators, then the actual characteristics of any given element should be clear to any participant, including hardware engineers and other personnel mentioned in 6.6.

6.5.2.2 The languages used at ESL level should allow taking due account of the system architecture, e.g. enable the assignment of functions to components, and support any fault tolerant design features.

6.5.3 Interface with design tools

The semantics of the languages used to express the requirement specification at ESL level may differ from the semantics of the HDL languages used during design. Examples where discrepancies may occur are in the interpretation of parallelism, the management of overflows, or the encoding of types and finite state machines.

a) If the semantics of the language used to express the requirement specification at ESL level differs from the semantics of the other languages used in the project, then discrepancies shall be identified for each involved item of the requirement specification;

b) each occurrence of a discrepancy within the requirement specification shall be documented. A generic list of discrepancies between the involved languages is a useful reference, but is not enough to clarify the Requirement Specification.

6.6 Requirements analysis and review

6.6.1 A critical analysis of the requirements shall be performed and documented, in order to find potential inconsistencies, omissions and ambiguities.

6.6.2 The scope of this analysis shall cover functional requirements and all other types of requirements, including those addressing abnormal behaviour such as unexpected input values or sequences.

6.6.3 The requirement specification shall be reviewed to check its completeness and its consistency.

6.6.4 For safety functions implemented in the HPD, process and I&C engineers shall participate in the review, as well as specialists of subsystems or components (including software) interfaced to the HPD.

7 Acceptance process for programmable integrated circuits, native blocks and pre-developed blocks

7.1 General

When developing the HPD, it is necessary to select and assess pre-developed items such as a blank integrated circuit (including their native blocks) or PDBs incorporated in the final HPD.

As these pre-developed items (or components) may include features not required for the HPD, the elaboration and the enforcement of specific “rules of use” may be recommended in order to restrict their use to what is needed and safe.

7.2 Component requirement specification

7.2.1 General

The requirements assigned to the pre-developed items (or components) result from the initial design activities of the HPD. For example the HPD requirements could include a specific pass-band filter, which the designer could implement using a PDB performing a Fast Fourier Transform.

Thus the component requirement specification (here for a Fast Fourier Transform PDB, defined by characteristics such as type of algorithm, radix size, decimation method, silicon area needed, etc.) differs from the HPD requirement specification (here for a Pass-band filter, defined by characteristics such as corner frequencies, gain, slopes, etc.).

7.2.1.1 A component requirement specification shall document the requirements applicable to each pre-developed item: blank integrated circuit, micro-electronic resources (seen as native blocks), associated tools when relevant or PDBs.

7.2.1.2 The component requirement specification shall state all the requirements, either in the document itself or by referencing sets of requirements stated at system or subsystem level (e.g. functional behaviour to be implemented).

7.2.1.3 Being the basis of the selection and use of the pre-developed item, the component requirement specification shall thus be understandable by all participants, including hardware and software designers when relevant, as well as verifiers, reviewers, and regulators.

7.2.1.4 The component requirement specification shall be unequivocal, verifiable and achievable, including for temporal aspects.

7.2.1.5 The component requirement specification shall be such that compliance with the requirements of IEC 61513 of the I&C system using this component can be demonstrated.

7.2.2 Requirements

The component requirement specification shall specify all characteristics required from the pre-developed item, in particular those of the list provided in 6.2.

NOTE The generic names of the characteristics (e.g. “function”) are identical to those of 6.2, but the contents differ in general as explained in 7.2.

7.2.3 Requirements analysis and review

7.2.3.1 A critical analysis of the component requirement specification shall be performed and documented, in order to find potential inconsistencies, lack of completeness or ambiguities.

7.2.3.2 The scope of this analysis shall cover functional requirements and all other types of requirements, including those addressing non-nominal behaviour such as unexpected input values or sequences.

7.2.3.3 The component requirement specification shall be formally reviewed by experts of all relevant domains to check its completeness and its consistency.

7.3 Rules of use

7.3.1 If the pre-developed item includes functions or operating modes that are not required to be implemented in the HPD, rules should be defined to prohibit the use of such functions and modes.

The use of functions or modes that are required to implement the HPD may be constrained by rules in order to improve design properties such as safety or testability.

7.3.2 If rules of use are established:

a) they shall be documented,

b) the quality plan shall give assurance that their fulfilment is verified during the project.

7.4 Selection

7.4.1 General

7.4.1.1 A documented analysis of each pre-developed item used in the HPD shall demonstrate that it fulfils the requirements of its component requirement specification, possibly with rules of use and modifications (see 7.6).

7.4.1.2 A user documentation for safety shall detail how designers are to use the pre-developed item consistently with its specification and design characteristics.

7.4.2 Documentation review

Documentation review is the primary method to demonstrate that the pre-developed item fulfils the component requirement specification.

7.4.2.1 This review should be based on the documentation of the pre-developed item including documentation of its design and its verification.

7.4.2.2 The documentation shall contain sufficient detail in order to demonstrate the fulfilment of the functional, electrical and temporal requirements of the pre-developed item.

7.4.2.3 The analysis of the documentation shall demonstrate that any functions and modes of the pre-developed item not used within the HPD do not impede the used ones.

7.4.3 Operating experience review

The operating experience of the pre-developed item may be invoked to compensate for some limited documentation weaknesses regarding its reliability or its design. If the operating experience is invoked then:

a) the analysis of the operating experience shall demonstrate that:

   1) its volume is commensurate to the reliability requirements,

   2) it has been collected in operating conditions equivalent to those in which the pre-developed item will be used,

   3) the actual use of the pre-developed item has been traced at the level of detail generally required by this Standard for the documentation;

b) the means and procedures used to collect the operating experience shall ensure that any pre-developed item failure that occurred in the analysed operation is recorded in such detail that a technical analysis can identify its cause as far as possible;

c) it shall be demonstrated by analysis of the failures recorded during operation that they do not impact the functions or the safety of the HPD;

d) the operating experience –and, if needed, complementary tests- shall demonstrate that the pre-developed item fulfils its requirements;

e) a documented technical analysis shall justify that all interactions of the pre-developed item with its environment are included within those covered by the operating experience;

f) the operating experience taken into consideration shall correspond to precisely identified versions of the pre-developed item and, when this item is specific to equipment, of the equipment in which it operates;

g) the operating experience should address the specific version of the pre-developed item or its sub-part used in the HPD; otherwise the differences between versions shall be analysed to demonstrate that the operating experience is relevant for the intended version.

7.4.4 Specific requirements related to the blank integrated circuits

7.4.4.1 The following aspect shall be addressed:

a) analysis of the adequacy of the programming mechanisms and circuitry;

b) demonstration that the programming process is fault-free or that any fault in this process is detected and correctly managed;

c) demonstration that the integrated circuit retains its programmed configuration for an adequate duration;

d) analysis of the potential for faults due to the additional internal and external mechanisms or power transients and justification according to the reliability requirements.

7.4.4.2 A detailed analysis shall demonstrate that:

a) the integrated circuit will be able to fulfil its component requirement specification,

b) the associated tools

– comply with Clause 15, and

– allow all verifications required by Clauses 8 and 9 (such as Static Timing Analysis).

7.4.4.3 The data needed to calculate the fault rate (in the sense of random physical faults) shall be available and based on sufficient operating experience.

7.4.4.4 The designers who design or implement the HPD shall have appropriate knowledge:

a) of the blank integrated circuit, including programming particularities, configuration and testing modes, protocols, pins and registers, and any electrical or logical specificity, and

b) of the associated tools, native blocks and PDBs. In particular, they should be able to predict, understand and (where necessary) control the choices made by the tools during synthesis, place and route.

7.5 Acceptance justification

7.5.1 A formal review shall examine the pre-developed item analysis, including the rules of use and the arrangements taken to ensure the compliance of each physical part used in production, to decide whether or not the pre-developed item is accepted for use in the HPD.

7.5.2 If the pre-developed item is accepted, any arrangement and rule of use taken into account in the analysis shall be applicable during the whole HPD life-cycle.

7.5.3 The review team shall include experts having skills relevant to the subjects (e.g. hardware technology, software) and engineers from the teams responsible for the components that are interfaced to the pre-developed item.

7.6 Modification for acceptance

7.6.1 If modifications of the pre-developed item are necessary to achieve acceptance, they shall be specified, designed, implemented and verified before the review.

7.6.2 These modifications shall be performed and documented in accordance with the requirements of this Standard regarding project structure and management, quality, specification of requirements, design, implementation and verification.

7.7 Modification after acceptance

The acceptance activities, including the review, shall be performed again after any modification of the pre-developed item involving its design or its micro-electronic aspects.

7.8 Acceptance documentation

The acceptance documentation of the pre-developed item shall be under configuration management.

7.8.1 The documentation shall include or shall make a reference to:

a) the requirement specification of the HPD,

b) all documents issued or invoked during the analysis of the pre-developed item,

c) all documents issued during modification of the pre-developed item,

d) the review report.

The documentation shall include any information necessary to use the pre-developed item correctly, taking into account constraints from its initial specification, from the rules of use, and from the modifications.

8 HPD design and implementation

8.1 General

This clause provides requirements and recommendations based on good practice for design and implementation, in order to meet appropriate safety features such as fault-free as possible and amenability to verification.

8.1.1 The development process shall define a design phase and an implementation phase.

8.2 Hardware Description Languages (HDL) and related tools

Even though the use of specific languages and tools cannot be required, the following may be considered as common basic rules for languages and tools used for the design and implementation of HPDs for class 1 systems.

8.2.1 Design and implementation should use Hardware Description Languages (HDL) and tools for simulation, synthesis, place and route.

NOTE When properly chosen and used, these tools improve essential aspects such as understandability of the descriptions, management of electrical and temporal constraints, verification, adequateness of coverage criteria, and documentation.

8.2.2 Even if 8.2.1 is not fulfilled, any documentation, analysis, or verification required by this Standard shall be provided.

8.2.3 The language in use:

a) shall follow strict (or well-defined) semantic and syntax rules;

b) shall have a syntax completely and clearly defined and documented;

c) should comply with a recognized Standard (e.g. IEEE 1076 for VHDL or IEEE 1364 for Verilog).

8.2.4 The use of the language should be restricted to a “safe” subset where appropriate, for example be restricted to features that are needed to implement the required functions and are synthesisable with standardized libraries (e.g. avoid the use of initial values, explicit delays or division).

8.2.5 The simulator in use shall produce results strictly compliant with the documented semantic of the language.

The simulator should comply with a recognized Standard (e.g. IEEE 1076 for VHDL or IEEE 1364 for Verilog).

8.2.6 Except as addressed in 8.2.7, only tools complying with the requirements of Clause 15 shall be used for analysis, simulation, synthesis, place and route. It is not necessary for users to repeat testing of the tools if this has already been performed and documented by the supplier.

8.2.7 If a tool partially compliant with the requirements of Clause 15 is used, additional verification of the results produced by this tool (e.g. netlist produced by a synthesis tool) shall provide evidence that the results are correct. Formal equivalence checking tools are valuable in achieving an error free design.

8.3 Design

8.3.1 General

Starting from the HPD requirement specification, the design initially aims at defining major choices such as the decomposition into modules (application-specific or pre-developed), the operation of the defensive design, as well as the identification of needed micro-electronic technologies (including their native blocks) and PDBs. Then an RTL description is built using HDLs. The following requirements aim at producing a clear and verifiable design.

8.3.1.1 The design phase shall produce a) a formalized description of the HPD, e.g. RTL and b) the associated documentation.

8.3.1.2 Communication links shall be designed in compliance with the requirements on data communication given in 5.4.2.4 of IEC 61513.

8.3.1.3 The design should allow easy verification.

8.3.1.4 Non-compliances with design rules should be justified.

8.3.2 Defensive design

8.3.2.1 When a selected native block or PDB (see Clause 7) is a processor core, it should support the requirements of IEC 60880 for self-supervision.

8.3.2.2 The design shall take into account the arrangements selected in the requirement specification to detect the faults and to elaborate the corresponding information within the HPD.

8.3.2.3 On fault detection, the HPD shall behave in accordance with the corresponding specified requirements.

8.3.3 Structure

8.3.3.1 A top down approach to design should be preferred to a bottom up approach.

NOTE Library items are the ultimate targets of the design. Therefore the use of libraries fulfilling the requirements of Clauses 7 and 15 is in line with the top-down approach and is recommended.

8.3.3.2 The structure of the design should be based on decomposition into modules. Related modules may be contained in a library.

8.3.3.3 Generic modules should be contained in libraries.

8.3.3.4 The structure should be simple and easy to understand, both in its overall design and in its details.

8.3.3.5 A conceptual model of the architecture should be generated at the beginning of the project.

8.3.4 Language and coding rules

8.3.4.1 In order to facilitate a stable and reliable design, proven design methodology and general good practice should be used.

8.3.4.2 In order to make the design more understandable and to reduce the potential for differences between the simulated and the synthesized behaviours:

a) a set of strict design rules which reflect the latest knowledge in terms of design safety and reliability shall be required by the quality plan and established;

b) the compliance with those design rules shall be enforced by appropriate means (e.g. review, tooling, etc.).

8.3.4.3 The list given below contains strongly recommended design considerations and techniques. However the list is not considered to be all-encompassing and parts may change with technology. Nevertheless any non-compliance with the rules in the following list shall be justified and taken into account in failure analysis:

a) only synthesizable features of the language should be used in the design of the HPD. The test and simulation environment (9.5) may use all language features. Any native blocks (see 3.9) which are already synthesized and routed in the pre-developed integrated circuit may be instantiated as they are, if they comply with Clause 7;

b) dedicated resources or design features (e.g. predefined clock trees and clock conditioning circuits, power rails, reset trees, etc.) should be used where appropriate;

c) coding rules should cover all relevant aspects, in particular naming of modules and signals, use of the structuring features (such as packages, functions, procedures, project libraries, instantiation), organization of the computations on critical paths, organization of processes, recommended constructs, forbidden constructs;

d) functions using side effects ("impure") should be forbidden in the design description. (Rationale: such a function can return different values when called several times with the same parameters. It is therefore very difficult to test and verify, as it breaks the basic concept of a function, and in fact of determinism);

NOTE 1 An impure function may also have side effects such as modifying objects out of their scope.

e) constructs that could lead to differences between simulated and synthesized behaviours should be forbidden. Depending on the language used, examples of such constructs may be the incomplete or conflicting assignments, use of "don't care" character in comparisons, comparisons (higher or lower) involving enumerated types (Rationale: simulation is an important verification method. If simulated and synthesized behaviours differ, then the verification chain is broken);

f) signals and variables should not be initialized at their declaration in the RTL description, but by an explicit mechanism such as reset (Rationale: initialization in HDL may lead to differences between simulated and synthesized behaviours);

g) use of explicit delays should be forbidden in the design description, as such delays lead to differences between simulated and synthesized behaviours;

NOTE 2 This does not forbid the existence of delays at system level or in the requirements of the HPD. It means that such delays cannot be implemented by a “delay” or “after” instruction in HDL, but e.g. by counters or shift registers.

h) creation of delays by means of combinatorial gates or by depending upon propagation delays along wires should be forbidden in the design description. If such design cannot be avoided, STA shall be done to justify the usage of such design (Rationale: such delays are not stable over parameters such as temperature, voltage, or from one part to another, or from one area of the die to another);

i) the types of the interface signals of the HPD should be defined in a clear and non-ambiguous way, preferably standardized, independently of any tool or micro-electronic technology;

j) HDL level definitions should not allow different interpretations, to avoid variations when compiled under different conditions. E.g. inputs / outputs of the HPD should be explicitly assigned to known pins.

NOTE 3 This subclause does not apply to the design of library components, which are build to be instantiated in different locations of future designs with different input/output allocations.

NOTE 4 To design HDL code that can be transferred between different technologies it is necessary for the pin allocation to be defined in a constraints file, not in the HDL code. Language features such as templates in VHDL-2008 may help doing this.

8.3.5 Synchronous vs asynchronous design

Synchronous design consists in enforcing the change in the state of the internal registers and of the outputs simultaneously only at times defined by a clock. It favours a modular and understandable design, it minimizes the potential for wrong behaviours due to glitches, and it favours the best use of synthesis and verification tools.

8.3.5.1 In order to facilitate stable, robust, and clearly structured designs:

a) a strictly synchronous architecture should be used;

b) non-compliances shall be justified.

8.3.5.2 The design shall ensure that signals at asynchronous interfaces are synchronized.

8.3.5.3 If an asynchronous architecture is used, a documented analysis of all paths shall demonstrate that the outputs comply with the Requirement Specification (Clause 6) and that there is no adverse glitch or metastability.

8.3.5.4 The HPD behaviour shall not be subject to the actual values of the internal propagation delays along wires and through gates

8.3.6 Power management

8.3.6.1 The internal electrical and temporal characteristics of the blank integrated circuit during power-up/start-up, power-down and sudden loss of power shall be known and taken into account in the design.

8.3.6.2 The behaviour of each pin (such as input or output type, impedance) during power-up/start-up, power-down and sudden loss of power shall be documented.

8.3.6.3 The use of HPDs based on programmable technology shall not rely on the assumption that they behave in accordance with their programmed behaviour (regarding e.g. functions, direction and impedance of each pin) during power-up/start-up, power-down and sudden loss of power, even in the case of one-time programmed devices.

8.3.6.4 The connection of input pins to a voltage source or to ground should follow the supplier’s application notes, in order to avoid potential current peaks during power-up/start-up, power-down and sudden loss of power.

8.3.6.5 When the power distribution is not predefined by the component supplier, particular care shall be taken in its design to avoid non-deterministic faults due to problems such as voltage transients due to current peaks on clock edges.

8.3.7 Initialization

8.3.7.1 The design shall have an input signal that puts all outputs, registers and finite state machines in a known and documented state.

8.3.7.2 The initialization signal, which is not always of purely digital nature, shall comply with the blank integrated circuit requirements such as rise time, fall time or monotonicity.

8.3.7.3 Asserting this signal shall produce the intended effect even when no clock activity is running.

8.3.7.4 De-asserting this signal shall be done in such a way that all outputs, registers and finite state machines are maintained in their initial known state until the clock activity is running.

8.3.8 Non-functional configurations

8.3.8.1 Special pins and registers that make the HPD switch to special configurations (such as test, diagnostic, debugging or programming) and which are not specified by the HPD Requirements Specification shall be analysed and configured in order to exclude any adverse impact on its functions.

8.3.8.2 The designers shall be familiar with the integrated circuit supplier documentation to know the characteristics given by the tools to the unused pins (input, output, high impedance, etc.).

8.3.8.3 The management of the configuration pins and registers of the HPD shall be documented.

8.3.9 Testability

8.3.9.1 Each function implemented in the HPD shall be testable (detection of failures), by means such as self-tests, periodic tests or observable contribution to a higher level function which is itself submitted to self-tests or periodic tests.

8.3.9.2 When self-test devices are used, their capability to perform their function shall be verified.

8.3.9.3 The actual coverage of fault detection (see 6.4.9) and periodic tests shall be determined and comply with the HPD Requirement Specification.

8.3.9.4 The consequences of faults shall be minimized, e.g. by detecting when states normally unreachable are reached and by taking a predefined action in such cases.

8.3.10 Design documentation

8.3.10.1 The end of the design phase shall be marked by production of the corresponding documentation.

8.3.10.2 The documentation shall describe and justify the adequacy of the design decisions in fulfilling the HPD requirement specification.

8.3.10.3 The design documentation shall be comprehensive enough so that implementation can proceed without further clarification.

8.3.10.4 The documentation shall describe the design decisions such as:

a) the organization in modules, as well as their interfaces and relations,

b) the control flows and data paths,

c) the protocols and algorithms,

d) the types, formats, and logic conventions of the signals,

e) the numbering of buses, the memory map,

f) the finite state machines definitions, encoding, and initializations,

g) the initialization value of all registers,

h) the test circuitry.

8.3.10.5 The design documentation shall define the variant actually used for each instantiation of each library component, to avoid ambiguities when variants with different speeds or electrical characteristics exist.

8.3.10.6 The design documentation shall include all parameters needed to unambiguously configure and use all native blocks and PDBs.

8.3.10.7 The design documentation shall include the estimated timings and electrical characteristics.

8.4 Implementation

8.4.1 General

Starting from the RTL description, the implementation synthesizes the gate-level description (netlist) of the HPD. Then place and route is performed and results in the physical description needed to produce the HPD, such as programming file or "bitstream".

8.4.2 Products

8.4.2.1 The implementation shall generate all information necessary to produce the HPD in a systematic way and to verify that each produced part complies with the design.

8.4.2.2 The implementation shall produce timing information to supplement the RTL description ("back-annotations") in order to precisely simulate the temporal behaviour taking into account all delays associated with gates and wires.

8.4.2.3 The back-annotated description shall be usable in the test-bench (9.5) and, when appropriate, in higher level tools such as board level simulation.

8.4.3 Files of parameters and constraints

The designer directs the synthesis, place and route operations with parameters and directives which specify constraints such as needed operating frequency, timing relations between signals, or fan-out. To fulfil these constraints (provided to the tools in “constraint files”) the tools may modify the placement to favour a given propagation path at the expense of other ones, duplicate one gate to reduce the load on each copy and thus increase their speed, etc.

Errors or omissions in parameters and constraints files may result in subtle non-deterministic faults, often not detectable during simulation and sensitive to normal variations of the micro-electronics process.

8.4.3.1 The files of parameters and constraints shall be built according to an auditable process.

8.4.3.2 The completeness and the correctness of the files of parameters and constraints shall be verified by the verification team (see Clause 9).

8.4.3.3 The files of parameters and constraints shall be documented and placed under configuration management.

8.4.4 Post-route analyses

8.4.4.1 A post-route analysis shall demonstrate the compliance of the design and implementation with the technology rules defined by the suppliers of the design and implementation tools and of the micro-electronic technology.

8.4.4.2 Post-route analyses or simulations (taking into account the post-route timing information, or back-annotations) shall confirm the cycle by cycle equivalence of the post-route description to the RTL description for fastest and slowest cases, including initialization, for example by using the two following steps:

a) demonstrating that the post-synthesis description is cycle by cycle equivalent to the RTL description,

b) demonstrating that the post-route description complies with the timing constraints.

8.4.4.3 Post-route simulations may use a subset of the test-bench cases used in the RTL simulations (see 9.5). It shall be justified that this subset covers the needs of equivalence demonstration. An alternative or complementary method to post-route simulation is to use a tool that will check that the RTL and the physical description level are mathematically equivalent. If this approach is adopted the tool used to perform this check shall be assessed for quality and suitability before use (see Clause 15).

8.4.4.4 Post-route timings shall be analysed.

8.4.4.5 The coverage of each function by self-supervision shall be analysed with respect to the required target (see 6.4.9) taking into account the effects the tools may have on the actual topology.

8.4.4.6 These analyses shall be detailed enough and documented, in order to allow further technical assessment by people not involved in the design and implementation.

8.4.4.7 Some of these analyses may be performed, optionally or automatically, by the tools. In that case it is not required to perform them again, but:

a) it shall be demonstrated that the analyses performed by the tools have appropriate coverage and correctness;

b) the analysis reports (including set-up and results) provided by the tools shall be included in the documentation.

8.4.4.8 If the analyses find non-compliances deemed acceptable then:

a) this acceptability shall be justified and documented;

b) all impacted documents shall be modified accordingly;

c) the quality plan shall ensure that any impact on other systems or components is documented and adequately taken into account by the people responsible for the impacted systems and components.

8.4.5 Redundancies introduced or removed by the tools

8.4.5.1 The replications of gates made by the tools to meet timing or technology constraints shall be analysed.

8.4.5.2 It shall be demonstrated that the additional states introduced by these replications are acceptable regarding the functional and safety requirements. It is recognized that gate replication is performed by many synthesis tools, but usually, this can be adequately controlled in the tool itself. However, care shall be taken as gate replication may cause problems if the same formal equivalency check is used to prove the RTL and the gate implementation.

8.4.5.3 As replication introduces new states they, shall be analysed to demonstrate that the safe behaviour of the design cannot be affected.

8.4.5.4 On the other hand, it shall be demonstrated that the logic optimization performed by the tools has not removed fault detection and tolerance mechanisms such as redundancies or processing of cases normally unreachable.

8.4.6 Finite state machines

8.4.6.1 The robustness of the final implementation of finite state machines shall be analysed.

8.4.6.2 In particular, finite state machines shall not have dead states other than those possibly specified in the HPD requirement specification.

NOTE A dead state is a state from which the finite state machine cannot reach any other state.

8.4.6.3 The potential additional states introduced by some encoding methods (such as "one-hot") shall be taken into account in failure analysis.

NOTE "one-hot" encoding uses one flip-flop per state to be represented; each particular state is represented by one specific flip-flop set to "true" and all others set to "false". Thus, only combinations with exactly one flip-flop set to "true" are valid. In case of failure, several flip-flops could be simultaneously set to "true", which would correspond to additional, undefined states.

8.4.7 Static timing analysis

8.4.7.1 A Static Timing Analysis (STA) shall be performed and documented for worst and best cases to calculate the margins, taking into account the timing information provided by the technology libraries and all relevant design and implementation tools.

8.4.7.2 If paths are excluded from STA (because seen as "false paths") or declared as multi-cycle paths, this decision shall be justified and documented.

8.4.7.3 STA shall demonstrate that the frequency of each clocked block is compatible with all non-excluded paths (see 8.4.7.2) with sufficient margin within the specified variability of the micro-electronic technology.

8.4.7.4 The effect of the clock skew on critical structures such as shift registers shall be analysed and documented.

NOTE The clock skew is the amount of time between the arrivals of the clock signal at different locations.

8.4.8 Implementation documentation

The end of the implementation phase shall be marked by production of the corresponding documentation including:

a) the gate-level description of the HPD, usable in the same test-bench as used at RTL level,

b) the technology specific description (e.g. "programming file") necessary to program the HPD and to test each part (13.2),

c) the back-annotations that take into account all delays associated with gates and wires,

d) the timings (such as frequency, set-up and hold times, rise and fall times, propagation times) and electrical characteristics (such as voltage levels, input currents, fan-out, impedances, and power consumption) predicted by the tools unless they are already defined in the datasheet.

8.4.8.1 The implementation documentation shall:

a) give access (by inclusion or reference) to the implementation of each block, sub-block or module,

b) describe the choices made, in particular regarding testability, clock and power distribution, reset and critical paths implementation.

8.4.8.2 The implementation documentation shall describe and justify:

a) the constraints and parameters provided to the tools,

b) the analysis performed to guarantee the compliance of the HPD with its requirement specification, and any differences found,

c) any iterations made on design and implementation,

d) any redundancy added or removed during implementation.

8.4.8.3 The documentation shall be detailed enough to allow an engineer not involved in the project to run the synthesis, place and route tools and get the same results (HPD and verification output), as well as to verify the completeness and the correctness of the post-route analyses.

8.4.8.4 The documentation shall describe the tests to be performed periodically in operation, with due care to the structural modifications introduced by the tools.

8.4.8.5 When the commitment of the integrated circuit supplier on the design or implementation is required before production, this commitment shall be included in the documentation.

8.5 System level tools and automated code generation

The requirements of the different components of a system may be captured using ESL tools that provide a textual or graphical description.

This subclause provides guidance applicable when an ESL description of the HPD requirements is used in an automated way to generate part or all of the HPD design. This generation is sometimes called “high-level synthesis”.

8.5.1 If a requirement specification written in an ESL language is used to automatically generate an RTL description of the HPD or a part of it:

a) the generated description should be straightforward and avoid unnecessary complexity;

b) this description should allow the behaviour of the device to be easily understood so that errors and ambiguities can be identified promptly by the hardware design engineers.

8.5.2 The ESL language and the associated tools, in particular those used for code generation and analysis, should comply with the requirements of 8.2.

8.5.3 If 8.5.2 is not fulfilled:

a) the ESL description of the HPD shall be translated into an HDL description compliant with the requirements of 8.2, which will be the basis for the following activities of design, implementation and verification,

b) these following activities shall comply with the requirements of this Standard.

8.5.4 Any non-conformance of the generated descriptions (such as RTL, synthesized, routed) with the requirements for design and implementation (8.3 and 8.4) shall be identified and justified.

8.5.5 If some of the analyses, verifications and reviews defined by this Standard in Clauses 8, 9 and 10 are not performed, it shall be formally proven that the products which have not been analysed, verified or reviewed are necessarily correct.

8.5.6 The generated products shall not be modified by direct manual action on the products.

8.5.7 The products shall be regenerated if anything has to be modified, for example with respect to findings from verification or review activities.

8.6 Documentation

This subclause gives the general documentation requirements for design and implementation of the HPD. It supplements the requirements specific to particular activities addressed in 8.1 to 8.5.

8.6.1 The end of the design and implementation phases shall be marked by the production of the HPD design specification.

This document serves as the basis for the formal design and implementation review and the subsequent production.

8.6.2 Sufficient detail shall be included so that production can proceed without further clarification.

8.6.3 The document should be structured according to the phases of the development process. The design specification may be expressed as one document or as an integrated set of documents.

8.6.4 If multiple documents are used, each document shall have a defined relationship to the other documents and shall contain a well-bounded subject-matter.

8.6.5 Documentation formats should be selected according to the specific topics, including:

a) narrative description;

b) arithmetic and logic expressions;

c) graphical representations, diagrams and drawings.

8.7 Design and implementation review

8.7.1 The design and implementation phase shall be terminated by a formal review.

8.7.2 The design and implementation review shall examine the documentation, covering design, implementation, analyses and verifications.

8.7.3 The review shall examine the completeness and correctness of the files of parameters and constraints provided to the design and implementation tools.

8.7.4 The review shall examine the completeness and correctness of the Static Timing Analysis (STA) and post-route analyses, to check the correctness and the robustness of the design and implementation with due consideration to the potential adverse effects induced by the modifications made by the tools (such as logic simplification or gate duplication).

8.7.5 The review team shall include hardware experts and engineers from the teams responsible for the system or components that use the HPD or are interfaced to it (such as electronic board or software).

9 HPD verification

9.1 General

The verification activities undertaken as part of the HPD development are usually under the responsibility of the I&C producer and are undertaken by staff independent of those performing the HPD design and implementation. The most appropriate way is to engage a verification team.

Additional verification activities may be undertaken as part of a third party assessment of the HPD and of its development process in order to provide assurance that it will meet its targets. There are many ways by which this independent verification role can be resourced and implemented, this often being a matter of national regulatory preference.

9.1.1 The verification team shall be composed of individuals who are not engaged in the development and who have the necessary competencies and knowledge. The following requirements define explicitly the level of independence required.

9.1.2 The management of the verification team shall be separate and independent from the management of the development team.

9.1.3 Communication between the verification team and the development team, whether for clarification or fault reporting, shall be conducted formally in writing at a level of detail which may be audited.

9.1.4 Interactions between the two parties should aim at maintaining the independence of judgment of the verification team.

9.1.5 The verification team shall have clearly defined responsibilities and obligations.

9.1.6 The output of each development phase (Figure 2) shall be verified.

9.1.7 The verification activities shall confirm the adequacy of the HPD requirement specification in fulfilling the system or subsystem requirements assigned to the HPD by the system or subsystem specification.

9.1.8 The verification activities shall confirm the adequacy of the selection and rules of use of each blank integrated circuit, micro-electronic technology, native block and PDB in fulfilling its component requirement specification (see Clause 7).

9.1.9 The verification activities shall confirm the adequacy of the HPD design specification in fulfilling the HPD requirement specification.

9.1.10 The verification activities shall confirm the compliance of the HPD with the HPD design specification (see Clause 8).

NOTE Post-implementation verification is of major importance to detect the potentially adverse effects of logic simplifications and gate duplications that may be performed by the tools, as well as the potential faults resulting from the tools themselves or from their use.

9.1.11 Each production activity should be started on a basis of verified input data/documents.

9.1.12 Verification of the product of a phase should be performed before the start of the next phase. Otherwise, this verification shall be performed before the verification of the next phase.

Possible preparatory work for a subsequent phase may be done before the precedent phase has been verified.

9.1.13 If input data/documents for an activity have been modified, that activity and subsequent activities shall be repeated as necessary to address potential impact.

9.2 Verification plan

9.2.1 The verification plan shall be established prior to starting HPD verification activities.

9.2.2 The plan shall document all the criteria, the techniques and tools to be utilized in the verification process.

9.2.3 It shall describe the activities to be performed to evaluate each item of the HPD, each tool involved in the development process, and each phase to show whether the HPD requirements specification is met.

9.2.4 The level of detail shall be such that a verification team can execute the verification plan and reach an objective judgement on whether or not the HPD meets its requirement specification.

9.2.5 The verification plan shall be prepared by a verification team addressing:

a) selection and justification of verification strategies according to the nature of the requirements, to the design and implementation characteristics, and to the micro-electronic technology;

b) selection and utilisation of the verification tools;

c) execution of verification;

d) documentation of verification activities;

e) evaluation of verification results gained from verification tools directly and from tests, evaluation of whether the safety requirements are met or not.

9.2.6 The verification plan shall document each test, including its goal, its expected results, and the criteria to decide whether the result is correct or not.

9.2.7 The tests designed according to functional aspects should result in extensively exercising the HPD.

9.2.8 The verification plan shall identify any objective evidence required to confirm the extent of testing. For that purpose the test coverage criteria chosen according to the design and implementation shall be justified and documented.

9.2.9 There shall be adequate provision for the processing and resolution of all safety issues raised during the verification activities performed either during the development by the I&C producer or by a third-party assessment.

9.2.10 All safety issues shall be adequately resolved through appropriate corrective modifications or mitigating dispositions.

9.3 Verification of the use of the pre-developed items

The correct configuration and use of the pre-developed items such as blank integrated circuits, native blocks and PDBs, as well as their mutual compatibility, shall be verified against the rules specified by their suppliers and against those elaborated during the activities described by Clause 7.

9.4 Verification of the design and implementation

9.4.1 The verification shall include test and analyses to address:

a) the adequacy of the design specification for the HPD requirement specification with respect to consistency and completeness down to and including the lowest block and module level;

b) the decomposition of the design into a hierarchy of blocks and modules and the way they are specified with respect to:

   1) testability for further verification;

   2) understandability by the development and verification teams;

   3) modifiability to allow for further modification;

c) the correct implementation of safety requirements;

9.4.2 The result of the verification shall be documented.

9.4.3 The documentation shall include the conclusions and identify clearly issues that need actions, such as:

a) items which do not conform to the requirements;

b) items which do not conform to the design and implementation rules;

c) modules, data, structures and algorithms poorly adapted to the problem.

9.5 Test-benches

9.5.1 A simulation and test program (test-bench) shall be developed and documented. As needed, this test-bench may consist in several implementations, each with a different scope and objective, e.g. some test-benches may be dedicated to modules and one or more to top-level testing.

9.5.2 The test-bench (structure) may be developed by the design team for its own testing needs and used by the verification team. However, the test vectors (inputs and expected outputs) required by this Standard shall be developed by the verification team, so as to reduce the potential for error masking and to give additional confirmation of understandability and completeness of the design documentation.

9.5.3 The test-bench shall

a) exercise each module in its environment simulated with all needed logical details,

b) provide sufficient timing resolution when used after implementation for the temporal aspects.

9.5.4 The test-bench shall include test cases which exercise all the features mentioned in the HPD requirement specification and in the design specification such as functions, modes, finite state machines, algorithms, protocols.

9.5.5 The test-bench should include all required inputs, sequences and timings, and record all output sequences and timings produced during execution, to make the test execution fully automated.

9.5.6 The test-bench should include the expected output sequences and timings as well as an automated comparison with those actually produced during test execution (with respect to the adequate criteria, see 9.2), so as to provide a global "pass/fail" output in addition to the detailed test results.

9.5.7 If manual inputs, observations or comparisons are required:

a) the involved data and activities shall be documented with sufficient detail to allow a person not involved in the project to repeat the test. This may need definition of cycle by cycle steps and bit level values,

b) a documented justification shall be provided, because manual inputs, observations and comparisons are potentially error-prone.

9.5.8 The test-bench has to accurately report all failures and not falsely report passes. It shall therefore be built in accordance with 10.4.6 and 15.2.

9.6 Test coverage

9.6.1 Test coverage criteria shall be selected and documented.

9.6.2 A documented analysis of the test coverage criteria shall demonstrate that they are sufficient regarding the HPD requirement specification and design/implementation characteristics, and that the test-bench provides sufficient observability to take a pass/fail decision for each covered element.

9.6.3 Such criteria may be related, for example, to instructions, decisions, expressions, paths, finite state machines, or processes. If a coverage criterion target could not be achieved, e.g. due to RTL structure (100 % path coverage is particularly difficult to achieve), then a documented justification shall be produced.

NOTE A path is a particular sequence of branches taken when executing the code.

9.6.4 Each module developed within the project shall be specifically tested.

9.7 Test execution

9.7.1 Tests shall be performed using the test-benches following the design phase on the RTL description, in order to confirm its correctness.

9.7.2 Tests shall be performed after the implementation phase to confirm that the post route description complies with the timing constraints, taking account of the timing information provided by the tools and libraries (back-annotations).

9.7.3 The tests (using simulation) shall be performed for both “worst case” (maximum propagation delay) and “best case” (minimum propagation delay).

9.7.4 The test results (values, sequences, and timings) shall be documented.

9.7.5 A documented analysis of any discrepancy shall decide whether it is acceptable or not.

9.8 Static verification

9.8.1 The following verification activities should be performed:

a) type and syntax checking,

b) parameter checking in call or instantiation of modules, functions, procedures, native blocks and PDBs,

c) out of range checking,

d) completeness of the sensitivity list of processes (see note),

e) completeness of the cases explicitly programmed in instructions and constructs with multiple choices,

f) detection of dead states in finite state machines,

g) detection of side effects in functions or macros, detection of shared objects,

h) logical and physical DRC (Design Rule Check), which test the netlist and other generated files for physical and logic errors.

NOTE “Sensitivity list” is an element of VHDL.

9.8.2 Static verification methods such as STA (see 8.4.7) may be used for some aspects of the verification if their principles are mathematically sound. In this case, the tools used to implement these methods shall:

a) have maturity and standardization similar to those required by this Standard for the simulation tools,

b) comply with the requirements of Clause 15 applicable to verification tools.

10 HPD aspects of system integration

10.1 General

The process of system integration is the combining of verified hardware (and software when applicable) components into subsystems and finally into the complete system. This process consists of two kinds of activities:

a) system integration: assembling and interconnecting verified hardware components (and software components when applicable) in order to build the intermediate and final targets. The assembly sequence as well as the degree of integration of the successive targets depend on the project characteristics;

b) integrated system verification: verifying that the components comply with their design specification, are capable of operating together, and comply with their interface requirements.

This clause gives requirements for the system integration in supplement to 6.2.5 of IEC 61513, when HPDs are involved.

10.2 HPD aspects of the system integration plan

This subclause amplifies the requirements of IEC 61513, 6.3.4., which shall apply.

10.2.1 This plan shall be prepared and documented in the design and implementation phases and verified against the class 1 system requirements.

10.2.2 This plan shall be prepared sufficiently early in the development process to ensure that all integration requirements are included in the design of the HPD, of the system and of its components.

10.2.3 This plan shall specify the Standards and procedures to be followed in the system integration phase.

10.2.4 This plan shall document those provisions of the system quality assurance plan that are applicable to the system integration.

10.2.5 The integration plan shall specify:

a) the sequences and timings of input signals to the system or subsystem being tested,

b) the sequences and timings of expected outputs from the system or subsystem being tested,

c) the acceptance criteria.

10.2.6 The system integration plan shall take into account the requirements to be fulfilled by the HPD through the design of the system, the design of the hardware and the design of any software. The plan shall also include the requirements for procedures and control methods covering:

a) system configuration control (5.5);

b) system integration;

c) integrated system verification;

d) fault resolution.

10.2.7 The system integration plan shall define both the identification and control aspects of configuration management, according to the requirements of IEC 61513, 6.3.2.3.

10.2.8 In the process of verifying the interactions of the HPD with the other components of the system, certain aspects may be verified at the level of subsystems (computing units) or at the level of the complete system if more practical. When verification by testing is not feasible at these levels, then:

a) all requirements of the HPD shall be verified by other means (e.g. white box testing),

b) the corresponding verification strategy shall be documented in the integration plan.

10.2.9 All interdependencies between the verification of the HPD and the verification of the integrated system shall be documented in the system integration plan.

10.3 Specific aspects of system integration

The specific procedures for the system integration depend on the characteristics of the system architecture.

10.3.1 Procedures shall be established and referenced by the system integration plan to cover the following activities:

a) the acquisition of the correct components according to the system configuration management plan (6.3.2.3 of IEC 61513) and to the production procedures (Clause 13);

b) the integration of the HPD into the system (e.g. component positioning, configuration, interconnection wiring);

c) the preliminary functional test of the integrated system functions (see requirements below);

d) the documentation of the outcomes of the integration process and the system configuration subjected to the tests;

e) the formal release of the integrated system for validation testing.

10.3.2 If the resolution of a fault requires a modification to the verified HPD or the design specification, that fault shall be reported according to the procedures established in 10.5.

10.3.3 Any faults detected during the system integration which are only mistakes in the integration process itself, and which do not affect any HPD document, shall be corrected by updating the system integration plan.

10.4 Verification of the integrated system

The verification of the integrated system determines whether or not the verified components and the subsystems have been properly integrated into the system, that they are compatible and perform as specified.

10.4.1 The system shall be as complete as is practical for this verification.

10.4.2 The test cases selected for system verification shall:

a) exercise all HPD interfaces and all basic operations;

b) exercise all interface characteristics of the HPD described in the requirement specification and in the design specification such as protocols, sequences, timings and electrical features;

c) have sufficient coverage to demonstrate that the HPD performs as required for all cases reachable in the system.

10.4.3 The system integration plan shall identify the tests to be performed for each HPD interface requirement.

10.4.4 The integrated system test program shall be reviewed and the test results evaluated by a verification team with a good knowledge of the system specification.

10.4.5 Equipment used for system verification shall be calibrated appropriately.

10.4.6 The verification software tools used shall comply with the requirements of Clause 15 addressing verification tools.

10.4.7 The verification of the integrated system shall demonstrate that all system components have appropriate performance (e.g. processing units and communication devices).

10.5 Fault resolution procedures

10.5.1 The requirements of IEC 61513, 6.3.2.4 (Fault resolution procedures) shall apply.

10.5.2 The fault resolution procedures shall ensure that any required modification to the HPD fulfils the requirements of Clause 12.

10.6 HPD aspects of the integrated system test report

10.6.1 The requirements of 6.4.5.2 of IEC 61513 shall apply.

10.6.2 The test results shall be retained in a form that makes them verifiable by persons not directly engaged in the verification plan or in the actual performance of the tests.

11 HPD aspects of system validation

11.1 General

HPDs are typically validated within the system validation phase. System validation is covered by IEC 61513. This Standard provides additional requirements to validate the performance (functional, temporal and electrical) of HPDs.

a) Testing shall be performed to validate the system and the HPD in accordance with the class 1 systems requirements.

b) Validation tests shall be performed on the system in its final assembly configuration including the final version of the HPD.

11.2 HPD aspects of the system validation plan

11.2.1 The system validation shall be conducted in accordance with a formal system validation plan.

11.2.2 The plan shall identify static and dynamic test cases.

11.2.3 The system validation plan shall be developed and the result of the validation evaluated by individuals who did not participate in the design and implementation.

11.3 System validation

11.3.1 The system shall be exercised by static and dynamic input signals simulating normal operation, anticipated operational occurrences and accident conditions requiring action.

11.3.2 Each category A function of the system shall be exercised by a set of tests confirming each required output signal in a single or combined manner.

11.3.3 The tests shall:

a) cover all the functions of the HPD requirement specification, in all modes (6.2);

b) cover all ranges of signals and computed parameters as far as practical;

c) cover the voting, other single or combined logics in comprehensive manner;

d) be made for all trip or protective signals in the final assembly configuration;

e) cover the required response to specified failures;

f) cover all other functions which have an impact on reactor safety.

11.3.4 In addition, the values of input signals, the expected output signals and the acceptance criteria shall be stated in the system validation plan.

11.3.5 Equipment used for validation shall be calibrated and configured (hardware and software parameters) appropriately.

11.4 HPD aspects of the system validation report

11.4.1 The system validation report shall document the results related to the HPD included in the system.

11.4.2 The report shall identify the hardware, the software when applicable, the system configuration used, the tool configurations used and the test equipment used (including its calibration and the simulation models) in compliance with IEC 61513, 6.4.6.2.b.

11.4.3 This report shall also identify any discrepancies found during the test.

11.4.4 This report shall summarise the results of the system validation.

11.4.5 This report shall assess the system compliance with all requirements.

11.4.6 The results shall be retained in a form that makes them verifiable by persons not directly engaged in the validation.

11.4.7 Simulations of the plant and its systems used for the validation shall be documented.

11.5 Fault resolution procedures

The requirements of 10.5 shall also apply to the aspects of system validation related to the HPD.

12 Modification

12.1 Modification of the requirements, design or implementation

12.1.1 The modification process and documentation shall comply with the requirements of IEC 61513 (6.2.8 and 6.4.7), of IEC 60987:2007 (Clause 12) and of IEC 60880:2006 (Clause 11).

12.1.2 All impacted documents shall be verified according to the requirements of Clause 9, by individuals who are not engaged in the design or implementation of the modification.

12.2 Modification of the micro-electronic technology

The supplier may update the micro-electronic technology (e.g. new version of a blank FPGA to increase the speed or to reduce the die size). Even if the new part is claimed to be "compatible" this does not imply that any given design will perform identically on both devices.

12.2.1 The acceptance process (Clause 7) shall be performed again, followed if necessary by all impacted phases of the life-cycle depending on the differences found.

12.2.2 The relevant verification activities shall be performed again and duly documented to ensure that all functional, electrical, and timing requirements are met.

12.2.3 Even if the new and old parts have the same logic configuration and are pin-to-pin compatible, the need to re-generate the programming files (e.g. because of variations in timings or voltages of the programming pulses) shall be assessed and documented.

13 HPD production

13.1 General

The scope of this Standard excludes the design and manufacture of the pre-developed micro-electronic resources (e.g. a blank FPGA) used as inputs by the development process of the HPD. “Production” in this Standard designates the final steps which deliver the integrated circuit ready for use in the I&C system.

13.2 Production tests

13.2.1 Tests shall verify the functions of the HPD as well as its temporal performance (such as frequency, rise and fall times, propagation time, etc.) and its electrical characteristics (such as power consumption, capacitances, etc.).

13.2.2 It should be demonstrated that the tests performed by the supplier of the integrated circuit fulfil the needs (see 13.2.1). The I&C producer does not need to repeat the tests performed by the supplier of the integrated circuit nor to know the corresponding test vectors.

13.2.3 If 13.2.2 is not fulfilled, then additional tests (with documented inputs, expected outputs and acceptance criteria) shall be performed by the I&C producer to cover the needs (see 13.2.1).

13.2.4 Production tests performed at board level (after assembly of the HPD onto the printed circuit board – e.g. by soldering) shall verify that the interface of the part is operational (such as “I/O pin stuck at” fault test, global functional test).

13.2.5 Each produced part shall pass the production tests or shall be rejected.

13.2.6 The test results shall be stored together with identification information such as batch number in order to support the diagnosis of potential process problems.

13.3 Programming files and programming activities

13.3.1 The programming files shall include error detection codes, and the programming equipment shall verify them.

13.3.2 For each produced part:

a) the configuration after programming shall be verified and

b) relevant traceability information (such as batch number, programming log file, characteristics of programmable switches before and after programming) shall be stored.

13.3.3 All procedures and requirements given by the integrated circuit supplier shall be fulfilled (e.g. to prevent electrostatic discharge).

13.3.4 Only tools guaranteed and supported by the integrated circuit supplier shall be used.

14 HPD aspects of installation, commissioning and operation

a) The process and documentation for installation, commissioning and operation shall comply with the requirements of IEC 61513 (6.2.7 and 6.3.6), of IEC 60987:2007 (Clause 10 and Clause 13) and of IEC 60880:2006 (Clause 12).

b) According to IEC 60671 I&C systems and equipment performing category A functions are periodically tested to demonstrate proper function. To achieve the required test coverage for the HPD, appropriate test techniques shall be used to increase the testability, e. g. boundary scan.

15 Software tools for the development of HPDs

15.1 General

Clause 14 of IEC 60880:2006 shall apply to the software tools used for HPD development, with the exception of 14.3.4.3, 14.3.4.4 and 14.3.4.5.

NOTE 1 Tool vendor’s technical evaluation (not limited to quality assurance) is an acceptable method to fulfil the requirement 14.2.2 of IEC 60880:2006, provided that the corresponding documentation is available.

NOTE 2 The “reliability” attribute of software tools addressed in Clause 14 of IEC 60880:2006 means here “trustworthiness” or “correctness”.

NOTE 3 ISO/IEC 9126 is superseded by ISO/IEC 25000.

NOTE 4 Libraries integrated in tools may be evaluated in the context of the tool evaluation.

NOTE 5 The verification of tool outputs addressed by 14.3.2.4 of IEC 60880:2006 may be performed by different ways, e.g. simulation with a simulator diverse from the synthesizing toolset.

15.2 Additional requirements for design, implementation and simulation tools

15.2.1 Software tools shall give access to the parameters that control the logic synthesis and the implementation (e.g. through settings).

15.2.2 Software tools should not add structures not directly traceable to HDL source statements (e.g. gate duplication to match timing requirement) without warning.

15.2.3 The designers shall have previous knowledge of the software tools, in particular they shall know how they perform on the structures and constructs used in the project.

15.2.4 If a software tool requires command line arguments these shall be in a script file (placed under configuration management) to avoid manual invocation errors.

NOTE 1 This is useful not only for the consistency; it also helps in assessing the origin of a fault, which may lie in the source code, in the tool or in the tool parameters. It may also be necessary in the assessment of the potential for CCF due to design and implementation tools.

15.2.5 When moving to a new version of a software tool that is responsible for a transformation of design information (e.g. logic synthesis or place and route), all affected simulation, analysis and verification activities shall be performed again.

NOTE 2 It can be justified by documented analysis that a given modification of a tool cannot affect the abovementioned activities, e.g. correction of some inconsistent behaviour in the tool graphical user interface.

NOTE 3 Activities which have been completed before the tool change do not need to be repeated.

16 Design segmentation or partitioning

16.1 Background

It is possible on some HPD devices to design and implement circuits that are allocated using physically different areas of the integrated circuit, have minimal or no interconnections together, and use no common hardware resources. Some HPDs support such areas, sometimes called “lakes”, with unused/unusable space between them. Some of the advantages of design segmentation or partitioning may include the implementation of auxiliary or support functions (this is not to be a replacement for redundant channels/trains in a design at system level).

16.2 Auxiliary or support functions

16.2.1 General

In general, auxiliary or support functions implemented on a HPD, even if not performing Category A functions, have the potential to interfere with category A functions of that HPD. Thus, unless it can be shown that the requirements of 16.2.2 are met, auxiliary or support functions shall be developed, implemented and verified according to the requirements of this Standard (i.e., as Category A functions).

16.2.2 Partitioning of auxiliary or support functions of category other than A

This Standard recognizes that it may be possible with specific design measures and partitioning of the HPD to ensure that auxiliary or support functions are independent of those of Category A and cannot inappropriately interfere with them. In such cases, provided the following requirements are met, auxiliary or support functions may be implemented on a class 1 HPD without the same rigour as for Category A functions:

a) it shall be demonstrated by design, implementation, assessment and systematic verification that the operation or failure of such auxiliary or support functions cannot interfere directly or indirectly with any Category A function, whether the cause of the failure is internal or external to the HPD (e.g. induced by the power supplies, a short circuit on a connected line, etc.),

b) this demonstration shall address all potential causes of interference e.g. functional, electrical, electromagnetic, thermal, etc.,

c) in particular the areas of the integrated circuit used to implement such auxiliary or support functions shall be physically different from those used to implement the Category A functions,

d) in case of modification of the HPD, it shall be demonstrated that the requirements of 16.2.2 are still fulfilled,

e) the interface between circuits implementing Category A functions and auxiliary or support functions shall be simple and fully verifiable,

f) data received by Category A functions from auxiliary or support functions shall be limited to static parameter values (e.g. calibration constants, set-points),

g) Category A functions shall not have any time dependence on receipt of data from auxiliary or support functions,

h) appropriate safety measures (e.g. safe communications protocols) shall be implemented for any communication between Category A functions and auxiliary or support functions such that all data transfer errors will be detected and a suitable safe response taken, or correct receipt of data is acknowledged.

17 Defences against HPD Common Cause Failure

17.1 Background

Systematic faults may be introduced in any design and implementation process due to human error; and therefore such faults may be introduced during HPD design and implementation (either in the developed part or in an included pre-existing design). HPDs could therefore potentially be affected by latent systematic faults which could, under some triggering event, lead to the CCF of multiple instantiations of a HPD design.

The potential for CCF across multiple systems is in the scope of higher level SC 45A Standards, in particular IEC 61513 and IEC 62340. The potential for CCF due to multiple instantiations of a HPD design in a given system is addressed by this Standard. As explained in Clause 1, “Scope and object”, this Standard defines development and verification processes and requirements which minimise the potential for HPDs to have systematic faults and therefore –as such faults can cause CCF-, also minimise the potential for CCF due to HPDs.

Additional requirements to address protection against systematic faults which may lead to CCF due to HPDs are provided in the following subclause.

17.2 Requirements

17.2.1 Aspects of the HPD development processes which may lead to CCF of multiple instantiations of the HPD design (and which are not already addressed by clauses within this Standard) shall comply as applicable with the relevant requirements of IEC 60880:2006, 13.1 (by replacing “software” with “HPD”).

NOTE 1 These aspects are typically related to the development of HDL programs.

17.2.2 An analysis according to the relevant requirements of IEC 60880:2006, 13.3.1, 13.3.2, 13.3.3, 13.3.4, 13.3.7 and 13.3.8 shall be performed as applicable to address aspects of the HPD development processes which may lead to CCF of multiple instantiations of the HPD design (and which are not already addressed by clauses within this Standard).

NOTE 2 Some requirements of these subclauses address CCF across systems at I&C architecture level, although they would be better located in the high-level Standard dedicated to CCF, IEC 62340. In order to keep the structure of SC 45A Standard series, it is suggested to move these requirements to IEC 62340 at the next maintenance cycle.

Annex A
(informative)

Documentation

This annex identifies typical documentation for each of the main clauses of this Standard. The contents may be organized into a set of documents different from those suggested in this annex, provided that the sections are clearly identified.

A.1 Project

a) project management plan

b) quality assurance plan

c) configuration management plan

A.2 HPD requirement specification

a) requirement specification

b) requirements analysis report

c) review report

A.3 Acceptance of blank integrated circuits, native blocks and PDBs

a) component requirement specification

b) user documentation for safety

c) other documentation of the component, including any information such as specification, design, test, operating experience

d) analysis report

e) document containing the rules of use

f) acceptance review report

A.4 HPD design and Implementation

a) design specification including:

   1) description of: breakdown into main modules, defensive design choices, identification of the micro-electronic technology, tools, native blocks and PDBs

   2) description of detailed design including:

      - RTL description
      - organizational choices (modules, sub-modules, interfaces, protocols, etc.)
      - preliminary electrical characteristics and timings

   3) description of implementation including:

      - gate-level description ("netlist"), technology specific description for production, back-annotations
      - implementation of modules, critical signals and power distribution, tool options, auxiliary files used for implementation such as "constraints files"
      - post-route analyses report, STA report
      - testability analysis, test vectors for periodic testing
      - electrical characteristics and detailed timings

b) review reports

A.5 HPD verification

a) verification plan

b) document containing: description of test-bench, coverage criteria, test cases

c) document containing the analysis and justification of coverage criteria

d) report including: test results and analysis (RTL level, post-synthesis, post-route), analysis of the fulfilment of the rules of use

A.6 HPD aspects of system integration

a) integration plan including: integration strategy and procedures, configuration management interface, test cases

b) specific aspects of integrated system test report, including identification of components and tools, test results and analysis, faults found and resolution

c) integration review report

A.7 HPD aspects of system validation

a) validation plan including test cases

b) report including: identification of components and tools, test results, test analysis, faults found and resolution

A.8 Modification

IEC 60880, Annex F gives the typical documentation list regarding the modification process:

a) anomaly report

b) modification request

c) modification report

d) modification control history

In addition, the documents related to the development phases affected by the modification have to be updated.

A.9 HPD production

a) document containing the production tests

b) document containing the results of production tests, the part identification information and the part programming information

A.10 Software tools for the development of HPDs

a) tool selection report (analysis of tool support, evaluation, acceptance, limits of applicability)

b) document describing the strategy for modification, upgrade or replacement

Annex B
(informative)

Development of HPDs

The development activities addressed by this Standard are based on Hardware Description Languages and design tools running on workstations, according to a flow whose broad outline is presented here to ease the understanding of the corresponding clauses of this Standard.

B.1 Optional capture of requirements at Electronic System Level

Capture of requirements is sometimes done by means of a high level description of the system to which the HPD belongs: this description includes the other hardware and software components. Each component is represented by a behavioural model, and these models exchange information through communication channels to simulate the intended system.

This description level is called "Electronic System Level", or ESL, and uses system description languages such as SystemC or System Verilog.

This description is typically executed (simulation) with functional test cases to estimate the relevance of different system architectures, select the best one, and finally set-up the requirements of each component including the HPD in terms of behaviour and interface.

B.2 Design

Starting from the requirements, this activity initially aims at defining the main design principles, such as the partition in pre-developed or bespoke modules, the organization of the self-supervision and the identification of the micro-electronic technology (including its Native Blocks) and PDBs that could be used.

Then an RTL (Register Transfer Level) description is built and tested by simulation. HDLs such as VHDL or Verilog are used. This is mostly not dependant on the micro-electronic technology that will be used.

This high level description is a synchronous parallel model of the HPD, describing its behaviour by means of signals transformed by combinatorial functions and sequentially transferred between registers triggered by one or more clocks.

The RTL description has structural aspects, showing the logical relations between modules which can be designed specifically or taken from libraries. It also has behavioural aspects, making it possible to describe the function of a module by means of algorithmic descriptions. This description is carried out by means of a HDL (Hardware Description Language), typically VHDL (IEEE 1076) or Verilog (IEEE 1364).

The RTL description needs to be synthesizable, which means that it can be translated automatically into a set of interconnected electronic gates. To achieve this property, the designer uses only a subset of the HDL language, while the full language may be used for example to create simulation environments.

In parallel to the design, it is of use to develop a "test-bench" with the same language: the RTL description of the HPD is included in a broader HDL program, which sends it input sequences and reads its outputs in order to test it by simulation. The test-bench may use non-synthesizable language features to ease the design of the tests (e.g.: access to files, printing, explicit time management). The test-bench is then used to check the RTL description, and can be associated with various tools for test generation and coverage measurement.

Static analysis tools are being introduced to provide complementary verification approach. They typically make it possible to prove whether some expected properties hold or not on an HDL description. Examples of static analyses are: checking of properties, assertion based verification, checking of equivalence between different design levels (e.g. RTL and netlist), or Static Timing Analysis.

B.3 Implementation

Starting from the RTL description, an electronic description is produced that allows the actual implementation in the selected electronic technology. The main stages are the logic synthesis and the place and route.

The different families of components such as FPGAs, standard cells, and so on provide different pre-characterizations of the physical behaviour of the final product. Thus, while the activities described hereinafter are intrinsically necessary, they may or may not be handled automatically by the associated tools. The following description gives an overview of these activities for a design based on standard cells.

The logic synthesis transforms the RTL description into a network of logic cells of the micro-electronic technology, called "netlist". Depending on this micro-electronic technology, these cells may be only elementary gates (such as AND, OR) or may include larger functions (such as counters).

Although tools similar to software compilers are used to perform the synthesis, the designer directs the process by providing information on the expected performance (such as clock frequency, delay between two signals, power consumption) and on how critical signals such as clocks are to be handled. This information is typically stored in "constraint files" which can be very large. Their elaboration can thus be difficult, and an error or omission may result in generating a circuit suffering from subtle non-reproducible faults, almost impossible to detect by simulation. The verification of the constraint files is thus an essential activity.

The place and route stage defines the physical location of the cells on the silicon die, and inter-connects them taking into account the technological constraints (existence and capacity of predefined routing channels) as well as the application constraints (such as maximum propagation delay between two given nodes).

As the number of gates increases, more and more of them are inter-connected. So, more and more inter-connections have to be routed across the die. Additionally the requirements for speed usually impose to keep some paths short. This last constraint may lead to modifying the placement of some gates, which in turn reacts on the whole routing scheme. Finding the "best" solution is a very hard problem (in the sense of computability), so only approximations may be found by the tools, which need to use advanced and evolutionary algorithms.

The description after place and route is produced in a format which depends on the micro-electronic technology. As the physical layout is known at this stage, the propagation times may be refined by taking into account the resistance and capacitance of each path. This information is typically used to back-annotate again the description, in order to simulate it in the test-bench with realistic propagation times for cells and wiring.

Moreover, the supplier of the micro-electronic technology provides the propagation times of the cells included in its library, using formats such as VHDL-VITAL (IEEE 1076.4). This timing information is included as "back-annotation" of the netlist description, and is taken into account in the “post implementation” simulation.

In addition to the verification by "post implementation simulation”, tools for static analysis make it possible to check the propagation times (STA: Static Timing Analysis) or the equivalence between different description levels.

B.4 Types of specific integrated circuits

B.4.1 General

The evolving technology offers many variants of specific integrated circuits, so this Standard provides requirements based on principles and not on specific details of each variant.

This clause provides an overview of the main available variants (note: their names are not always used consistently in the industry).

From a theoretical point of view, any computable function can be implemented with only one type of well chosen elementary gate such as “NAND” (“*A nand B*” is “*not (A and B)*”). Therefore, the range of functions which can be implemented within a given circuit depends essentially on its size (number of gates) and on its internal connectivity which allows a more or less efficient use of the gates.

B.4.2 PAL (Programmable Array Logic)

PALs are low-size devices typically organized in OR/AND array in order to implement logic equations having the form of sum of products such as: *output = (A and B and not C) or (not B and not C) or (D).*

PALs are made specific by configuring connections, typically by blowing fuses or in some cases by configuring reprogrammable switches.

The AND structure is programmable, i.e. the product expression before programming is: (*A and not A and B and not B and C and not C, etc.*), where each term corresponds to one configurable connection. According to the functional requirement, the unneeded terms are removed to produce e.g. (*A and not C*).

The OR structure is fixed: the inputs of the “OR” are a fixed number of such programmable products, e.g. *(A and not C) or (A and not B) or (D).*

Low-level languages such as PALASM are typically used to configure PALs: the designer inputs the logic equations to be implemented and the tool translates them into a map of connections. No behavioural description such as in VHDL or Verilog is possible with such languages.

PALs typically provide a few inputs and outputs (e.g. 10 inputs, 8 outputs) and they are equivalent to a few hundreds gates at most. Due to this limited size they are not in the scope of this Standard.

B.4.3 PLD, CPLD (Programmable Logic Device, Complex PLD)

PLDs and CPLDs are essentially large arrays of PALs interconnected together, but new families may offer additional features.

Like PALs, they are based on sum of products with fixed structure, thus the signal routing from input to output is fixed and propagation delay times are quite constant. Of course, when additional features such as feedback paths or specialized logic are offered this property may be lost.

The size of CPLDs reaches the equivalent of tens of thousands gates.

B.4.4 FPGA

FPGAs are organized as a large number of programmable logic blocks including provisions for combinatorial logic and storage. These blocks are interconnected by a hierarchy of

programmable interconnects, and programmable input/output pads are provided (direction, impedance, voltage, and memorization are typically programmable). Specific paths are usually provided for critical signals such as clocks. FPGAs may in addition include specialized logic blocks such as memory, processor core, standardized interfaces, etc.

The gate equivalence of FPGAs is not really relevant because their complex and different structures make it difficult to predict how many blocks are needed for a given function. Some FPGAs include hundreds of thousands of programmable blocks, hundreds of inputs/outputs, and are made of billions of transistors.

FPGAs may retain their function (“configuration”) by using means such as:

a) Static RAM (the configuration is volatile, copied at start-up from an external memory),

b) Flash memory (the configuration is stored in non-volatile but reprogrammable internal memory elements),

c) Anti-fuse (the configuration is permanent; such devices are “One Time Programmable”).

The susceptibility of the configuration to SEU (Single Event Upset) and neutron/alpha radiation is high for Static RAM, low for Flash, and very low for anti-fuse parts.

B.4.5 Gate array, or pre-diffused integrated circuit

The integrated circuit supplier prepares in advance standard integrated circuits in which all transistors are already made but are not interconnected. The specific function to be implemented is synthesized into a specific interconnection of the transistors.

This approach involves non-recurring costs associated with the production of the specific masks for the metal layers (interconnection), but may offer a lower part cost compared to FPGAs because no silicon is used to implement the programmable circuitry. However, this technology seems to be increasingly replaced by FPGAs.

B.4.6 Standard cells

The supplier offers a micro-electronic technology and designs with it a range of standard cells such as elementary combinatorial gates, flip-flops, adders, counters, etc. These cells have known characteristics such as area, input current, capacitance and propagation delay. They are designed in such a way that they have the same height and different width, so they can be placed on the integrated circuit in rows in order to ease routing and power supplying.

The functional and physical characteristics of the cells are described in the technology library, which is provided to the I&C designer. This library is used during logic synthesis (see Clause B.3) which transforms the RTL description into a netlist of these cells, which are then placed on the integrated circuit and interconnected. After completion of functional and technology related verifications, the masks needed to produce the integrated circuits are fabricated and production may begin.

This approach involves higher non-recurring costs compared to gate-arrays because all masks are specific, but offers a lower part cost because the size of the integrated circuit is exactly what is needed. The availability of different cells for each type, optimizing different aspects such as speed, area, or power consumption, allows a better optimization of each area of the design, still under control of the I&C designer with only HDL related tools.

B.4.7 Full custom ASIC, or raw ASIC

This technology involves a specific design of all aspects of the integrated circuit, down to the transistor level, with specific tools. This implies very high non-recurring costs which need large volumes to be economically justified. These circuits are not in the scope of this Standard.

Bibliography

IEC 60780, *Nuclear power plants – Electrical equipment of the safety system – Qualification*

IEC 61226, *Nuclear power plants – Instrumentation and control important to safety – Classification of instrumentation and control functions*

IEC 62342, *Nuclear power plants – Instrumentation and control systems important to safety – Management of ageing*

ISO 9001, *Quality management systems – Requirements*

ISO/IEC 25000, *Software engineering – Software product Quality Requirements and Evaluation (SQuaRE)*

_____

SOMMAIRE

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

_____

CENTRALES NUCLÉAIRES DE PUISSANCE – INSTRUMENTATION ET CONTRÔLE-COMMANDE IMPORTANTS POUR LA SÛRETÉ – DÉVELOPPEMENT DES CIRCUITS INTÉGRÉS PROGRAMMÉS EN HDL POUR LES SYSTÈMES RÉALISANT DES FONCTIONS DE CATÉGORIE A

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
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 brevet. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale CEI 62566 a été établie par le sous-comité 45A: Instrumentation et contrôle-commande des installations nucléaires, du comité d'études 45 de la CEI: Instrumentation nucléaire.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
45A/859/FDIS	45A/865/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 stabilité indiquée sur le site web de la CEI sous "http://webstore.iec.ch" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

INTRODUCTION

a) Contexte technique, questions importantes et structure de la norme

Les systèmes électroniques de classe 1 (selon la CEI 61513) employés dans les centrales nucléaires de puissance et qui sont nécessaires dans les situations d'urgence doivent être entièrement validés et qualifiés avant d'être utilisés en phase d'exploitation.

Dans les systèmes programmés classiques, on peut distinguer le matériel du logiciel. Le matériel est principalement conçu avec des composants standardisés remplissant des fonctions électroniques prédéfinies tels que des microprocesseurs, des temporisateurs ou encore des contrôleurs de réseau, alors que le logiciel est utilisé pour coordonner les différentes parties du matériel et pour réaliser les fonctions de l'application nucléaire.

Aujourd'hui, les concepteurs d'instrumentation et de contrôle-commande (I&C) peuvent bâtir des fonctions d'application directement à l'intérieur d'un circuit intégré, en utilisant des circuits tels que les FPGA ou des technologies similaires. La fonction d'un tel circuit intégré n'est pas définie par le fournisseur du composant physique ou de la technologie micro-électronique, mais par le concepteur d'instrumentation et de contrôle-commande.

Les circuits intégrés traités dans la présente norme sont:

1) basés sur des ressources micro-électroniques prédéveloppées,
2) développés au sein d'un projet d'I&C,
3) développés au moyen de Langages de Description de Matériel (HDL) et d'outils associés, utilisés pour réaliser les exigences par un assemblage adéquat des ressources micro-électroniques prédéveloppées.

Par conséquent, ces circuits sont nommés « circuits intégrés programmés en HDL » (HPD). Les instructions HDL qui décrivent un HPD peuvent inclure l'instanciation de Blocs Prédéveloppés (PDB) qui sont typiquement fournis sous la forme de bibliothèques, de macros, ou de blocs de Propriété Intellectuelle.

Les HPD peuvent constituer des solutions efficaces pour réaliser les fonctions requises par un projet d'I&C. Cependant, la vérification et la validation peuvent être limitées en raison du grand nombre de chemins internes et de leur observabilité limitée, si le HPD n'a pas été conçu en pensant à sa vérifiabilité.

Afin d'atteindre la fiabilité élevée exigée pour les systèmes d'I&C importants pour la sûreté, le développement des HPD doit respecter des exigences de procédé et des exigences techniques strictes, telles que celles indiquées dans la présente norme, concernant notamment la spécification des exigences, la sélection des circuits intégrés vierges et des PDB, la conception et la réalisation, la vérification, et les procédures de fonctionnement et de maintenance.

La présente norme est destinée aux concepteurs de matériel, aux opérateurs de centrales nucléaires de puissance (producteurs d'électricité), et aux autorités de sûreté. Les organismes réglementaires y trouveront des recommandations pour évaluer des aspects importants comme la conception, la réalisation, la vérification et la validation des HPD.

b) Position de la présente norme dans la collection de normes du SC 45A de la CEI

La CEI 61513 est un document de premier niveau de la collection des normes du SC 45A de la CEI et fournit des recommandations applicables à l'I&C au niveau du système. Elle est complétée par des recommandations au niveau matériel (CEI 60987) et logiciel (CEI 60880 et CEI 62138). La CEI 62340 fournit des exigences visant à réduire et surmonter la possibilité d'une défaillance de cause commune de fonctions de catégorie A.

La CEI 62566 est un document de deuxième niveau de la collection des normes du SC 45A de la CEI qui concerne les activités de développement des HPD. Elle complète la CEI 60987 qui aborde les problèmes génériques de la conception du matériel des systèmes informatisés. Elle renvoie à la CEI 60880 quand des questions identiques à celles du développement des logiciels sont traitées.

Pour de plus amples détails sur la structure de la collection des normes du SC 45A de la CEI, voir le point d) de cette introduction.

c) Recommandations et limites relatives à l'application de cette norme

Il est important de noter que la présente norme n'établit pas d'exigence fonctionnelle supplémentaire concernant les systèmes de sûreté.

Les aspects pour lesquels des exigences et des recommandations particulières ont été produites sont les suivants:

1) approche de spécification des exigences, de conception, de réalisation et de vérification des circuits intégrés programmés en HDL (HPD, voir 3.7), ainsi que des aspects de l'intégration et de la validation du système liés aux HPD;
2) approche d'analyse et de sélection des circuits intégrés vierges, technologies micro-électroniques et Blocs Prédéveloppés (PDB, voir 3.11) utilisés pour développer les HPD;
3) procédures de modification et de contrôle de configuration des HPD;
4) exigences relatives à la sélection et à l'utilisation des outils logiciels utilisés pour développer les HPD.

Il est reconnu que les techniques numériques se développent à un rythme soutenu, et qu'il n'est pas possible pour une norme de faire référence à toutes les techniques nouvelles de conception.

Pour garantir la pertinence de la présente norme dans les années futures, l'accent a été mis sur les principes plutôt que sur des technologies spécifiques. Si de nouvelles techniques apparaissent, il devrait être possible d'évaluer leur adéquation en appliquant les principes de sûreté contenus dans la présente norme.

d) Description de la structure de la collection des normes du SC 45A de la CEI et relations avec d'autres documents de la CEI et d'autres organisations (AIEA, ISO)

Le document de niveau supérieur de la collection de normes produites par le SC 45A de la CEI est la norme CEI 61513. Cette norme traite des exigences relatives aux systèmes et équipements d'instrumentation et de contrôle-commande (systèmes d'I&C) utilisés pour accomplir les fonctions importantes pour la sûreté des centrales nucléaires, et structure la collection de normes du SC 45A de la CEI.

La CEI 61513 fait directement référence aux autres normes du SC 45A de la CEI traitant de sujets génériques, tels que la catégorisation des fonctions et le classement des systèmes, la qualification, la séparation des systèmes, les défaillances de cause commune, les aspects logiciels et les aspects matériels relatifs aux systèmes programmés, et la conception des salles de commande. Il convient de considérer que ces normes, de second niveau, forment, avec la norme CEI 61513, un ensemble documentaire cohérent.

Au troisième niveau, les normes du SC 45A de la CEI, qui ne sont généralement pas référencées directement par la norme CEI 61513, sont relatives à des matériels particuliers, à des méthodes ou à des activités spécifiques. Généralement ces documents, qui font référence aux documents de deuxième niveau pour les activités génériques, peuvent être utilisés de façon isolée.

Un quatrième niveau qui est une extension de la collection de normes du SC 45A de la CEI correspond aux rapports techniques qui ne sont pas des documents normatifs.

La CEI 61513 a adopté une présentation similaire à celle de la publication fondamentale de sécurité CEI 61508, avec un cycle de vie de sûreté d'ensemble et un cycle de vie de sûreté des systèmes. Au niveau sûreté nucléaire, elle est l'interprétation des exigences générales de la CEI 61508-1, de la CEI 61508-2 et de la CEI 61508-4 pour le secteur nucléaire. Dans ce domaine, la CEI 60880 et la CEI 62138 correspondent à la CEI 61508-3 pour le secteur nucléaire.

La CEI 61513 fait référence aux normes ISO ainsi qu'aux documents AIEA GS-R-3 et AIEA GS-G-3.1 pour ce qui concerne l'assurance qualité.

Les normes produites par le SC 45A de la CEI sont élaborées de façon à être en accord avec les principes de sûreté fondamentaux du Code AIEA sur la sûreté des centrales nucléaires, ainsi qu'avec les guides de sûreté de l'AIEA, en particulier avec le document d'exigences NS-R-1 qui établit les exigences de sûreté relatives à la conception des centrales nucléaires et avec le guide de sûreté NS-G-1.3 qui traite de l'instrumentation et du contrôle-commande importants pour la sûreté des centrales nucléaires. La terminologie et les définitions utilisées dans les normes produites par le SC 45A sont conformes à celles utilisées par l'AIEA.

CENTRALES NUCLÉAIRES DE PUISSANCE – INSTRUMENTATION ET CONTRÔLE-COMMANDE IMPORTANTS POUR LA SÛRETÉ – DÉVELOPPEMENT DES CIRCUITS INTÉGRÉS PROGRAMMÉS EN HDL POUR LES SYSTÈMES RÉALISANT DES FONCTIONS DE CATÉGORIE A

1 Domaine d'application et objet

1.1 Considérations générales

La présente Norme internationale énonce des exigences pour atteindre une fiabilité élevée dans les « circuits intégrés programmés en HDL » (HPD) destinés aux systèmes d'I&C des centrales nucléaires de puissance réalisant des fonctions de sûreté de catégorie A telles que définies par la CEI 61226.

La programmation des HPD repose sur des Langages de Description de Matériel (HDL) et des outils logiciels associés. Ils sont typiquement basés sur des FPGA vierges ou des technologies micro-électroniques similaires. Les circuits intégrés d'usage général tels que les microprocesseurs ne sont pas des HPD.

La présente norme énonce des exigences sur:

a) un cycle de vie de développement dédié concernant chaque phase du développement des HPD, notamment la spécification des exigences, la conception, la réalisation, la vérification, l'intégration et la validation,

b) la planification et des activités complémentaires telles que la modification et la production,

c) la sélection des composants prédéveloppés, notamment les ressources micro-électroniques (telles que FPGA ou CPLD vierges) et les instructions HDL représentant des Blocs Prédéveloppés (PDB),

d) l'utilisation de principes de simplicité et de déterminisme reconnus pour leur importance dans l'atteinte d'une réalisation « exempte de défauts » des fonctions de catégorie A,

e) les outils utilisés pour concevoir, réaliser et vérifier les HPD.

La présente norme n'impose pas d'exigence sur le développement des ressources micro-électroniques, qui sont généralement disponibles dans le commerce sous forme d'éléments « sur étagère », et ne sont pas développées selon des normes d'assurance qualité nucléaire. Elle concerne les développements effectués à partir de ces ressources micro-électroniques dans un projet d'I&C, avec des HDL et des outils associés.

La présente norme fournit des recommandations visant à éviter autant que possible les défauts latents résiduels dans les HPD, et à réduire la susceptibilité aux simples défauts et aux défaillances de cause commune (DCC) potentielles. Les exigences de la présente norme pour une documentation claire et complète devraient faciliter l'application efficace de la CEI 62340.

Les aspects de la fiabilité liés à la qualification environnementale et aux défaillances dues au vieillissement ou à la dégradation physique ne sont pas abordés dans la présente norme. D'autres normes traitent de ces aspects, en particulier la CEI 60987, la CEI 60780 et la CEI 62342.

Le paragraphe 5.7 de la CEI 60880:2006 contient des exigences au sujet de la sécurité qui concernent le développement des HPD lorsqu'elles sont applicables.

1.2 Utilisation de la présente norme

La présente norme énonce des lignes directrices et des exigences pour des conceptions et réalisations vérifiables, lorsque qu'une justification est nécessaire par exemple en raison de la fonction exécutée ou de l'importance pour la sûreté de son comportement. Les systèmes d'I&C de classe 1 peuvent utiliser des HPD ne nécessitant pas une démonstration complète de conformité aux exigences de la présente norme, par exemple lorsqu'ils n'implémentent pas la logique d'une fonction de sûreté. Toutefois, il convient que les écarts par rapport aux exigences de la présente norme soient justifiés.

La présente norme décrit les activités visant à développer les HPD, organisées en un cycle de vie dédié. Elle décrit également les activités et les recommandations à suivre en complément des exigences de la CEI 61513 pour l'intégration et la validation des systèmes lorsqu'ils incluent des HPD.

Les exigences de la CEI 60987 relatives au développement de dispositifs logiques programmables sont applicables, en plus de celles de la présente norme, pour les HPD inclus dans des systèmes d'I&C de classe 1.

NOTE En cas d'exigences contradictoires, celles de la présente norme remplacent celles de la CEI 60987 pour les HPD de classe 1.

2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

CEI 60671, *Centrales nucléaires de puissance – Systèmes d'instrumentation et de contrôle-commande importants pour la sûreté – Essais de surveillance*

CEI 60880:2006, *Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Aspects logiciels des systèmes programmés réalisant des fonctions de catégorie A*

CEI 60987:2007, *Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Exigences applicables à la conception du matériel des systèmes informatisés*

CEI 61513:2011, *Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Exigences générales pour les systèmes*

CEI 62138, *Centrales nucléaires – Instrumentation et contrôle-commande importants pour la sûreté – Aspects logiciels des systèmes informatisés réalisant des fonctions de catégorie B ou C*

CEI 62340, *Centrales nucléaires de puissance – Systèmes d'instrumentation et de contrôle-commande importants pour la sûreté – Exigences permettant de faire face aux défaillances de cause commune (DCC)*

AIEA guide NS-G-1.3:2005, *Systèmes d'instrumentation et de contrôle-commande importants pour la sûreté des centrales nucléaires*

3 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.1
circuit intégré spécifique, ASIC
circuit intégré conçu pour des applications spécifique

[CEI 60050-521:2002, 521-11-18]

NOTE Circuit intégré spécialisé conçu pour les objectifs d'une société. Il intègre des fonctions sur mesure définies par cette société.

3.2
bloc
une des parties constituant une conception; un bloc peut se décomposer en d'autres blocs

NOTE Un bloc est soit un Bloc Prédéveloppé, soit un Bloc Natif, soit un bloc développé au cours du projet considéré.

3.3
défaillance de cause commune, DCC
défaillance d'au moins deux structures, systèmes ou composants due à un seul évènement ou une seule cause spécifique

[AIEA Glossaire de Sûreté 2007]

NOTE Les causes communes peuvent être internes ou externes au système d'I&C.

[CEI 61513]

3.4
niveau système électronique, ESL
description de haut niveau d'un système électronique, basée sur un ensemble de processus représentant les fonctionnalités de composants tels que des microprocesseurs, des mémoires, des unités de calcul spécialisées, ou des canaux de transmission

NOTE Cette description permet au concepteur de répartir le système en composants, d'évaluer ses performances pour différentes affectations de fonctions aux composants, et d'établir les exigences relatives aux composants.

Elle est typiquement réalisée avec des langages tels que SystemC (IEEE 1666), SystemVerilog (IEEE 1800), ou Matlab (R).

3.5
réseau de portes programmable sur site, FPGA
circuit intégré qui peut être programmé sur site par le fabricant de contrôle-commande. Il comprend des blocs logiques programmables (combinatoires et séquentiels), des interconnexions programmables entre ceux-ci, et des blocs programmables pour les entrées et/ou les sorties. La fonction est ensuite définie par le concepteur du contrôle-commande, et non par le fabricant du circuit intégré.

NOTE Bien que les FPGA soient essentiellement des dispositifs numériques, certains peuvent inclure des entrées et sorties analogiques ainsi que des convertisseurs de signaux analogiques en numérique. Les FPGA peuvent inclure des fonctions numériques avancées telles que des multiplieurs, des mémoires dédiées et des cœurs de microprocesseurs.

3.6
langage de description de matériel, HDL
langage permettant de décrire formellement les fonctions et/ou la structure d'un composant électronique, à des fins documentaires, de simulation ou de synthèse

NOTE Les HDL les plus utilisés sont VHDL (IEEE 1076) et Verilog (IEEE 1364).

3.7
circuit intégré programmé en HDL, HPD
circuit intégré configuré (pour des systèmes d'I&C de centrales nucléaires de puissance) avec des HDLs et outils associés.

NOTE 1 Les HDL et outils associés (par exemple simulateur, synthétiseur) sont utilisés pour réaliser les exigences par un assemblage adéquat de ressources micro-électroniques prédéveloppées.

NOTE 2 Le développement de HPD peut utiliser des Blocs Prédéveloppés.

NOTE 3 Les HPD sont typiquement basés sur des FPGA ou des technologies micro-électroniques similaires.

3.8
module
une des parties constituant une conception; un module peut se décomposer en d'autres modules

NOTE « Module » est synonyme de « Bloc »; « Bloc » est souvent utilisé dans le contexte de la conception électronique. « Module » est nécessaire dans la présente norme pour référencer des exigences de la CEI 60880 qui l'utilisent.

3.9
bloc natif
bloc représentant une ressource préexistante du circuit intégré, par exemple une porte OU ou un bloc plus complexe tel qu'un multiplieur ou un contrôleur de transmission série. La programmation du HPD configure et connecte les Blocs Natifs pour réaliser la fonction requise.

3.10
liste d'interconnexions (Netlist)
description d'un composant électronique en termes d'interconnexions entre ses éléments terminaux (c'est-à-dire les Blocs Natifs).

3.11
bloc prédéveloppé, PDB
bloc fonctionnel prédéveloppé utilisable dans une description en HDL

NOTE 1 Les PDB sont typiquement fournis sous la forme de bibliothèques, de macros, ou de blocs de Propriété Intellectuelle. Ils sont utilisés pour le développement du HPD et incorporés dans celui-ci.

NOTE 2 L'incorporation d'un PDB dans un HPD peut nécessiter un travail significatif, par exemple la synthèse d'un circuit électronique à partir de ses instructions HDL, le placement des composants de ce circuit sur les structures matérielles du circuit intégré physique et le routage des interconnexions.

3.12
logiciel prédéveloppé, PDS
logiciel qui existe déjà, qui peut ou non être un produit commercial, et dont l'utilisation est envisagée

[CEI 60880]

3.13
réseau logique programmable, PLD
circuit intégré composé d'éléments logiques avec un motif d'interconnexions, dont des parties sont programmables par l'utilisateur.

[CEI 60050-521:2002, 521-11-01]

NOTE 1 Différents types de PLD existent, par exemple les EPLD (PLD effaçables), et les CPLD (PLD complexes).

NOTE 2 Les différences entre FPGA et PLD ne sont pas strictement définies, mais PLD désigne habituellement un dispositif plus simple que FPGA.

3.14
niveau transfert de registre, RTL
modèle parallèle synchrone d'un circuit électronique, décrivant son comportement au moyen de signaux traités selon une logique combinatoire et transférés entre registres sur des impulsions d'horloge. Le modèle RTL est typiquement écrit en HDL ou généré à partir d'un code source HDL.

4 Symboles and abréviations

ASIC: Circuit intégré spécifique (Application Specific Integrated Circuit)
DCC: Défaillance de cause commune
CPLD: Réseau logique programmable complexe (Complex Programmable Logic Device)
DRC: Contrôle des règles de conception (Design Rule Check)
ESL: Niveau système électronique (Electronic System Level)
FPGA: Réseau de portes programmable sur site (Field Programmable Gate Array)
HDL: Langage de description de matériel (Hardware Description Language)
HPD: Circuit intégré programmé en HDL (HDL-Programmed Device)
IP: Propriété Intellectuelle (Intellectual Property)
I&C: Instrumentation et contrôle-commande (Instrumentation and Control)
PAL: Réseau logique programmable (Programmable Array Logic)
PDB: Bloc prédéveloppé (Pre-Developed Block)
PDS: Logiciel prédéveloppé (Pre-Developed Software)
PLD: Réseau logique programmable (Programmable Logic Device)
RAM: Mémoire vive (Random Access Memory)
RTL: Niveau transfert de registre (Register Transfer Level)
SEU: Défaut transitoire dû à une particule unique (Single Event Upset)
SRAM: Mémoire vive statique (Static RAM)
STA: Analyse temporelle statique (Static Timing Analysis)
VHDL: HDL pour circuits intégrés très rapides (Very High Speed Integrated Circuit HDL)
V&V: Vérification et validation

5 Exigences générales pour les projets HPD

5.1 Considérations générales

Cet article commence par situer le HPD dans le système d'I&C décrit par la CEI 61513, puis il décrit le cycle de vie du développement du HPD qui structure le projet HPD.

Enfin, il énonce des exigences pour les projets HPD, pour l'assurance qualité et pour la gestion de configuration. Ces sujets étant semblables dans les processus de développement de logiciel, les exigences sont définies par renvoi aux paragraphes adéquats de la CEI 60880 complétés au besoin par des exigences spécifiques aux HPD.

Le domaine d'application de la présente norme, défini à l'Article 1, exclut le développement des technologies micro-électroniques et des circuits intégrés vierges. En conséquence, les formulations comme « développement du HPD », « cycle de vie du HPD », « conception du HPD » ou « vérification du HPD » désignent ce qui est effectué au sein du projet d'I&C, en partant de ces technologies micro-électroniques ou de ces circuits intégrés vierges, pour réaliser le circuit intégré spécifique destiné à être utilisé dans le système d'I&C.

5.2 Cycle de vie

Le processus de réalisation des systèmes d'I&C destinés aux centrales nucléaires de puissance est précisé dans la CEI 61513 qui introduit le concept de cycle de vie de sûreté d'un système. Ce cycle de vie de sûreté est un moyen de contrôler le processus de développement, et son adoption permet également d'obtenir les preuves nécessaires à la justification du fonctionnement correct des systèmes de sûreté. Il définit des exigences sur la production de systèmes, mais n'impose pas d'organisation spécifique du projet (voir Figure 1).

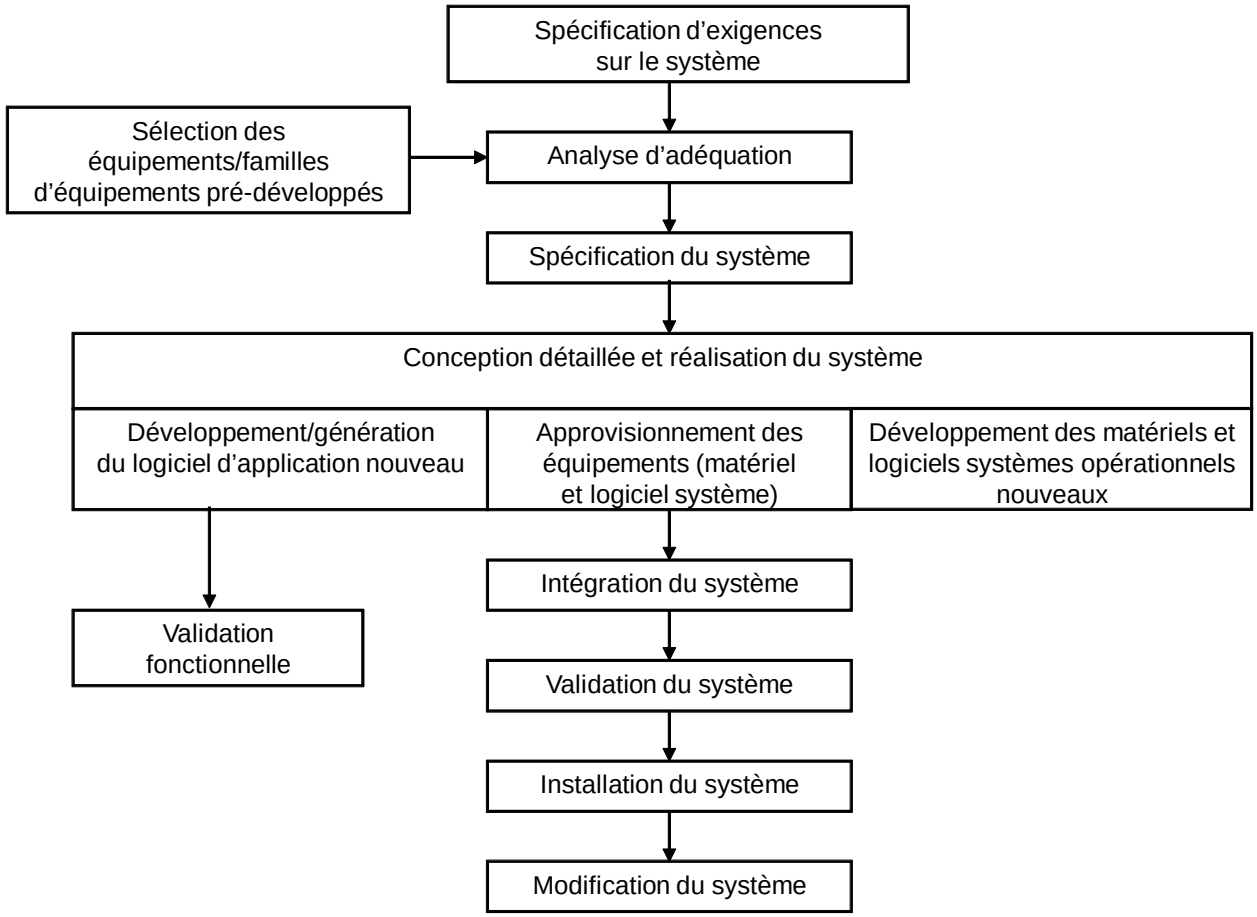

Figure 1 – Cycle de vie de sûreté du système (informatif, tel que défini par la CEI 61513)

Le cycle de vie de sûreté d'un système de la CEI 61513 est complété dans la CEI 60880 (pour les fonctions de catégorie A) et dans la CEI 62138 (pour les fonctions de catégories B et C) pour le développement des logiciels, et dans la CEI 60987 pour le développement du matériel des systèmes programmés. Les exigences de la présente norme s'appliquent au développement des HPD dans les systèmes de classe 1, en plus de celles de la CEI 60987.

NOTE En cas d'exigences contradictoires, celles de la présente norme remplacent celles de la CEI 60987 pour les HPD de classe 1.

Le développement des HPD fait appel à des outils informatiques qui tendent à structurer le processus de développement selon un cycle comportant des activités dédiées à la capture des exigences, à la conception et à la réalisation, à l'intégration et à la validation, combinées aux activités de vérification et de test.

Les phases de conception et de réalisation d'un système de la CEI 61513 présentées en Figure 1 (notamment « Approvisionnement des équipements (matériel et logiciel système) » et « Développement des nouveaux matériels et logiciels systèmes opérationnels», constituent

des phases essentielles du cycle de vie de sûreté système de la CEI 61513. Ces phases sont détaillées à la Figure 2 pour préciser les phases qui se situent entre la spécification des exigences et la validation pour les composants du système qui sont des HPD.

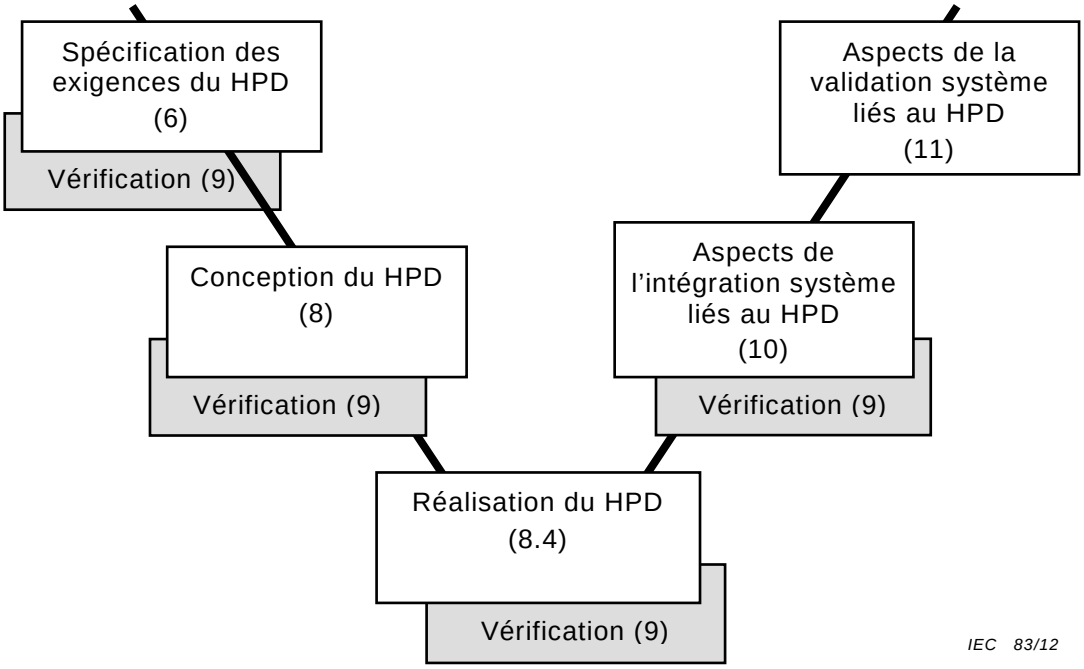

Figure 2 – Cycle de vie de développement du HPD

Les concepteurs ont généralement recours à des éléments prédéveloppés tels que des circuits intégrés programmables ou des Blocs Prédéveloppés (PDB) pour réaliser les circuits intégrés spécifiquement adaptés aux besoins du projet. Les activités dédiées à la sélection de ces éléments prédéveloppés sont traitées à l'Article 7 de la présente norme. Elles peuvent être effectuées parallèlement aux premières phases du cycle de vie décrit à la Figure 2, sous réserve que toutes les dépendances soient formellement gérées et documentées.

Le cycle de vie présenté en Figure 2 concerne le développement d'un HPD, qui peut être effectué parallèlement à celui d'autres composants (logiciels ou matériels) du système comme le montre la Figure 1. L'ensemble de ces développements convergent lors des phases d'intégration et de validation du cycle de vie du système.

L'approche de développement proposée est basée sur le modèle traditionnel du « cycle en V » car cette approche a été adoptée dans d'autres normes et elle est recommandée par le guide NS-G 1.3 de l'AIEA, mais elle autorise les ajustements nécessaires en reconnaissant que certaines phases du développement peuvent être effectuées automatiquement par des outils et que ce développement peut être itératif.

Il n'existe souvent pas de frontière distincte et bien identifiée entre l'intégration d'un composant donné et l'intégration du système. Par conséquent, dans la présente norme, on considère que l'intégration d'un HPD fait partie de l'intégration du système. De même, sa validation fait partie de la validation du système.

Selon la fonction assurée par le HPD, le système ou sous-système à considérer lors de l'intégration peut aller:

1) du système d'I&C si le HPD réalise la logique d'une fonction de sûreté,
2) à une carte électronique ou à une armoire s'il réalise une fonction (interne à la carte ou à l'armoire) dont une analyse adéquate a démontré qu'elle ne peut affecter les sorties d'aucune fonction de sûreté du système environnant.

La situation généralement la plus critique du point de vue de la sûreté apparaît quand le HPD réalise directement la logique d'une fonction de sûreté.

Les activités suivantes viennent en appui du processus de développement du HPD:

a) gestion de projet (voir 5.3),

b) assurance qualité et contrôle de la qualité (voir 5.4),

c) gestion de configuration (voir 5.5),

d) vérification (voir Article 9).

D'autres activités concernent la sélection d'outils d'aide au développement (voir Article 15), la production de la documentation (voir Annexe A), et la gestion des modifications (voir Article 12).

5.3 Gestion du projet HPD

5.3.1 Considérations générales

5.3.1.1 Chaque HPD doit être développé au sein d'un projet HPD dédié.

5.3.1.2 Le projet HPD doit respecter les exigences de 5.4 de la CEI 60880:2006 (en remplaçant "logiciel" par "HPD").

NOTE 1 Une liste typique de documents requis au long du cycle de vie est fournie en Annexe A de la présente norme.

NOTE 2 Les entrées à documenter selon 5.4.6 de la CEI 60880:2006 incluent les paramètres des activités automatisées des outils logiciels (par exemple: optimisation temporelle, optimisation de la densité, etc.).

5.3.1.3 Le processus de développement peut être itératif; une phase peut débuter avant l'achèvement des activités de la phase précédente, mais ne doit se terminer que si les phases précédentes sont achevées et si ses sorties sont cohérentes avec les entrées fournies par les activités de ces phases précédentes.

5.3.1.4 Les phases du projet HPD doivent inclure la spécification des exigences, la conception et la réalisation du HPD.

5.3.2 Autres exigences

5.3.2.1 La sélection des éléments prédéveloppés utilisés dans le projet doit être effectuée en respectant les exigences de l'Article 7 de la présente norme.

5.3.2.2 Les critères de transition d'une phase à l'autre doivent être définis.

5.3.2.3 Les critères d'achèvement des phases doivent avoir un contenu méthodologique et technique, et être suffisamment détaillés pour que leur évaluation nécessite une analyse en profondeur des réalisations de la phase.

5.3.2.4 La documentation (voir 5.4.11 de la CEI 60880:2006) doit inclure la description des fonctions réalisées par le HPD et de ses interfaces.

5.4 Plan d'assurance qualité pour le HPD

Un plan d'assurance qualité pour le HPD doit exister et doit respecter les exigences de 5.5 de la CEI 60880:2006 (en remplaçant « logiciel » par « HPD »).

NOTE Dans ce cadre, « langage » signifie « langage informatique ».

5.5 Gestion de configuration

5.5.1 La gestion de configuration du HPD doit respecter les exigences de 5.6 de la CEI 60880:2006 (en remplaçant « logiciel » par « HPD »).

NOTE La séparation requise par l'article 5.6.6 de la CEI 60880:2006 s'applique à la documentation et aux fichiers d'ordinateur utilisés ou produits par le projet HPD.

5.5.2 La gestion de configuration doit enregistrer les éléments suivants:

a) documentation des modules (blocs) développés dans le projet et des PDB,

b) marquage identifiant les circuits intégrés,

c) fichiers d'ordinateur utilisés pour la simulation, la vérification et la production,

d) paramètres des activités automatisées des outils logiciels (voir Article 15), comme « optimisation temporelle, optimisation de la densité » pour l'activité de placement et routage,

e) identification des versions des outils logiciels (voir Article 15) incluant les « correctifs » appliqués, et des bibliothèques générales ou liées à des technologies particulières.

6 Spécification des exigences du HPD

6.1 Considérations générales

6.1.1 Le document de spécification des exigences doit donner toutes les exigences du HPD, soit dans le document lui-même soit par renvoi à des ensembles d'exigences établies au niveau système ou sous-système (par exemple, comportement fonctionnel à implémenter).

6.1.2 La spécification des exigences doit être compréhensible par tous les participants, notamment par les concepteurs de matériel et les personnes mentionnées en 6.6.

6.1.3 La spécification des exigences doit être non équivoque, vérifiable, et réalisable, y compris pour les aspects temporels.

6.1.4 Si le HPD réalise une fonction de sûreté, sa spécification des sxigences doit découler des exigences du système d'I&C hébergeant cette fonction, et doit faire partie de la spécification du sous-système qui utilise le HPD.

6.1.5 La spécification des exigences du HPD doit décrire ce qu'il a à faire, et non la manière dont il le fait.

6.1.6 Un processus documenté, formel et permettant l'audit doit être défini et mis en œuvre pour établir la spécification des exigences.

6.1.7 La spécification des exigences doit être telle qu'il soit possible de vérifier sa conformité avec la spécification des exigences du système d'I&C. Si le HPD est utilisé par un sous-système du système d'I&C, il doit également être possible de vérifier la conformité avec la spécification de conception du système.

6.1.8 La spécification des exigences doit tenir compte de toutes les conditions de fonctionnement de la centrale jusqu'au niveau du HPD pour les fonctions concernées.

6.1.9 Les exigences d'interface avec d'autres systèmes ou composants doivent être traitées conformément à la CEI 61513.

6.1.10 Les exigences d'interface avec d'autres systèmes ou composants doivent être documentées.

6.1.11 Si elles ne font pas partie des exigences du HPD mais résultent de décisions de conception du HPD, les exigences d'interface suivantes doivent être documentées:

a) performances électriques et temporelles (par exemple impédance d'entrée, temps d'établissement et de maintien des entrées, fréquence de fonctionnement, facteur de charge des sorties, temps de propagation entre une entrée et les sorties associées),

b) profils des signaux d'interface et des alimentations électriques,

c) dissipation thermique, température de fonctionnement et refroidissement nécessaire.

6.2 Aspects fonctionnels de la spécification des exigences

Ce paragraphe décrit le contenu de la spécification des exigences directement lié aux besoins fonctionnels. Les paragraphes 6.3 and 6.4 traitent d'autres aspects devant être inclus dans la spécification des exigences.

La spécification des exigences doit préciser:

a) les fonctions devant être assurées par le HPD,

b) les différents modes de fonctionnement du HPD ainsi que les conditions de transition correspondantes, incluant la mise sous tension et l'initialisation,

c) les interfaces du HPD et ses interactions avec son environnement (opérateurs et autres composants d'I&C), incluant les rôles, protocoles, types, formats, numérotation des bits, domaines de valeur et contraintes des entrées et des sorties,

d) les paramètres du HPD qui peuvent être modifiés manuellement en cours de fonctionnement et leurs rôles,

e) les performances du HPD requises, en particulier en matière de temps de réponse,

f) ce que le HPD a l'obligation de ne pas faire ou d'éviter, lorsque c'est pertinent,

g) toute hypothèse sur l'environnement du HPD (par exemple caractéristiques électriques et temporelles des entrées et sorties, alimentations électriques, profils particuliers à la mise sous tension, refroidissement).

6.3 Conception déterministe

La spécification des exigences doit exiger que le fonctionnement du HPD soit déterministe par conception. Cela signifie que toute séquence d'entrées donnée respectant les spécifications électriques et temporelles produit toujours les mêmes sorties.

NOTE Les FPGA modernes et d'autres circuits intégrés traités par la présente norme peuvent contenir des blocs fonctionnels analogiques (par exemple convertisseurs analogique vers numérique) sujets au bruit électronique, aux erreurs de discrétisation, etc. Des variations dans les réponses de ces blocs fonctionnels analogiques dues à ces causes, ainsi que leur impacts sur la réponse du HPD, ne constituent pas des écarts à la conception déterministe.

6.4 Détection des défauts et tolérance aux fautes

Les de 5.3 et 5.4 de la CEI 60987 concernant la fiabilité du point de vue des défaillances aléatoires et de la qualification environnementale s'appliquent. Cela inclut les fautes dues aux SEU et aux particules (neutron, alpha) lorsque c'est pertinent.

La conception défensive est typiquement basée sur une combinaison de techniques (par exemple redondance, vote, contrôle de parité ou par redondance cyclique, chien de garde, contrôles de domaine et de vraisemblance).

6.4.1 La spécification des exigences doit décrire les exigences de conception défensive visant à détecter les défauts et à tolérer les fautes.

6.4.2 Il convient que les avantages de la conception défensive soient proportionnés avec la complexité induite. L'objectif global est de tenir compte de la testabilité du HPD au cours de la conception et de la réalisation, en utilisant des moyens de détection internes et externes pour atteindre une couverture élevée des défauts.

6.4.3 La spécification des exigences doit décrire les dispositions pour la détection des dysfonctionnements du HPD, en prenant en compte les dispositions déjà prises aux niveaux sous-système et système.

6.4.4 Ces dispositions peuvent nécessiter que le HPD élabore des sorties supplémentaires, soit pour exploitation par un mécanisme externe tel qu'un chien de

garde, soit pour compléter la couverture de la surveillance réalisée par un dispositif de test externe.

6.4.5 Il convient que la conception défensive permette la détection des comportements erronés (tels que la corruption de données ou les écarts par rapport à des algorithmes de traitement spécifiés, ou encore les écarts par rapport à des conditions de fonctionnement spécifiées), des transmissions de données erronées entre unités de traitement, et des modifications non souhaitées de mémoires ou de données de configuration.

6.4.6 La conception défensive ne doit pas exercer d'influence néfaste sur les fonctions du système d'I&C, ni empêcher le HPD de respecter ses exigences de temps de réponse.

6.4.7 La spécification des exigences doit décrire le comportement temporel et logique attendu (par exemple les valeurs de sortie et les informations spécifiques délivrées) quand un défaut est détecté.

6.4.8 Ce comportement doit être conforme au comportement du système demandé par la spécification du système, et avec les exigences de conception du système de la CEI 61513.

6.4.9 La spécification des exigences doit contenir et justifier l'objectif de couverture de détection des défauts à atteindre par la conception défensive.

6.5 Capture des exigences avec des outils ESL

6.5.1 Considérations générales

La présente norme ne prescrit pas de méthode spécifique pour la capture des exigences du HPD. Si elles sont capturées avec des outils au niveau système électronique (ESL, voir Article B.1), les exigences de 6.5.2 et 6.5.3 s'appliquent à ces outils et à leur utilisation.

Dans ce cas, la possible similitude entre le langage de spécification des exigences et celui utilisé pour la conception peut rendre plus difficile le respect de 6.1.5 (séparation entre ce qui est à faire (l'exigence) et la façon dont cela est fait (la conception)). Des dispositions peuvent être nécessaires pour le respecter, par exemple des commentaires pour spécifier les entrées, les sorties et les algorithmes.

6.5.2 Exigences relatives au formalisme des outils ESL

6.5.2.1 Si les exigences du HPD sont capturées à l'aide d'un outil ESL:

a) cet outil doit offrir un formalisme doté d'une sémantique rigoureuse et compréhensible (standardisation de la structure et de la présentation, modularité, commentaires pertinents);

b) le formalisme de l'outil doit être compréhensible par tous les participants;

c) si l'outil propose des mécanismes flexibles pour redéfinir les fonctions et opérateurs, il convient que les caractéristiques réelles de tout élément donné soient claires pour tous les participants, y compris les concepteurs du matériel et les autres personnes mentionnées en 6.6.

6.5.2.2 Il convient que les langages ESL utilisés permettent la prise en compte de l'architecture du système, par exemple en autorisant l'assignation de fonctions à des composants, et supportent les caractéristiques de conception tolérantes aux fautes.

6.5.3 Interface avec les outils de conception

La sémantique du langage utilisé pour exprimer la spécification des exigences au niveau ESL peut différer de celle du langage HDL employé pour la conception. Des exemples d'écarts

possibles sont l'interprétation du parallélisme, la gestion des dépassements de capacité ou le codage des types et des machines à états finis.

a) Si la sémantique du langage utilisé pour exprimer la spécification des exigences au niveau ESL diffère de la sémantique des autres langages employés dans le projet, les écarts doivent être identifiés pour chaque élément concerné de la spécification des exigences;

b) chaque occurrence d'écart au sein de la spécification des exigences doit être documentée. Une liste générique d'écarts entre les langages concernés peut constituer une référence utile, mais ne suffit pas à clarifier la spécification des exigences.

6.6 Analyse et revue des exigences

6.6.1 Une analyse critique des exigences doit être effectuée et documentée, pour trouver les éventuelles incohérences, lacunes et ambiguïtés.

6.6.2 La portée de cette analyse doit inclure les exigences fonctionnelles et les autres, y compris celles relatives aux comportements non nominaux tels que les valeurs d'entrée ou les séquences imprévues.

6.6.3 La spécification des exigences doit être revue pour vérifier son exhaustivité et sa cohérence.

6.6.4 Pour les fonctions de sûreté exécutées par le HPD, les ingénieurs spécialistes du procédé et de l'I&C, ainsi que les spécialistes des sous-systèmes ou des composants (y compris logiciels) qui sont interfacés avec le HPD, doivent participer à la revue.

7 Processus d'acceptation des circuits intégrés programmables, des blocs natifs et des blocs prédéveloppés

7.1 Considérations générales

Développer un HPD nécessite de choisir et d'évaluer des éléments prédéveloppés tels qu'un circuit intégré vierge (et ses blocs natifs) ou des PDB incorporés dans le HPD final.

Comme ces éléments prédéveloppés (ou composants) peuvent comporter des caractéristiques non requises pour le HPD, des "règles d'utilisation" spécifiques peuvent devoir être élaborées et appliquées afin de limiter leur utilisation à ce qui est nécessaire et sûr.

7.2 Spécification des exigences du composant

7.2.1 Considérations générales

Les exigences assignées aux éléments prédéveloppés (ou composants) proviennent des activités de conception initiale du HPD. Par exemple, les exigences du HPD peuvent comporter un filtrage passe-bande particulier, que le concepteur peut réaliser en utilisant un bloc prédéveloppé PDB effectuant une Transformée de Fourier Rapide.

Ainsi, la spécification des exigences du composant (ici, un bloc prédéveloppé PDB effectuant une Transformée de Fourier Rapide, défini par des caractéristiques telles que le type d'algorithme, la taille des papillons, la méthode de décimation, la surface de silicium nécessaire, etc.) diffère de la spécification des exigences du HPD (ici, un filtre passe-bande défini par des fréquences de coupure, gains, pentes, etc.).

7.2.1.1 Une spécification des exigences du composant doit documenter les exigences applicables à chaque élément prédéveloppé: circuit intégré vierge, ressources micro-électroniques (vues comme des blocs natifs), outils associés le cas échéant or PDB.

7.2.1.2 La spécification des exigences du composant doit établir toutes les exigences, soit dans le document lui-même, soit par renvoi à des ensembles d'exigences établies au niveau système ou sous-système (par exemple, comportement fonctionnel à mettre en œuvre).

7.2.1.3 Constituant la base du choix et de l'utilisation du composant, la spécification des exigences du composant doit être compréhensible par tous les participants, notamment par les concepteurs de matériels et de logiciels le cas échéant, ainsi que les vérificateurs, les participants aux revues et les autorités de sûreté.

7.2.1.4 La spécification des exigences du composant doit être non équivoque, vérifiable, et réalisable, y compris pour les aspects temporels.

7.2.1.5 La spécification des exigences du composant doit être telle que la conformité du système d'I&C utilisant ce composant aux exigences de la CEI 61513 puisse être démontrée.

7.2.2 Exigences

La spécification des exigences du composant doit documenter toutes les caractéristiques attendues de l'élément prédéveloppé, notamment celles de la liste de 6.2.

NOTE Les noms génériques des caractéristiques (par exemple « fonction ») sont identiques à ceux de 6.2, mais le contenu diffère en général, comme indiqué en 7.2.

7.2.3 Analyse et revue des exigences

7.2.3.1 Une analyse critique de la spécification des exigences du composant doit être réalisée et documentée, afin de découvrir les éventuelles incohérences, lacunes ou ambiguïtés.

7.2.3.2 La portée de cette analyse doit inclure les exigences fonctionnelles et les autres, y compris celles relatives aux comportements non nominaux tels que les valeurs d'entrée ou les séquences imprévues.

7.2.3.3 La Spécification des exigences du composant doit être revue formellement par des experts de tous les domaines concernés pour vérifier son exhaustivité et sa cohérence.

7.3 Règles d'utilisation

7.3.1 Si l'élément prédéveloppé comporte des fonctions ou des modes de fonctionnement non requis dans le HPD, il convient que des règles d'utilisation soient définies pour empêcher l'utilisation de ces fonctions et modes.

L'utilisation de fonctions ou de modes de fonctionnement requis pour le HPD peut être contrôlée par des règles pour améliorer des propriétés de conception telles que la sûreté ou la testabilité.

7.3.2 Si des règles d'utilisation sont établies:

a) elles doivent être documentées,

b) le plan qualité doit garantir que leur application est vérifiée dans le cadre du projet.

7.4 Sélection

7.4.1 Considérations générales

7.4.1.1 Une analyse documentée de chaque élément prédéveloppé utilisé dans le HPD doit démontrer qu'il satisfait aux exigences de sa spécification des exigences du composant, éventuellement avec des règles d'utilisation et des modifications (voir 7.6).

7.4.1.2 Un document de sûreté utilisateur doit préciser comment les concepteurs utiliseront l'élément prédéveloppé en accord avec ses spécifications et ses caractéristiques de conception.

7.4.2 Revue de la documentation

La revue de la documentation est la méthode principale pour démontrer que l'élément prédéveloppé respecte la spécification des exigences du composant.

7.4.2.1 Il convient que cette analyse soit basée sur la documentation de l'élément prédéveloppé, en particulier la documentation de sa conception et de sa vérification.

7.4.2.2 La documentation doit être suffisamment détaillée pour démontrer le respect des exigences fonctionnelles, électriques et temporelles de l'élément prédéveloppé.

7.4.2.3 L'analyse de la documentation doit démontrer que les fonctions et les modes de l'élément prédéveloppé non utilisés dans le HPD ne perturbent pas ceux qui sont utilisés.

7.4.3 Revue de l'expérience de fonctionnement

L'expérience de fonctionnement de l'élément prédéveloppé peut être invoquée pour compenser certaines insuffisances limitées de la documentation concernant la fiabilité ou la conception. Si l'expérience de fonctionnement est invoquée:

a) l'analyse de l'expérience de fonctionnement doit démontrer que:
   1) son volume est proportionné aux exigences de fiabilité,
   2) elle a été obtenue dans des conditions de fonctionnement équivalentes à celles dans lesquelles l'élément prédéveloppé sera utilisé,
   3) l'utilisation réelle de l'élément prédéveloppé a été observée au niveau de détail généralement exigé par la présente norme pour la documentation;

b) les moyens et procédures utilisés pour collecter l'expérience de fonctionnement doivent garantir que toute défaillance de l'élément prédéveloppé lors de la période considérée est enregistrée avec suffisamment de détails pour qu'une analyse technique puisse en identifier la cause lorsque c'est possible;

c) il doit être démontré par analyse des défaillances enregistrées en cours de fonctionnement qu'elles n'ont pas d'influence sur les fonctions ou la sûreté du HPD;

d) l'expérience de fonctionnement -et au besoin des tests complémentaires- doivent démontrer que l'élément prédéveloppé respecte ses exigences;

e) une analyse technique documentée doit justifier que toutes les interactions de l'élément prédéveloppé avec son environnement sont incluses dans celles couvertes par l'expérience de fonctionnement;

f) l'expérience de fonctionnement prise en considération doit correspondre à des versions précisément identifiées de l'élément prédéveloppé et, quand celui-ci est spécifique à un équipement, de l'équipement dans lequel il fonctionne;

g) il convient que l'expérience de fonctionnement concerne la version spécifique de l'élément prédéveloppé ou de sa sous-partie utilisée dans le HPD; sinon, les différences entre les deux versions doivent être analysées pour démontrer que l'expérience de fonctionnement est pertinente pour la version envisagée.

7.4.4 Exigences particulières pour les circuits intégrés vierges

7.4.4.1 Les points suivants doivent être traités:

a) analyse de l'adéquation des mécanismes et circuiteries de programmation;

b) démonstration que le processus de programmation est exempt de défauts, ou que tout défaut dans ce processus est détecté et correctement traité;

c) démonstration que le composant conserve sa configuration programmée pendant une durée appropriée;

d) analyse des défauts potentiels dus aux composants internes et externes supplémentaires, ou aux transitoires d'alimentation et justification vis-à-vis des exigences de fiabilité.

7.4.4.2 Une analyse détaillée doit démontrer que:

a) le circuit intégré pourra respecter sa spécification des exigences du composant,

b) les outils associés

– respectent les exigences des Articles 15 et

– permettent toutes les vérifications exigées aux Articles 8 and 9 (comme l'analyse temporelle statique STA).

7.4.4.3 Les données nécessaires au calcul de taux de défauts (au sens de défauts physiques aléatoires) doivent être disponibles et basées sur une expérience de fonctionnement suffisante.

7.4.4.4 Les personnes chargées de la conception ou de la réalisation du HPD doivent posséder des connaissances appropriées:

a) du circuit intégré vierge, notamment de ses particularités de programmation, modes de configuration et de test, protocoles, broches et registres, ainsi que toute spécificité électrique ou logique, et

b) des outils associés, blocs natifs et PDB. En particulier, il convient qu'elles soient capables de prévoir, comprendre et (lorsque c'est nécessaire) contrôler les choix faits par les outils lors de la synthèse, du placement et du routage.

7.5 Justification de l'acceptation

7.5.1 Une revue formelle doit examiner l'analyse de l'élément prédéveloppé, notamment les règles d'utilisation et les dispositions prises pour garantir la conformité de chaque exemplaire physique employé en production, pour décider si l'élément prédéveloppé est accepté ou non pour utilisation dans le HPD.

7.5.2 Si l'élément prédéveloppé est accepté, toutes les dispositions et règles d'utilisation prises en compte dans l'analyse doivent s'appliquer pendant tout le cycle de vie du HPD.

7.5.3 L'équipe de revue doit comprendre des experts ayant des compétences en rapport avec les domaines concernés (par exemple technologie du matériel, logiciel) et des ingénieurs des équipes responsables des composants possédant une interface avec l'élément prédéveloppé.

7.6 Modification pour l'acceptation

7.6.1 Si des modifications de l'élément prédéveloppé sont nécessaires pour son acceptation, elles doivent être spécifiées, conçues, réalisées et vérifiées avant la revue.

7.6.2 Ces modifications doivent être effectuées et documentées conformément aux exigences de la présente norme en matière de structuration et de gestion du projet, de qualité, de spécification des exigences, de conception, de réalisation et de vérification.

7.7 Modification après l'acceptation

Le processus d'acceptation, y compris la revue, doit être réitéré après toute modification de l'élément prédéveloppé concernant sa conception ou ses aspects micro-électroniques.

7.8 Documentation d'acceptation

La documentation d'acceptation de l'élément prédéveloppé doit être placée en gestion de configuration.

7.8.1 La documentation doit inclure ou renvoyer:

a) à la spécification des exigences du HPD,

b) à tous les documents émis ou invoqués lors de l'analyse de l'élément prédéveloppé,

c) à tous les documents émis lors de la modification de l'élément prédéveloppé,

d) au rapport de revue.

La documentation doit inclure toutes les informations nécessaires pour utiliser correctement l'élément prédéveloppé, en tenant compte des contraintes découlant de sa spécification initiale, des règles d'utilisation, et des modifications.

8 Conception et réalisation du HPD

8.1 Considérations générales

Cet article précise les exigences et les recommandations issues des bonnes pratiques de conception et de réalisation afin d'obtenir un HPD ayant les caractéristiques de sûreté appropriées, aussi exempt de défaut que possible et apte à la vérification.

8.1.1 Le processus de développement doit définir une phase de conception et une phase d'implémentation.

8.2 Langages de description de matériel (HDL) et outils associés

Même si l'utilisation de langages et d'outils spécifiques ne peut pas être exigée, les points suivants peuvent être considérés comme des règles de base communes aux langages et aux outils employés pour la conception et la réalisation des HPD pour les systèmes de classe 1.

8.2.1 Il convient que la conception et la réalisation utilisent des langages de description de matériel (HDL) et des outils pour la simulation, la synthèse, le placement et le routage.

NOTE Des outils convenablement choisis et utilisés améliorent des aspects essentiels tels que l'intelligibilité des descriptions, la gestion des contraintes électriques et temporelles, la vérification, la pertinence des critères de couverture et la documentation.

8.2.2 Même si l'article 8.2.1 n'est pas respecté, tous les documents et toutes les analyses ou vérifications requises par la présente norme doivent être fournies.

8.2.3 Concernant le langage utilisé:

a) il doit suivre des règles strictes (ou bien définies) de sémantique et de syntaxe;

b) il doit avoir une syntaxe définie et documentée de façon claire et exhaustive;

c) il convient qu'il respecte une norme reconnue (par exemple IEEE 1076 pour VHDL ou IEEE 1364 pour Verilog).

8.2.4 Il convient, quand cela se justifie, de limiter l'utilisation du langage à un sous-ensemble « sûr », par exemple aux caractéristiques nécessaires pour réaliser les fonctions requises et synthétisables avec des bibliothèques normalisées (par exemple éviter l'utilisation de valeurs initiales, de retards explicites ou de divisions).

8.2.5 Le simulateur utilisé doit produire des résultats strictement conformes à la sémantique documentée du langage.

Il convient que le simulateur respecte une norme reconnue (par exemple IEEE 1076 pour VHDL ou IEEE 1364 pour Verilog).

8.2.6 A part dans le cas traité en 8.2.7, seuls des outils conformes aux exigences de l'Article 15 doivent être utilisés pour l'analyse, la simulation, la synthèse, le placement et le routage. Il n'est pas nécessaire que les utilisateurs répètent les tests des outils si ces tests ont déjà été réalisés et documentés par le fournisseur.

8.2.7 Si un outil ne respectant que partiellement les exigences de l'Article 15 est utilisé, une vérification supplémentaire des résultats produits par cet outil (par exemple liste d'interconnexions –netlist- générée par un outil de synthèse) doit démontrer que ces résultats sont corrects. Les outils de vérification formelle d'équivalence (« equivalence checking ») aident à obtenir une conception exempte de défauts.

8.3 Conception

8.3.1 Considérations générales

A partir de la spécification des exigences du HPD, la conception vise initialement à définir les principaux choix comme la décomposition en modules (spécifiques à l'application ou prédéveloppés), le fonctionnement de la conception défensive, ainsi que l'identification des technologies micro-électroniques nécessaires (y compris leurs blocs natifs) et des PDB. Ensuite, une description RTL est élaborée en utilisant des HDL. Les exigences suivantes ont pour but de parvenir à une conception claire et vérifiable.

8.3.1.1 La phase de conception doit produire a) une description formalisée du HPD, par exemple RTL et b) la documentation associée.

8.3.1.2 Les canaux de communication doivent être conçus conformément aux exigences sur les communications de données de 5.4.2.4 de la CEI 61513.

8.3.1.3 Il convient que la conception permette une vérification aisée.

8.3.1.4 Il convient de justifier les écarts par rapport aux règles de conception.

8.3.2 Conception défensive

8.3.2.1 Si un bloc natif ou un PDB utilisé (voir l'Article 7) est un cœur de processeur, il convient qu'il permette le respect des exigences de la CEI 60880 en matière d'autosurveillance.

8.3.2.2 La conception doit tenir compte des dispositions retenues dans la spécification des exigences pour détecter les défauts et pour élaborer les informations correspondantes à l'intérieur du HPD.

8.3.2.3 Lors de la détection d'un défaut, le HPD doit se comporter conformément aux exigences spécifiées.

8.3.3 Structure

8.3.3.1 Il convient de préférer une approche descendante à une approche montante.

NOTE Les bibliothèques sont les cibles ultimes de la conception. L'utilisation de bibliothèques conformes aux exigences des Articles 7 and 15 est donc en accord avec l'approche descendante et elle est conseillée.

8.3.3.2 Il convient que la structure de la conception soit basée sur une décomposition en modules. Ces modules peuvent être contenus dans une bibliothèque.

8.3.3.3 Il convient de mettre les modules génériques dans des bibliothèques.

8.3.3.4 Il convient que la structure soit simple et compréhensible, aussi bien dans sa conception générale que dans ses détails.

8.3.3.5 Il convient qu'un modèle conceptuel de l'architecture soit produit au début du projet.

8.3.4 Langage et règles de codage

8.3.4.1 Afin de faciliter une conception stable et fiable, il convient d'utiliser une méthodologie de conception éprouvée et de bonnes pratiques générales.

8.3.4.2 Afin d'améliorer la compréhensibilité de la conception et de réduire la possibilité de différences entre les comportements simulés et synthétisés:

a) un ensemble de strictes règles de codage reflétant les connaissances les plus récentes en matière de sûreté de conception et de fiabilité doit être exigées par le plan qualité et mis en place;

b) le respect de ces règles doit être assuré par des moyens appropriés (par exemple revues, outils, etc.).

8.3.4.3 La liste suivante contient des approches et techniques de conception fortement conseillées. Toutefois, la liste n'est pas considérée comme exhaustive et des parties peuvent évoluer avec la technologie. Néanmoins, toute non-conformité aux règles le la liste suivante doit être justifiée et prise en compte dans l'analyse des défaillances:

a) il convient que la conception du HPD n'utilise que des éléments synthétisables du langage. L'environnement de test et de simulation (voir 9.5) peut utiliser tous les éléments du langage. Les blocs natifs (voir 3.9) déjà synthétisés et routés dans le circuit intégré prédéveloppé peuvent être instanciés tels qu'ils sont, s'ils sont conformes à l'Article 7;

b) il convient d'utiliser lorsque c'est pertinent les ressources dédiées ou les éléments de conception fournis (par exemple arbres d'horloge prédéfinis et circuits de conditionnement d'horloge, rails d'alimentation, arbres de réinitialisation, etc.);

c) il convient que ces règles de codage couvrent tous les aspects concernés, en particulier l'appellation des modules et des signaux, l'utilisation des éléments de structuration (comme les packages, les fonctions, les procédures, les bibliothèques du projet, l'instanciation), l'organisation des traitements sur les chemins critiques, l'organisation des processus, les constructions recommandées et les constructions interdites;

d) il convient d'interdire les fonctions utilisant des effets de bords (« impures ») dans la description de la conception. (Justification: une telle fonction peut retourner des valeurs différentes quand elle est appelée plusieurs fois avec les mêmes paramètres. Elle est donc très difficile à tester et à vérifier, car elle brise le concept de fonction, et en fait de déterminisme);

NOTE 1 Une fonction impure peut aussi avoir des effets de bord comme la modification d'objets en dehors de sa portée.

e) il convient d'interdire les constructions qui pourraient induire des différences entre les comportements simulés et synthétisés. Selon le langage utilisé, de telles constructions peuvent par exemple être des affectations incomplètes ou conflictuelles, l'utilisation du caractère "peu importe" dans des comparaisons, des comparaisons (supérieures ou inférieures) impliquant des types énumérés (Justification: la simulation est une méthode de vérification importante. Si les comportements simulés et synthétisés diffèrent, la chaîne de vérification est rompue);

f) il convient d'initialiser les signaux et les variables non pas dans leur déclaration de la description RTL, mais par un mécanisme explicite tel qu'une initialisation (Justification: l'initialisation en HDL peut induire des différences entre les comportements simulés et synthétisés);

g) il convient d'interdire l'utilisation de retards explicites dans la description de la conception, car ces retards engendrent des différences entre les comportements simulés et synthétisés;

NOTE 2 Cela n'empêche pas l'existence de retards au niveau système ou dans les exigences du HPD. Cela signifie que de tels retards ne peuvent pas être réalisés par des instructions HDL comme « delay » ou « after » mais, par exemple, par des compteurs ou des registres à décalage.

h) il convient d'interdire dans la description de la conception la création de retards au moyen de portes combinatoires ou de retards dépendant des temps de propagation dans les interconnexions. Si cela ne peut être évité, une analyse temporelle statique (STA) doit

justifier l'utilisation d'une telle conception (Justification: ces retards ne sont pas stables par rapport à des paramètres tels que la température, la tension, ou d'un exemplaire du circuit intégré à l'autre, ou d'une zone du circuit intégré à une autre);

i) il convient que les types des signaux d'interface du HPD aient une définition claire et exempte d'ambiguïté, de préférence normalisée, indépendante de tout outil ou technologie micro-électronique;

j) il convient que les définitions au niveau HDL ne puissent pas être interprétées de plusieurs manières, pour éviter des variations lorsque la compilation est répétée dans des conditions différentes. Par exemple, il convient que les entrées/sorties du HPD soient explicitement assignées à des broches connues.

NOTE 3 Cette disposition ne s'applique pas à la conception des composants des bibliothèques, qui sont prévus pour être instanciés à différents emplacements de conceptions futures avec des assignations différentes des entrées/sorties.

NOTE 4 Pour concevoir du code HDL portable sur différentes technologies, il est nécessaire que l'assignation des broches soit définie dans un fichier de contraintes et non dans le code HDL. Des éléments du langage tels que les « templates » en VHDL-2008 peuvent y aider.

8.3.5 Conception synchrone ou asynchrone

La conception synchrone consiste à modifier l'état des registres internes et des sorties simultanément et seulement à des moments définis par une horloge. Cela favorise une conception modulaire et compréhensible, tout en minimisant le risque de comportements erronés dus à des aléas temporels (« glitch »); cela favorise également la meilleure utilisation des outils de synthèse et de vérification.

8.3.5.1 Afin de favoriser des conceptions stables, robustes et clairement structurées:

a) il convient d'utiliser une architecture strictement synchrone;

b) les écarts doivent être justifiés.

8.3.5.2 La conception doit garantir la synchronisation des signaux aux interfaces asynchrones.

8.3.5.3 Si une architecture asynchrone est utilisée, une analyse documentée de tous les chemins doit démontrer que les sorties respectent la spécification des exigences (voir Article 6) et qu'il n'existe ni aléas temporels ni métastabilité néfastes.

8.3.5.4 Le comportement du HPD ne doit pas dépendre des valeurs réelles des temps de propagation internes à travers les portes et les interconnexions.

8.3.6 Gestion de l'alimentation

8.3.6.1 Les caractéristiques électriques et temporelles internes du circuit intégré vierge lors de la mise sous tension, du démarrage, de la mise hors tension et de la perte soudaine d'alimentation doivent être connues et prises en compte dans la conception.

8.3.6.2 Le comportement de chaque broche (par exemple le type entrée ou sortie, l'impédance) lors de la mise sous tension, du démarrage, de la mise hors tension et de la perte soudaine d'alimentation doit être documenté.

8.3.6.3 L'utilisation de HPD basés sur des technologies programmables ne doit pas reposer sur l'hypothèse qu'ils suivent leur comportement programmé (vis-à-vis par exemple des fonctions, de la direction et de l'impédance de chaque broche) lors de la mise sous tension, du démarrage, de la mise hors tension et de la perte soudaine d'alimentation, même dans le cas de circuits programmables une seule fois.

8.3.6.4 Il convient que la connexion des broches d'entrée à une source de tension ou à la terre suive les notes d'application du fournisseur, afin d'éviter les pics de courant

potentiels lors de la mise sous tension, du démarrage, de la mise hors tension et de la perte soudaine d'alimentation.

8.3.6.5 Si la distribution des alimentations n'est pas prédéfinie par le fournisseur du composant, un soin particulier doit être apporté à sa conception pour éviter les fautes non déterministes dues à des problèmes tels que les transitoires de tension résultant de pics de courant au moment des fronts d'horloge.

8.3.7 Initialisation

8.3.7.1 La conception doit inclure un signal d'entrée plaçant toutes les sorties, tous les registres et toutes les machines à états finis dans un état connu et documenté.

8.3.7.2 Ce signal d'entrée, qui n'est pas toujours de nature purement numérique, doit être conforme aux exigences du circuit intégré vierge telles que les temps de montée et de descente ou la monotonie des transitions.

8.3.7.3 L'activation de ce signal doit produire l'effet prévu, même si aucune activité d'horloge n'existe.

8.3.7.4 La désactivation de ce signal doit être réalisée de manière à ce que toutes les sorties, tous les registres et toutes les machines à états finis soient maintenus dans leur état initial connu jusqu'à ce que l'activité d'horloge soit établie.

8.3.8 Configurations non fonctionnelles

8.3.8.1 Les broches et registres spécifiques qui font passer le HPD dans des configurations particulières (telles que le test, le diagnostic, le débogage ou la programmation) et qui ne sont pas spécifiées dans la Spécification des Exigences du HPD doivent être analysés et configurés afin d'exclure toute influence néfaste sur ses fonctions.

8.3.8.2 Les concepteurs doivent maîtriser la documentation du fournisseur du circuit intégré afin de connaître les caractéristiques données par les outils aux broches non utilisées (entrée, sortie, haute impédance, etc.).

8.3.8.3 La gestion des broches et registres de configuration du HPD doit être documentée.

8.3.9 Testabilité

8.3.9.1 Chaque fonction réalisée par le HPD doit être testable (détection des défaillances), par des moyens comme les autotests, les tests périodiques ou la contribution observable à une fonction de niveau supérieur elle-même soumise à des autotests ou à des tests périodiques.

8.3.9.2 Si des dispositifs d'autotest sont employés, leur capacité à assurer leur fonction doit être surveillée.

8.3.9.3 La couverture effective de la détection des défauts (voir 6.4.9) et des tests périodiques doit être déterminée et respecter la spécification des exigences du HPD.

8.3.9.4 Les conséquences des défauts doivent être minimisées, par exemple en détectant l'atteinte d'états normalement inatteignables et en menant une action prédéfinie dans ce cas.

8.3.10 Documentation de conception

8.3.10.1 La fin de la phase de conception doit être marquée par la production de la documentation correspondante.

8.3.10.2 Cette documentation doit décrire et justifier la pertinence des décisions de conception relativement au respect de la spécification des exigences du HPD.

8.3.10.3 La documentation de conception doit être suffisamment complète pour que la réalisation puisse se dérouler sans autre éclaircissement.

8.3.10.4 La documentation doit décrire les décisions de conception telles que:

a) l'organisation en modules, ainsi que leurs interfaces et leurs relations,

b) les flux de contrôle et les chemins des données,

c) les protocoles et les algorithmes,

d) les types, les formats et les conventions logiques des signaux,

e) la numérotation des bus, la carte de la mémoire (« mapping »),

f) les définitions, le codage et les initialisations des machines à états finis,

g) les valeurs d'initialisation de tous les registres,

h) les circuits de test.

8.3.10.5 La documentation de conception doit définir la variante effectivement utilisée pour chaque instanciation d'un composant de bibliothèque, pour éviter les ambigüités dues à l'existence de variantes ayant des caractéristiques électriques ou temporelles différentes.

8.3.10.6 La documentation de conception doit inclure tous les paramètres nécessaires pour configurer et utiliser sans ambiguïté tous les blocs natifs et PDB.

8.3.10.7 La documentation de conception doit inclure les caractéristiques électriques et temporelles estimées.

8.4 Réalisation

8.4.1 Considérations générales

A partir de la description RTL, la réalisation synthétise la description logique au niveau porte (liste d'interconnexions, « netlist ») du HPD. Le placement et le routage sont ensuite effectués et aboutissent à la description physique nécessaire pour produire le HPD, par exemple le fichier de programmation ou le « bit stream ».

8.4.2 Produits

8.4.2.1 La réalisation doit apporter toutes les informations nécessaires pour produire de façon systématique le HPD et pour vérifier que chaque exemplaire produit est conforme à la conception.

8.4.2.2 La réalisation doit produire les informations temporelles permettant de compléter la description RTL (« rétro-annotations ») afin de simuler précisément le comportement temporel en tenant compte de tous les retards associés aux portes et aux lignes d'interconnexion.

8.4.2.3 La description rétro-annotée doit être utilisable sur le banc de test (voir 9.5) et, le cas échéant, dans des outils de niveau supérieur tels qu'une simulation au niveau carte.

8.4.3 Fichiers de paramètres et de contraintes

Le concepteur dirige les opérations de synthèse, de placement et de routage avec des paramètres et des directives qui spécifient des contraintes telles que la fréquence de fonctionnement nécessaire, les relations temporelles entre signaux ou la sortance. Pour respecter ces contraintes (transmises aux outils dans des « fichiers de contraintes »), les outils peuvent modifier le placement pour favoriser un chemin de propagation donné aux dépens d'autres, dupliquer une porte pour réduire la charge sur chaque copie et augmenter ainsi leur vitesse, etc.

Des erreurs ou omissions dans les fichiers de paramètres et de contraintes peuvent entraîner des défauts non déterministes subtils, souvent indétectables lors de la simulation et sensibles aux variations normales du procédé de fabrication microélectronique.

8.4.3.1 Les fichiers de paramètres et de contraintes doivent être élaborés selon un processus permettant l'audit.

8.4.3.2 L'exhaustivité et l'exactitude des fichiers de paramètres et de contraintes doivent être vérifiées par l'équipe de vérification (voir Article 9).

8.4.3.3 Les fichiers de paramètres et de contraintes doivent être documentés et soumis à la gestion de configuration.

8.4.4 Analyses post-routage

8.4.4.1 Une analyse post-routage doit démontrer la conformité de la conception et de la réalisation aux règles technologiques définies par les fournisseurs des outils et de la technologie micro-électronique.

8.4.4.2 Des analyses ou simulations post-routage (tenant compte des informations temporelles, ou rétro-annotations) doivent confirmer l'équivalence cycle par cycle de la description post-routage avec la description RTL, pour les cas les plus rapides et les plus lents, y compris pour les initialisations, par exemple en utilisant les deux étapes suivantes:

a) démontrer que la description post-synthèse est équivalente cycle par cycle avec la description RTL,

b) démontrer que la description post-routage est conforme aux contraintes temporelles.

8.4.4.3 Les simulations post-routage peuvent utiliser un sous-ensemble des cas du banc de test utilisé pour les simulations RTL (voir 9.5). Il doit être justifié que ce sous-ensemble couvre les besoins de la démonstration d'équivalence. Une méthode alternative ou complémentaire à la simulation post-routage consiste à utiliser un outil vérifiant que les descriptions RTL et physique sont mathématiquement équivalentes. Si cette approche est adoptée, la qualité et l'adéquation de l'outil utilisé pour effectuer cette vérification doivent être évaluées avant son utilisation (voir Article 15).

8.4.4.4 Les aspects temporels post-routage doivent être analysés.

8.4.4.5 La couverture de chaque fonction par l'autosurveillance doit être analysée en fonction compte de l'objectif requis (voir 6.4.9) en tenant compte des effets que les outils peuvent avoir sur la topologie réelle.

8.4.4.6 Ces analyses doivent être suffisamment détaillées et documentées pour permettre une évaluation technique ultérieure par des personnes non impliquées dans la conception et dans la réalisation.

8.4.4.7 Certaines de ces analyses peuvent être effectuées, sur option ou automatiquement, par les outils. Dans ce cas il n'est pas exigé de les effectuer à nouveau mais:

a) il doit être démontré que les analyses effectuées par les outils ont une couverture et une correction appropriées;

b) les rapports d'analyse (incluant les réglages et les résultats) fournis par les ouils doivent être inclus dans la documentation.

8.4.4.8 Si les analyses découvrent des écarts jugés acceptables:

a) cette acceptation doit être justifiée and documentée;

b) tous les documents impactés doivent être modifiés en conséquence;

c) le plan qualité doit garantir que tout impact sur d'autres systèmes ou composants est documenté et convenablement pris en compte par les personnes responsables des systèmes ou composants impactés.

8.4.5 Redondances introduites ou supprimées par les outils

8.4.5.1 Les réplications de portes effectuées par les outils pour satisfaire les contraintes temporelles ou technologiques doivent être analysées.

8.4.5.2 Il doit être démontré que les états supplémentaires introduits par ces réplications sont acceptables eu égard aux exigences fonctionnelles et aux exigences de sûreté. Il est reconnu que la réplication de portes est effectuée par de nombreux outils de synthèse, mais habituellement, cela peut être adéquatement contrôlé par l'outil lui-même. Cependant, la prudence est nécessaire car la réplication de portes peut engendrer des problèmes si la même vérification formelle d'équivalence est utilisée pour démontrer la correction du niveau RTL et de la réalisation au niveau porte.

8.4.5.3 Comme la réplication introduit de nouveaux états, ceux-ci doivent être analysés pour démontrer que le comportement sûr de la conception ne peut pas être affecté.

8.4.5.4 D'autre part, il doit être démontré que l'optimisation logique effectuée par les outils n'a pas supprimé les mécanismes de détection de faute et de tolérance aux fautes tels que les redondances ou le traitement de cas normalement inatteignables.

8.4.6 Machines à états finis

8.4.6.1 La robustesse de la réalisation finale des machines à états finis doit être analysée.

8.4.6.2 En particulier, les machines à états finis ne doivent pas comporter d'états morts autres que ceux éventuellement précisés dans la spécification des exigences du HPD.

NOTE Un état mort est un état à partir duquel la machine à états finis ne peut atteindre aucun autre état.

8.4.6.3 Les états supplémentaires potentiels introduits par certaines méthodes de codage (telles que le codage « one-hot ») doivent être pris en compte dans l'analyse de défaillances.

NOTE Le codage « one-hot » utilise une bascule par état à représenter; chaque état est représenté par une bascule particulière à « vrai » et les autres à « faux ». Ainsi, seules les combinaisons avec exactement une bascule à « vrai » sont valides. En cas de défaillance, plusieurs bascules peuvent être simultanément à « vrai », ce qui correspondrait à des états supplémentaires, non définis.

8.4.7 Analyse temporelle statique

8.4.7.1 Une analyse temporelle statique (STA) doit être réalisée et documentée pour le meilleur et le pire cas afin de calculer les marges, en tenant compte des informations temporelles fournies par les bibliothèques technologiques et tous les outils de conception et de réalisation concernés.

8.4.7.2 Si des chemins sont exclus de la STA (car considérés comme des « faux chemins ») ou déclarés comme des chemins multi-cycles, cette décision doit être justifiée et documentée.

8.4.7.3 La STA doit démontrer que la fréquence de chaque bloc muni d'une horloge est compatible avec tous les chemins non exclus (voir 8.4.7.2) avec une marge suffisante, pour toute la variabilité spécifiée du procédé de fabrication micro-électronique.

8.4.7.4 L'effet du décalage d'horloge résiduel (« clock skew ») sur les structures critiques telles que les registres à décalage doit être analysé et documenté.

NOTE Le décalage d'horloge résiduel (« clock skew ») est le délai séparant l'arrivée du signal d'horloge à différents emplacements du circuit intégré.

8.4.8 Documentation de réalisation

La fin de la phase de réalisation doit être marquée par la production de la documentation correspondante, incluant:

a) la description au niveau porte du contenu du HPD, utilisable dans le même banc de test que celui utilisé au niveau RTL,

b) la description technologique spécifique (par exemple « fichier de programmation ») nécessaire à la programmation du HPD et au test de chaque exemplaire produit (voir 13.2),

c) les rétros-annotations tenant compte de tous les retards associés aux portes et aux lignes d'interconnexion,

d) les caractéristiques temporelles (telles que la fréquence, les temps d'établissement et de maintien, les temps de montée et de descente, les temps de propagation) et électriques (telles que les niveaux de tension, les courants d'entrée, les sortances, les impédances, la consommation électrique) prévues par les outils sauf s'ils sont déjà définis dans la fiche de spécifications du circuit intégré vierge.

8.4.8.1 La documentation de réalisation doit:

a) donner accès (par inclusion ou renvoi) à la réalisation de chaque bloc, sous-bloc ou module,

b) décrire les choix effectués, en particulier en matière de testabilité, de distribution des horloges et des alimentations électriques, de réinitialisation et d'implémentation des chemins critiques.

8.4.8.2 La documentation de réalisation doit décrire et justifier:

a) les contraintes et les paramètres fournis aux outils,

b) l'analyse effectuée pour garantir la conformité du HPD avec sa spécification des exigences, et le cas échéant les différences constatées,

c) le cas échéant les itérations réalisées sur la conception et la réalisation,

d) le cas échéant les redondances ajoutées ou supprimées lors de la réalisation.

8.4.8.3 La documentation doit être suffisamment détaillée pour permettre à un ingénieur non impliqué dans le projet de faire fonctionner les outils de synthèse, de placement et de routage et d'obtenir les mêmes résultats (HPD et produits de la vérification), mais aussi de vérifier l'exhaustivité et l'exactitude des analyses post-routage.

8.4.8.4 La documentation doit décrire les tests à effectuer périodiquement en cours de fonctionnement, avec une attention particulière sur les modifications de structure introduites par les outils.

8.4.8.5 Si l'engagement du fournisseur du circuit intégré sur la conception ou la réalisation est nécessaire avant la production, cet engagement doit être inclus dans la documentation.

8.5 Outils de niveau système et génération automatique de code

Les exigences des différents composants d'un système peuvent être capturées en utilisant des outils ESL qui fournissent une description textuelle ou graphique.

Ce paragraphe fournit des recommandations supplémentaires applicables quand une description ESL est employée de façon automatisée pour générer en totalité ou en partie la conception du HPD. Cette approche est parfois appelée « synthèse de haut niveau ».

8.5.1 Si une spécification des exigences écrite dans un langage ESL est utilisée pour générer automatiquement une partie ou la totalité de la description RTL du HPD:

a) il convient que la description générée soit directe et évite toute complexité inutile;

b) il convient que la description permette aux concepteurs de matériel de comprendre aisément le comportement du circuit, de façon à identifier rapidement les erreurs et ambigüités.

8.5.2 Il convient que le langage ESL et les outils associés, en particulier ceux utilisés pour la génération de code et les analyses, respectent les exigences de 8.2.

8.5.3 Si 8.5.2 n'est pas respecté:

a) la description ESL du HPD doit être traduite en un description HDL conforme aux exigences de 8.2, qui constituera la base des activités ultérieures de conception, réalisation vérification,

b) ces activités ultérieures doivent respecter les exigences de la présente norme.

8.5.4 Tout écart de conformité des descriptions générées (par exemple RTL, synthétisée, routée) par rapport aux exigences de conception et de réalisation (voir 8.3 et 8.4) doit être identifié et justifié.

8.5.5 Si certaines des analyses, vérifications et revues définies par la présente norme aux Articles 8, 9 et 10 ne sont pas effectuées, il doit être formellement démontré que les produits qui n'ont pas été analysés, vérifiés ou revus sont nécessairement corrects.

8.5.6 Les produits générés ne doivent pas être modifiés par une action manuelle directe sur ceux-ci.

8.5.7 Les produits doivent être générés à nouveau si quelque chose a besoin d'être modifié, par exemple suite aux résultats des activités de vérification ou de revue.

8.6 Documentation

Ce paragraphe précise les exigences générales en matière de documentation pour la conception et la réalisation du contenu du HPD. Il complète les exigences spécifiques des activités particulières traitées en 8.1 à 8.5.

8.6.1 La fin des phases de conception et de réalisation doit être marquée par la production de la spécification de conception du HPD.

Ce document sert de base à la revue formelle de conception et de réalisation ainsi qu'à la production qui en découle.

8.6.2 Suffisamment de détails doivent être donnés pour que la production puisse se dérouler sans autre éclaircissement.

8.6.3 Il convient de structurer le document en fonction des phases du processus de développement. La spécification de conception peut se présenter comme un document unique ou un ensemble intégré de documents.

8.6.4 Si un ensemble intégré de documents est utilisé, chaque document doit avoir une relation définie avec les autres documents et doit aborder un domaine bien délimité.

8.6.5 Il convient de choisir le format des documents en fonction de sujet spécifique, y compris:

a) les descriptions narratives;

b) les expressions arithmétiques et logiques;

c) les représentations graphiques, diagrammes et dessins.

8.7 Revue de conception et de réalisation

8.7.1 Les phases de conception et de réalisation doivent se terminer par une revue formelle.

8.7.2 La revue de conception et de réalisation doit examiner la documentation en couvrant la conception, la réalisation, les analyses et les vérifications.

8.7.3 La revue doit examiner l'exhaustivité et l'exactitude des fichiers de paramètres et de contraintes transmis aux outils de conception et de réalisation.

8.7.4 La revue doit examiner l'exhaustivité et l'exactitude de l'analyse temporelle statique (STA) et des analyses post-routage, pour vérifier l'exactitude et la robustesse de la conception et de la réalisation, en tenant compte des effets néfastes potentiels induits par les modifications effectuées par les outils (telles que la simplification logique ou la duplication de portes).

8.7.5 L'équipe de revue doit inclure des experts du matériel et des ingénieurs des équipes responsables du système ou des composants qui utilisent le HPD ou qui sont interfacés avec lui (tels que la carte électronique ou le logiciel).

9 Vérification du HPD

9.1 Considérations générales

Les activités de vérification menées dans le cadre du développement du HPD relèvent généralement de la responsabilité du fabricant d'I&C, et sont assurées par une équipe indépendante de celles effectuant la conception et la réalisation du HPD. Le meilleur moyen est de constituer une équipe de vérification.

Des activités de vérification supplémentaires peuvent être entreprises par une tierce partie au titre de l'évaluation du HPD et de son processus de développement, afin de donner l'assurance qu'il répond à ses objectifs. Il y a de nombreuses façons d'organiser et réaliser ce rôle de vérification indépendante, celle-ci relevant souvent de la réglementation nationale.

9.1.1 L'équipe de vérification doit être composée de personnes non engagées dans le développement et possédant les compétences et les connaissances nécessaires. Les exigences suivantes définissent explicitement le niveau d'indépendance exigé.

9.1.2 La direction de l'équipe de vérification doit être séparée et indépendante de celle de l'équipe de développement.

9.1.3 La communication entre l'équipe de vérification et l'équipe de développement, que ce soit pour demande de clarification ou pour constat d'erreur, doit se faire de manière formelle par écrit à un niveau de précision qui puisse être audité.

9.1.4 Il convient que les interactions entre les deux parties visent à maintenir l'indépendance du jugement de l'équipe de vérification.

9.1.5 L'équipe de vérification doit avoir des responsabilités et des obligations clairement définies.

9.1.6 La production de chaque phase de développement (Figure 2) doit être vérifiée.

9.1.7 Les activités de vérification doivent confirmer l'adéquation de la spécification des exigences du HPD pour satisfaire aux exigences du système ou sous-système assignées au HPD par la spécification du système ou du sous-système.

9.1.8 Les activités de vérification doivent confirmer l'adéquation de la sélection et des règles d'utilisation de chaque circuit intégré vierge, technologie micro-électronique, bloc natif et PDB pour satisfaire aux exigences qui lui sont assignées par la spécification des exigences du composant (voir Article 7).

9.1.9 Les activités de vérification doivent confirmer l'adéquation de la spécification de conception du HPD pour satisfaire à la spécification des exigences du HPD.

9.1.10 Les activités de vérification doivent confirmer la conformité du HPD à la spécification de conception du HPD (voir Article 8).

NOTE La vérification post-réalisation a une importance majeure dans la détection des effets potentiellement néfastes des simplifications logiques et des duplications de portes que les outils peuvent effectuer, ainsi que des fautes potentielles dues aux outils ou à leur utilisation.

9.1.11 Il convient que chaque activité de production débute sur la base de données/documents vérifiés.

9.1.12 Il convient que la vérification du produit d'une phase soit réalisée avant le début de la phase suivante. Sinon, cette vérification doit être réalisée avant la vérification de la phase suivante.

Un travail préparatoire pour une phase à venir peut être réalisé avant que la phase précédente ne soit vérifiée.

9.1.13 Si les documents/données d'entrée pour une activité ont été modifiés, cette activité et les suivantes doivent être reprises si besoin pour traiter l'impact potentiel.

9.2 Plan de vérification

9.2.1 Le plan de vérification doit être établi avant de débuter les activités de vérification du HPD.

9.2.2 Ce plan doit expliciter tous les critères, techniques et outils à utiliser dans le processus de vérification.

9.2.3 Il doit décrire les activités à réaliser pour évaluer chaque élément du HPD, chaque outil impliqué dans le processus de développement et chaque phase pour vérifier si la spécification des exigences du HPD est respectée.

9.2.4 Le niveau de détail doit être tel qu'une équipe de vérification puisse exécuter le plan de vérification et aboutir à un jugement objectif sur le fait que le HPD respecte ou non sa spécification des exigences.

9.2.5 Le plan de vérification doit être préparé par une équipe de vérification traitant:

a) la sélection et la justification de stratégies de vérification en fonction de la nature des exigences, des caractéristiques de conception et de réalisation, et de la technologie micro-électronique;

b) la sélection et l'utilisation des outils de vérification;

c) l'exécution de la vérification;

d) la documentation des activités de vérification;

e) l'évaluation des résultats de la vérification obtenus directement à partir des outils de vérification et des tests, l'évaluation du respect ou non des exigences de sûreté.

9.2.6 Le plan de vérification doit documenter chaque test, y compris son objectif, ses résultats attendus, et les critères permettant de décider si le résultat est correct ou non.

9.2.7 Il convient que les tests conçus en fonction d'aspects fonctionnels se traduisent par une sollicitation extensive du HPD.

9.2.8 Le plan de vérification doit identifier toutes les preuves objectives nécessaires pour confirmer l'étendue des tests. À cet effet, les critères de couverture de test choisis en fonction de la conception et de la réalisation doivent être justifiés et documentés.

9.2.9 Des dispositions adéquates doivent être prévues pour le traitement et la résolution de tout problème de sûreté soulevé pendant les activités de vérification réalisées

lors du développement par le fabricant d'I&C ou lors d'une évaluation par une tierce partie.

9.2.10 Tout problème de sûreté doit être résolu par des modifications correctives ou des dispositions de compensation appropriées.

9.3 Vérification de l'utilisation des éléments prédéveloppés

La bonne configuration et la bonne utilisation des éléments prédéveloppés tels que circuits intégrés vierges, blocs natifs et PDB, ainsi que leur compatibilité, doivent être vérifiées au regard des règles spécifiées par leurs fournisseurs et de celles élaborées pendant les activités de l'Article 7.

9.4 Vérification de la conception et de la réalisation

9.4.1 La vérification doit inclure des tests et des analyses pour traiter:

a) l'adéquation de la spécification de conception par rapport à la spécification des exigences du HPD en matière de cohérence et d'exhaustivité, jusque et y compris au plus bas niveau de bloc et de module;

b) la décomposition de la conception en une hiérarchie de blocs et de modules, et la façon dont ils sont spécifiés du point de vue de:

   1) la testabilité en vue de la vérification;

   2) la compréhensibilité par les équipes de développement et de vérification;

   3) l'aptitude à modification ultérieure;

c) la réalisation correcte des exigences de sûreté.

9.4.2 Le résultat de la vérification doit être documenté.

9.4.3 La documentation doit inclure les conclusions et identifier clairement les points nécessitant une action, tels que:

a) les éléments non conformes aux exigences;

b) les éléments non conformes aux règles de conception et de réalisation;

c) les modules, données, structures et algorithmes mal adaptés au problème.

9.5 Bancs de test

9.5.1 Un programme de simulation et de test (« banc de test ») doit être développé et documenté. Au besoin, ce banc de test peut se composer de plusieurs parties ayant des portées et des objectifs différents, par exemple plusieurs bancs de test peuvent être dédiés au test des modules et un ou plusieurs au test d'ensemble.

9.5.2 Le banc de test (la structure) peut être développé par l'équipe de conception pour ses propres besoins en matière de tests et utilisé par l'équipe de vérification. Cependant, les cas de test (entrées et sorties attendues) exigés par la présente norme doivent être développés par l'équipe de vérification, afin de réduire le risque de masquage d'erreur et de fournir une confirmation supplémentaire de la compréhensibilité et de l'exhaustivité de la documentation de conception.

9.5.3 Le banc de test doit:

a) exercer chaque module dans son environnement simulé avec tous les détails logiques nécessaires,

b) avoir une résolution temporelle suffisante lorsqu'il est utilisé après la réalisation pour les aspects temporels.

9.5.4 Le banc de test doit inclure les cas de test exerçant tous les éléments mentionnés dans la spécification des exigences du HPD et dans la spécification de conception, tels que les fonctions, modes, machines à états finis, algorithmes, protocoles.

9.5.5 Il convient que le banc de test inclue toutes les séquences et tous les temps d'entrée nécessaires, et mémorise toutes les séquences et tous les temps de sortie produits pendant l'exécution, pour rendre l'exécution du test entièrement automatique.

9.5.6 Il convient que le banc de test inclue les séquences et les temps de sortie attendus, ainsi qu'une comparaison automatisée avec ceux réellement produits pendant l'exécution du test (relativement aux critères appropriés, voir 9.2), de manière à fournir un résultat « succès/échec » global en plus des résultats détaillés du test.

9.5.7 Si des saisies, observations ou comparaisons manuelles sont requises:

a) les valeurs et les activités impliquées doivent être documentées avec suffisamment de détails pour permettre à une personne non impliquée dans le projet de refaire le test. Cela peut nécessiter une définition des étapes cycle par cycle et des valeurs au niveau bits,

b) une justification documentée doit être fournie parce que les saisies, observations ou comparaisons manuelles sont potentiellement sujettes à erreur.

9.5.8 Le banc de test a pour mission de faire précisément état de toutes les défaillances et de ne pas faire à tort état de comportements corrects. Il doit par conséquent être réalisé en respectant 10.4.6 and 15.2.

9.6 Couverture des tests

9.6.1 Des critères de couverture de test doivent être sélectionnés et documentés.

9.6.2 Une analyse documentée des critères de couverture de test doit démontrer qu'ils sont suffisants, en tenant compte de la spécification des exigences du HPD et des caractéristiques de conception/réalisation, et que le banc de test offre une observabilité suffisante pour produire une décision succès/échec pour chaque élément couvert.

9.6.3 Ces critères peuvent être liés, par exemple, aux instructions, décisions, expressions, chemins, machines à états finis ou processus. Si un objectif de critère de couverture ne peut être atteint, par exemple en raison de la structure RTL (une couverture de 100 % des chemins est particulièrement difficile à atteindre), une justification documentée doit être produite.

NOTE Un chemin est une suite particulière de branches suivie lors d'une exécution du code.

9.6.4 Chaque module développé au sein du projet doit être spécifiquement testé.

9.7 Exécution des tests

9.7.1 Des tests doivent être effectués au moyen des bancs de test après la phase de conception, sur la description RTL, afin de confirmer son exactitude.

9.7.2 Des tests doivent être effectués après la phase de réalisation pour confirmer que la description post-routage respecte les contraintes temporelles, en tenant compte des informations temporelles fournies par les outils et les bibliothèques (rétro-annotations).

9.7.3 Les tests (utilisant la simulation) doivent être effectués pour le « pire cas » (temps de propagation maximal) et pour le « meilleur cas » (temps de propagation minimal).

9.7.4 Les résultats des tests (valeurs, séquences et caractéristiques temporelles) doivent être documentés.

9.7.5 Une analyse documentée de chaque écart doit décider s'il est acceptable ou non.

9.8 Vérification statique

9.8.1 Il convient d'effectuer les activités de vérification suivantes:

a) vérification du typage et de la syntaxe,

b) vérification des paramètres d'appel ou d'instanciation des modules, fonctions, procédures, blocs natifs et PDB,

c) vérification des débordements,

d) exhaustivité de la liste de sensibilité des processus (voir note),

e) l'exhaustivité des cas explicitement programmés dans les instructions et les constructions à choix multiples,

f) détection des états morts dans les machines à états finis,

g) détection des effets de bord dans les fonctions ou macros, détection des objets partagés,

h) vérification des règles de conception (DRC) logiques et physiques, qui analysent la liste d'interconnexions (netlist) et les autres fichiers générés pour y rechercher des erreurs physiques et logiques.

NOTE La liste de sensibilité (« sensitivity list ») est un élément du langage VHDL.

9.8.2 Des méthodes de vérification statique telles que la STA (voir 8.4.7) peuvent être utilisées pour certains aspects de la vérification si leurs principes sont mathématiquement fondés. Dans ce cas, les outils employés pour mettre en œuvre ces méthodes doivent:

a) avoir une maturité et une normalisation similaires à celles exigées par la présente norme pour les outils de simulation,

b) être conformes aux exigences de l'Article 15 applicables aux outils de vérification.

10 Aspects de l'intégration du système liés au HPD

10.1 Considérations générales

Le processus d'intégration du système est la combinaison des composants matériels (et logiciels le cas échéant) vérifiés dans des sous-systèmes et finalement dans le système complet. Ce processus consiste en deux types d'activités:

a) intégration du système: assemblage et interconnexion des composants matériels (et des composants logiciels le cas échéant) vérifiés tels que définis dans les documents de conception afin de construire les cibles intermédiaires et finale. La séquence d'assemblage ainsi que le degré d'intégration des cibles successives dépendent des caractéristiques du projet;

b) vérification du système intégré: vérifier que les composants sont conformes à leurs spécifications de conception, sont capables de fonctionner ensemble, et respectent leurs exigences d'interfaçage.

Cet article précise les exigences relatives à l'intégration du système, en complément de l'article 6.2.5 de la CEI 61513, quand des HPD sont impliqués.

10.2 Aspects du plan d'intégration du système liés au HPD

Ce paragraphe développe les exigences de la CEI 61513, 6.3.4, qui doivent être appliquées.

10.2.1 Ce plan doit être préparé et documenté lors de la phase de conception et de réalisation, et vérifié par rapport aux exigences des systèmes de classe 1.

10.2.2 Ce plan doit être préparé suffisamment tôt dans le processus de développement pour assurer que toutes les exigences d'intégration seront prises en compte dans la conception du HPD, du système et de ses composants.

10.2.3 Ce plan doit préciser les normes et les procédures à suivre dans la phase d'intégration du système.

10.2.4 Ce plan doit documenter les dispositions du plan d'assurance qualité du système qui sont applicables à l'intégration du système.

10.2.5 Le plan d'intégration doit spécifier:

a) les séquences et les caractéristiques temporelles des signaux d'entrée du système ou du sous-système testé,

b) les séquences et les caractéristiques temporelles des signaux attendus en sortie du système ou du sous-système testé,

c) les critères d'acceptation.

10.2.6 Le plan d'intégration du système doit tenir compte des exigences assignées au HPD par la conception du système, du matériel et le cas échéant du logiciel. Le plan doit également inclure les exigences relatives aux procédures et aux méthodes de contrôle incluant:

a) la gestion de configuration du système (voir 5.5);

b) l'intégration du système;

c) la vérification du système intégré;

d) la résolution des défauts.

10.2.7 Le plan d'intégration du système doit définir les deux aspects (identification et contrôle) de la gestion de configuration selon les exigences de l'article 6.3.2.3 de la CEI 61513.

10.2.8 Lors du processus de vérification des interactions du HPD avec les autres composants du système, certains aspects peuvent être vérifiés au niveau des sous-systèmes (unités de calcul) ou au niveau du système complet si c'est plus pratique. Si une vérification par des tests n'est pas possible à ces niveaux:

a) toutes les exigences du HPD doivent être vérifiées par d'autres moyens (par ex. test en boîte claire),

b) la stratégie de vérification correspondante doit être documentée dans le plan d'intégration.

10.2.9 Toutes les interdépendances entre la vérification du HPD et celle du système intégré doivent être documentées dans le plan d'intégration du système.

10.3 Aspects spécifiques de l'intégration du système

Les procédures spécifiques à l'intégration du système dépendent des caractéristiques de l'architecture de ce système.

10.3.1 Ces procédures doivent être établies et référencées dans le plan d'intégration du système pour couvrir les activités suivantes:

a) l'approvisionnement des composants corrects conformément au plan de gestion de configuration du système (6.3.2.3 de la CEI 61513) et aux procédures de production (voir Article 13);

b) l'intégration du HPD dans le système (par exemple mise en place du composant, configuration, câblage des interconnexions);

c) les tests préliminaires des fonctions du système intégré (voir les exigences ci-dessous);

d) la documentation des produits du processus d'intégration et de la configuration du système soumise aux tests;

e) la livraison formelle du système intégré pour les tests de validation.

10.3.2 Si la résolution d'un défaut nécessite une modification du HPD vérifié ou de la spécification de conception, ce défaut doit être signalé conformément aux procédures établies par 10.5.

10.3.3 Tout défaut détecté lors de l'intégration du système, résultant exclusivement d'erreurs du processus d'intégration lui-même et n'affectant aucun document du HPD, doit être corrigé par mise à jour du plan d'intégration du système.

10.4 Vérification du système intégré

La vérification du système intégré détermine si les composants et sous-systèmes vérifiés ont été correctement intégrés dans le système, s'ils sont compatibles et s'ils fonctionnent comme spécifié.

10.4.1 Le système doit être aussi complet que possible pour cette vérification.

10.4.2 Les cas de test sélectionnés pour la vérification du système doivent:

a) solliciter toutes les interfaces et opérations de base du HPD;

b) solliciter toutes les caractéristiques d'interfaçage du HPD décrites dans la spécification des exigences et dans la spécification de conception, comme les protocoles, les séquences, les caractéristiques temporelles et électriques;

c) avoir une couverture suffisante pour démontrer que le HPD fonctionne comme requis dans toutes les situations atteignables dans le système.

10.4.3 Le plan d'intégration du système doit identifier les tests à effectuer pour chaque exigence d'interfaçage du HPD.

10.4.4 Les tests du système intégré doivent être revus et leurs résultats évalués par une équipe de vérification ayant une bonne connaissance des spécifications du système.

10.4.5 L'équipement utilisé pour vérifier le système doit être étalonné de manière appropriée.

10.4.6 Les outils logiciels de vérification utilisés doivent respecter les exigences de l'Article 15 concernant les outils de vérification.

10.4.7 La vérification du système intégré doit démontrer que tous les composants du système ont les performances attendues (par exemple unités de traitement et dispositifs de communication).

10.5 Procédures de résolution des défauts

10.5.1 Les exigences de la CEI 61513, 6.3.2.4 (Procédures de résolution des défauts) doivent s'appliquer.

10.5.2 Les procédures de résolution des défauts doivent garantir que toute modification requise du HPD respecte les exigences de l'Article 12.

10.6 Aspects du rapport de test du système intégré lié au HPD

10.6.1 Les exigences de la CEI 61513, 6.4.5.2 doivent s'appliquer.

10.6.2 Les résultats des tests doivent être conservés sous une forme telle qu'ils pourront être vérifiés par des personnes non directement impliquées dans le plan de vérification ou dans l'exécution effective des tests.

11 Aspects de la validation du système liés au HPD

11.1 Considérations générales

Les HPD sont typiquement validés durant la phase de validation du système. La validation du système est couverte par la CEI 61513. La présente norme précise des exigences supplémentaires pour valider les performances (fonctionnelles, temporelles et électriques) des HPD.

a) Des tests doivent être effectués pour valider le système et le HPD conformément aux exigences relatives aux systèmes de classe 1.

b) Les tests de validation doivent être effectués sur le système dans la configuration d'assemblage finale, incluant la version finale du HPD.

11.2 Aspects du plan de validation du système liés au HPD

11.2.1 La validation du système doit être conduite en respectant un plan formel de validation du système.

11.2.2 Ce plan doit identifier des cas de test statiques et dynamiques.

11.2.3 Le plan de validation du système doit être développé et le résultat de la validation évalué par des personnes n'ayant participé ni à la conception ni à la réalisation.

11.3 Validation du système

11.3.1 Le système doit être sollicité par des signaux d'entrées statiques et dynamiques simulant le fonctionnement normal, les incidents d'exploitation et les conditions d'accident demandant une action.

11.3.2 Chaque fonction de catégorie A du système doit être sollicitée par des tests confirmant chaque signal de sortie requis, de façon isolée ou combinée.

11.3.3 Les tests doivent:

a) couvrir toutes les fonctions de la spécification des exigences du HPD, dans tous les modes (voir 6.2);

b) couvrir autant que possible toutes les plages des signaux et des paramètres calculés;

c) couvrir de façon exhaustive les votes et les autres logiques simples ou combinées;

d) être effectués pour tous les signaux de déclenchement ou de protection dans la configuration finale d'assemblage;

e) couvrir les réponses requises aux défaillances spécifiées;

f) couvrir toutes les autres fonctions ayant un impact sur la sûreté du réacteur.

11.3.4 De plus, les valeurs des signaux d'entrée, les signaux de sorties attendus et les critères d'acceptation doivent être établis dans le plan de validation du système.

11.3.5 L'équipement utilisé pour la validation doit être étalonné et configuré (paramètres matériels et logiciels) de manière appropriée.

11.4 Aspects du rapport de validation du système liés au HPD

11.4.1 Le rapport de validation du système doit documenter les résultats liés aux HPD inclus dans le système.

11.4.2 Le rapport doit identifier le matériel, éventuellement le logiciel, la configuration du système utilisée, la configuration des outils utilisés et l'équipement de test utilisé (incluant son étalonnage et les modèles de simulation) conformément à la CEI 61513, 6.4.6.2.b.

11.4.3 Ce rapport doit également identifier tous les écarts découverts pendant le test.

11.4.4 Ce rapport doit fournir un résumé des résultats de la validation du système.

11.4.5 Ce rapport doit évaluer la conformité du système à toutes les exigences.

11.4.6 Les résultats doivent être conservés sous une forme telle qu'ils pourront être vérifiés par des personnes non directement impliquées dans la validation.

11.4.7 Les simulations de la centrale et de ses systèmes utilisées pour la validation doivent être documentées.

11.5 Procédures de résolution des défauts

Les exigences de 10.5 doivent également s'appliquer aux aspects de la validation du système liés au HPD.

12 Modification

12.1 Modification des exigences, de la conception ou de la réalisation

12.1.1 Le processus de modification et la documentation associée doivent être conformes aux exigences de la CEI 61513 (6.2.8 et 6.4.7), de la CEI 60987:2007 (Article 12) et de la CEI 60880:2006 (Article 11).

12.1.2 Tous les documents impactés doivent être vérifiés conformément aux exigences de l'Article 9, par des personnes qui ne sont pas engagées dans la réalisation de la modification.

12.2 Modification de la technologie micro-electronique

Le fournisseur peut actualiser la technologie micro-électronique (par exemple, nouvelle version d'un FPGA vierge pour augmenter la vitesse ou réduire la surface de silicium). Même si le nouveau circuit est revendiqué comme « compatible », cela n'implique pas que n'importe quelle conception fonctionnera à l'identique sur les deux circuits.

12.2.1 Le processus d'acceptation (voir Article 7) doit être à nouveau exécuté, suivi si nécessaire des phases du cycle de vie impactées en fonction des différences constatées

12.2.2 Les activités de vérification et de simulation concernées doivent être à nouveau exécutées et dûment documentées pour s'assurer que toutes les exigences fonctionnelles, électriques et temporelles sont respectées.

12.2.3 Même si le nouveau circuit intégré vierge a la même configuration logique que l'ancien et est compatible broche à broche avec lui, le besoin de régénérer les fichiers de programmation (par exemple à cause de variations des durées ou tensions des impulsions de programmation) doit être évalué et documenté.

13 Production du HPD

13.1 Considérations générales

Le domaine d'application de la présente norme exclut la conception et la fabrication des ressources micro-électroniques prédéveloppées (par exemple un FPGA vierge) utilisées en entrée du processus de développement du HPD. Le terme « production » dans la présente norme désigne les étapes finales qui ont pour résultat le circuit intégré prêt à être utilisé dans le système d'I&C.

13.2 Tests de production

13.2.1 Des tests doivent vérifier les fonctions du HPD, ainsi que ses performances temporelles (telles que fréquence, temps de montée et de descente, temps de propagation, etc.) et ses caractéristiques électriques (telles que consommation, capacités, etc.).

13.2.2 Il convient de démontrer que les tests effectués par le fournisseur du circuit intégré répondent aux besoins (voir 13.2.1). Le fabricant d'I&C n'a pas besoin de répéter les tests effectués par le fournisseur du circuit intégré ni de connaître les vecteurs de test correspondants.

13.2.3 Si 13.2.2 n'est pas respecté, des tests supplémentaires (avec une documentation des entrées, sorties attendues et critères d'acceptation) doivent être effectués par le fabricant d'I&C pour répondre aux besoins (voir 13.2.1).

13.2.4 Des tests de production effectués au niveau carte électronique (après assemblage du HPD sur la carte, par exemple par soudure) doivent vérifier que l'interface du

circuit intégré est opérationnelle (par exemple des tests de “broche d’entrée/sortie collée”, de défauts, de fonctionnalité globale).

13.2.5 Tout circuit intégré produit doit passer avec succès les tests de production ou être rejeté.

13.2.6 Les résultats des tests doivent être enregistrés avec les informations d’identification telles que le numéro de lot afin de permettre le diagnostic d’éventuels problèmes de production.

13.3 Fichiers de programmation et activités de programmation

13.3.1 Les fichiers de programmation doivent inclure des codes de détection d’erreur et l’équipement de programmation doit les vérifier.

13.3.2 Pour chaque circuit intégré produit:

a) la configuration après programmation doit être vérifiée et

b) les informations de traçabilité pertinentes (telles que le numéro de lot, le fichier journal de programmation, les caractéristiques des commutateurs programmables avant et après programmation) doivent être conservées.

13.3.3 Toutes les procédures et exigences spécifiées par le fournisseur du circuit intégré doivent être respectées (par ex. pour éviter une décharge électrostatique).

13.3.4 Seuls des outils garantis et privilégiés par le fournisseur du circuit intégré doivent être utilisés.

14 Aspects de l’installation, du démarrage et du fonctionnement liés au HPD

a) Le processus et la documentation d’installation, de démarrage et de fonctionnement doivent respecter les exigences de CEI 61513 (6.2.7 et 6.3.6), de la CEI 60987:2007 (Articles 10 et 13) et de la CEI 60880:2006 (Article 12).

b) Selon la CEI 60671, les systèmes et équipements d’I&C réalisant des fonctions de catégorie A sont testés périodiquement pour démontrer leur bon fonctionnement. Pour que les tests aient la couverture nécessaire relativement au HPD, des techniques appropriées doivent être employées pour améliorer la testabilité, par exemple le « boundary scan ».

15 Outils logiciels pour le développement des HPD

15.1 Considérations générales

L’Article 14 de la CEI 60880:2006 doit s’appliquer aux outils logiciels utilisés pour le développement des HPD, à l’exception de 14.3.4.3, 14.3.4.4 et 14.3.4.5.

NOTE 1 L’évaluation technique faite par le fournisseur de l’outil (non limitée à l’assurance qualité) constitue une méthode acceptable pour respecter l’exigence 14.2.2 de la CEI 60880:2006, à condition que la documentation correspondante soit disponible.

NOTE 2 La propriété « fiabilité » des outils logiciels mentionnée à l’Article 14 de la CEI 60880:2006 signifie ici « crédibilité » ou « correction ».

NOTE 3 L’ISO/IEC 9126 est remplacée par l’ISO/IEC 25000.

NOTE 4 Les bibliothèques intégrées dans un outil peuvent être évaluées dans le contexte de l’évaluation de l’outil.

NOTE 5 La vérification des sorties de l’outil mentionnée en 14.3.2.4 de la CEI 60880:2006 peut être effectuée de différentes façons, par exemple par simulation avec un simulateur diversifié par rapport à celui de l’outillage de synthèse.

15.2 Exigences additionnelles pour les outils de conception, réalisation et simulation

15.2.1 Les outils logiciels doivent donner accès aux paramètres contrôlant la synthèse logique et la réalisation (par exemple au moyen de réglages).

15.2.2 Il convient que les outils logiciels n'ajoutent pas de structure non directement liée à des instructions sources HDL (par exemple duplication de portes pour respecter des exigences temporelles) sans avertissement.

15.2.3 Les concepteurs doivent avoir une connaissance préalable des outils logiciels, ils doivent en particulier savoir comment les outils fonctionnent sur les structures et les constructions utilisées dans le projet.

15.2.4 Si un outil logiciel nécessite des arguments de ligne de commande, ceux-ci doivent être contenus dans un fichier de script (placé sous gestion de configuration) pour éviter les erreurs d'invocation manuelle.

NOTE 1 Ceci est utile non seulement pour la cohérence, mais aussi pour aider à trouver l'origine d'un défaut, qui peut se trouver dans le code source, dans l'outil ou dans les paramètres de l'outil. Cela peut également être nécessaire pour l'évaluation du risque de DCC due aux outils de conception et de réalisation.

15.2.5 Lors du passage à une nouvelle version d'un outil logiciel transformant une information de conception (par exemple synthèse logique, placement ou routage), toutes les activités de simulation, d'analyse et de vérification affectées doivent être à nouveau effectuées.

NOTE 2 Il peut être justifié par une analyse documentée qu'une modification donnée d'un outil ne peut pas affecter les activités mentionnées, par exemple la correction d'un comportement anormal de l'interface graphique de l'outil.

NOTE 3 Les activités terminées avant la modification de l'outil n'ont pas besoin d'être répétées.

16 Segmentation de la conception ou partitionnement

16.1 Bases

Il est possible dans certains HPD de concevoir et réaliser des circuits alloués à des zones physiquement différentes du circuit intégré, ayant peu ou pas d'interconnexions entre eux et n'utilisant pas de ressource matérielle commune. Certains HPD permettent l'existence de telles zones, parfois nommées « lacs », séparées par des espaces non utilisés ou non utilisables. Les avantages de la segmentation de conception ou du partitionnement peuvent inclure la réalisation de fonctions auxiliaires ou support (l'objectif n'est pas de remplacer les trains ou canaux redondants de la conception du système).

16.2 Fonctions auxiliaires ou support

16.2.1 Considérations générales

En général, les fonctions auxiliaires ou support d'un HPD, même si elles n'exécutent pas de fonctions de catégorie A, peuvent interférer avec des fonctions de catégorie A de ce HPD. Par conséquent, à moins qu'il ne soit possible de démontrer le respect des exigences de 16.2.2, les fonctions auxiliaires ou support doivent être développées, réalisées et vérifiées selon les exigences de la présente norme (c'est-à-dire comme des fonctions de catégorie A).

16.2.2 Partitionnement de fonctions auxiliaires ou support de catégorie autre que A

La présente norme reconnaît qu'il peut être possible, au moyen de dispositions spécifiques de conception et de partitionnement du HPD, d'assurer que des fonctions auxiliaires ou support sont indépendantes de celles de catégorie A et ne peuvent interférer de façon néfaste avec elles. Dans ce cas, à condition que les exigences suivantes soient respectées, des fonctions auxiliaires ou support peuvent être réalisées dans un HPD de classe 1 sans la rigueur appliquée aux fonctions de catégorie A:

a) il doit être démontré par conception, réalisation, évaluation et vérification systématique que le fonctionnement ou la défaillance de telles fonctions auxiliaires ou support ne peuvent interférer directement ou indirectement avec aucune fonction de catégorie A, que la cause de la défaillance soit interne ou externe au HPD (par exemple induite par les alimentations électriques, par un court-circuit sur une connexion, etc.),

b) cette démonstration doit traiter toutes les causes possibles d'interférence, c'est-à-dire fonctionnelles, électriques, électromagnétiques, thermiques, etc.,

c) en particulier, les zones du circuit intégré utilisées pour réaliser ces fonctions auxiliaires ou support doivent être physiquement différentes de celles utilisées pour les fonctions de catégorie A,

d) en cas de modification du HPD, il doit être démontré que les exigences de 16.2.2 sont encore respectées,

e) l'interface entre les circuits réalisant des fonctions de catégorie A et des fonctions auxiliaires ou support doit être simple et entièrement vérifiable,

f) les données reçues par les fonctions de catégorie A provenant de fonctions auxiliaires ou support doivent être limitées à des valeurs de paramètres statiques (par exemple constantes de calibration, valeurs de réglage),

g) les fonctions de catégorie A ne doivent avoir aucune dépendance temporelle vis-à-vis de la réception de données provenant de fonctions auxiliaires ou support,

h) des dispositions de sûreté appropriées (par exemple protocoles de communication sûrs) doivent être mis en œuvre pour toute communication entre les fonctions de catégorie A et les fonctions auxiliaires ou support, de façon à détecter toutes les erreurs de transmission et fournir une réponse sûre appropriée, ou à acquitter une réception de données correcte.

17 Défense contre les défaillances de cause commune dues aux HPD

17.1 Bases

Des défauts systématiques peuvent être introduits dans tout processus de conception et de réalisation à cause d'erreurs humaines; ainsi, de tels défauts peuvent être introduits pendant la conception et la réalisation d'un HPD (dans la partie développée ou dans un élément prédéveloppé inclus). Les HPD pourraient donc être affectés par des défauts latents systématiques qui pourraient, sous certaines conditions de déclenchement, conduire à la DCC d'instanciations multiples d'une conception de HPD.

La possibilité de DCC de systèmes multiples est dans le domaine d'application de normes de plus haut niveau du SC 45A, en particulier la CEI 61513 et la CEI 62340. La possibilité de DCC de multiples instanciations d'une conception de HPD dans un même système est traitée par la présente norme. Comme indiqué à l'Article 1, « Domaine d'application et objet », la présente norme définit des exigences et des processus de développement et de vérification qui minimisent la possibilité de défauts systématiques des HPD et donc –comme de tels défauts peuvent causer des DCC- minimisent aussi la possibilité de DCC due aux HPD.

Des exigences supplémentaires pour la protection contre les fautes systématiques pouvant conduire à une DCC due aux HPD sont précisées dans le paragraphe suivant.

17.2 Exigences

17.2.1 Les aspects du processus de développement du HPD qui peuvent conduire à la DCC d'instanciations multiples de la conception du HPD (et ne sont pas déjà traités par des articles de la présente norme) doivent respecter lorsque c'est applicable les exigences pertinentes de la CEI 60880:2006, 13.1 (en remplaçant « logiciel » par « HPD »).

NOTE 1 Ces aspects sont typiquement liés au développement de programmes HDL.

17.2.2 Une analyse selon les exigences pertinentes de la CEI 60880:2006, 13.3.1, 13.3.2, 13.3.3, 13.3.4, 13.3.7 et 13.3.8 doit être effectuée lorsque c'est applicable pour

traiter les aspects du processus de développement du HPD qui peuvent conduire à la DCC d'instanciations multiples de la conception du HPD (et ne sont pas déjà traités par des articles de la présente norme).

NOTE 2 Certaines exigences de ces articles traitent des DCC de plusieurs systèmes au niveau de l'architecture d'I&C, alors qu'elles seraient mieux placées dans la norme de plus haut niveau dédiée aux DCC, la CEI 62340. Afin de préserver la structure de la collection de normes du SC 45A, il est proposé de déplacer ces exigences dans la CEI 62340 lors du prochain cycle de révision.

Annexe A
(informative)

Documentation

Cette annexe identifie une documentation typique pour chacun des principaux articles de la présente norme. Le contenu peut être organisé en un ensemble de documents autres que ceux suggérés dans cette annexe, pourvu que les articles soient clairement identifiés.

A.1 Projet

a) plan de gestion du projet

b) plan d'assurance qualité

c) plan de gestion de configuration

A.2 Spécification des exigences du HPD

a) spécification des exigences

b) rapport d'analyse des exigences

c) rapport de revue

A.3 Processus d'acceptation des circuits intégrés programmables, des blocs natifs et des blocs prédéveloppés

a) spécification des exigences du composant

b) document de sûreté utilisateur

c) autres documents du composant, incluant toute information telle que spécification, conception, test, expérience de fonctionnement

d) rapport d'analyse

e) document contenant les règles d'utilisation

f) rapport de revue d'acceptation

A.4 Conception et réalisation du HPD

a) spécification de conception incluant:

1) description: de la décomposition en modules principaux, des choix de conception défensive, de la technologie micro-électronique, des outils, des blocs natifs et PDB

2) description de la conception détaillée incluant:

- description RTL
- choix d'organisation (modules, sous-modules, interfaces, protocoles, etc.)
- caractéristiques électriques et temporelles préliminaires

3) description de la réalisation incluant:

- description au niveau des portes (« netlist »), description spécifique à la technologie en vue de la production, rétro-annotations
- réalisation des modules, des signaux critiques et de la distribution des alimentations électriques, options des outils, fichiers auxiliaires utilisés pour la réalisation tels que les fichiers de contraintes
- rapport d'analyses post-routage, rapport de STA
- analyse de la testabilité, vecteurs de test pour les tests périodiques

- caractéristiques électriques et temporelles détaillées

b) rapports de revue

A.5 Vérification du HPD

a) plan de vérification

b) document contenant la description du banc de test, les critères de couverture, les cas de test

c) document contenant l'analyse et la justification des critères de couverture

d) rapport incluant les résultats des tests et leur analyse (aux niveaux RTL, post-synthèse et post-routage), l'analyse du respect des règles d'utilisation

A.6 Aspects de l'intégration du système liés au HPD

a) plan d'intégration comprenant la stratégie et les procédures d'intégration, l'interface avec la gestion de configuration, les cas de test

b) aspects spécifiques du rapport de test du système intégré comprenant l'identification des composants et des outils, les résultats des tests et leur analyse, les défauts constatés et leur résolution

c) rapport de revue de l'intégration

A.7 Aspects de la validation du système liés au HPD

a) plan de validation incluant les cas de test

b) rapport comprenant l'identification des composants et des outils, les résultats des tests et leur analyse, les défauts constatés et leur résolution

A.8 Modification

L'annexe F de la CEI 60880 indique la liste de documents typique relative au processus de modification:

a) compte-rendu d'anomalie

b) demande de modification

c) compte-rendu de modification

d) historique de la gestion des modifications

De plus, les documents associés aux phases de développement affectées par la modification nécessitent d'être actualisés.

A.9 Production du HPD

a) document contenant les tests de production

b) document contenant les résultats des tests de production et les informations d'identification et de programmation des circuits

A.10 Outils logiciels pour le développement des HPD

a) rapport de sélection des outils (analyse du support des outils, évaluation, acceptation, limites d'applicabilité)

b) document décrivant la stratégie de modification, de mise à niveau ou de remplacement

Annexe B
(informative)

Développement des HPD

Les activités de développement traitées par la présente norme sont basées sur des langages de description de matériel (HDL) et des outils de conception fonctionnant sur des stations de travail, selon un processus dont les grandes lignes sont présentées ici pour faciliter la compréhension des articles correspondants de la présente norme.

B.1 Capture optionnelle des exigences au niveau système électronique (ESL)

La capture des exigences est parfois effectuée au moyen d'une description de haut niveau du système auquel le HPD appartient, englobant les autres composants matériels et logiciels. Chaque composant est représenté par un modèle comportemental, et ces modèles échangent des informations par des canaux de transmission pour simuler le système à développer.

Ce niveau de description est appelé « niveau système électronique », ou ESL, et recourt à des langages de description de système tels que SystemC ou System Verilog.

Cette description est typiquement exécutée (par simulation) avec des cas de test fonctionnels pour évaluer différentes architectures du système, choisir la meilleure, puis établir les exigences de chaque composant, dont le HPD, en termes de comportement et d'interface.

B.2 Conception

A partir des exigences, cette activité vise à définir les principes de conception, comme la répartition en modules prédéveloppés ou sur mesure, l'organisation de l'autosurveillance et l'identification de la technologie micro-électronique (incluant ses blocs natifs) et des PDB qui pourraient être utilisés.

Puis une description au niveau transfert de registre (RTL) est créée et testée par simulation. Des HDL tels que VHDL ou Verilog sont employés. Ce niveau ne dépend quasiment pas de la technologie micro-électronique qui sera utilisée.

Cette description de haut niveau est un modèle parallèle synchrone du HPD, décrivant son comportement au moyen de signaux transformés par des fonctions combinatoires et transférés séquentiellement entre des registres déclenchés par une ou plusieurs horloges.

La description RTL comporte des aspects structurels, montrant les relations logiques entre des modules conçus spécifiquement ou tirés de bibliothèques. Elle comporte aussi des aspects comportementaux, qui permettent de décrire la fonction d'un module au moyen de descriptions algorithmiques. Cette description est réalisée au moyen d'un langage de description de matériel (HDL), typiquement VHDL (IEEE 1076) ou Verilog (IEEE 1364).

Il est nécessaire que la description RTL soit synthétisable, pour qu'elle puisse être automatiquement traduite en un ensemble de portes électroniques interconnectées. Pour cela, le concepteur n'utilise qu'un sous-ensemble du langage HDL, alors que le langage complet peut servir par exemple à créer des environnements de simulation.

Parallèlement à la conception, un « banc de test » est souvent créé avec le même langage: la description RTL du HPD est intégrée dans un programme HDL plus large qui lui envoie des entrées et lit ses sorties pour la tester par simulation. Le banc de test peut employer des caractéristiques non synthétisables du langage pour faciliter la conception des tests (par ex.,

accès aux fichiers, impression, gestion du temps). Le banc de test sert à vérifier la description RTL et peut être associé à des outils pour produire des tests et mesurer leur couverture.

L'introduction d'outils d'analyse statique offre une approche de vérification complémentaire. Ils permettent typiquement de démontrer qu'une description HDL possède ou non certaines propriétés. Comme exemples d'analyses statiques, on citera: la vérification de propriétés, la vérification basée sur les assertions, la preuve d'équivalence entre différents niveaux de conception (par ex., RTL et liste d'interconnexions) ou l'analyse temporelle statique (STA).

B.3 Réalisation

A partir de la description RTL, une description électronique est produite pour permettre la réalisation dans la technologie micro-électronique choisie. Les principales étapes sont la synthèse logique et le placement et routage.

Les différentes familles de composants telles que FPGA, circuits pré-caractérisés, etc., fournissent différentes pré-caractérisations du comportement physique du produit final. Ainsi les activités décrites ci-après, intrinsèquement nécessaires, peuvent ou non être prises en charge automatiquement par les outils associés. La description suivante donne une vue d'ensemble de ces activités pour une conception basée sur des cellules pré-caractérisées.

La synthèse logique transforme la description RTL en un réseau de cellules logiques de la technologie micro-électronique, appelé « liste d'interconnexions ». Selon la technologie, ces cellules peuvent être uniquement des portes élémentaires (par exemple ET, OU), ou peuvent inclure des fonctions plus importantes (telles que des compteurs).

Bien que la synthèse utilise des outils similaires aux compilateurs logiciels, le concepteur dirige le processus en fournissant des informations sur les performances attendues (comme la fréquence d'horloge, le retard entre deux signaux, la consommation) et sur la façon d'implanter des signaux critiques tels que des horloges. Ces informations sont typiquement placées dans des « fichiers de contraintes » qui peuvent être très gros. Leur élaboration peut donc être difficile, et une erreur ou omission peut mener à un circuit affecté de défauts subtils, non reproductibles, pratiquement impossibles à détecter par simulation. La vérification des fichiers de contraintes est donc une activité essentielle.

L'étape de placement et routage définit l'emplacement physique des cellules sur le silicium, et les interconnecte en tenant compte des contraintes technologiques (existence et capacité de canaux de routage prédéfinis) et des contraintes de l'application (telles que le temps de propagation maximal entre deux nœuds donnés).

Quand le nombre de portes croît, de plus en plus d'interconnexions sont requises et doivent être acheminées à travers la puce. De plus, les exigences de vitesse imposent généralement de garder certains chemins courts. Cette dernière contrainte peut conduire à déplacer certaines portes, ce qui impacte le schéma global de routage. Trouver la « meilleure » solution est très difficile (au sens des capacités de calcul), aussi seules des approximations peuvent être trouvées par les outils, qui doivent utiliser des algorithmes avancés et évolutifs.

La description après placement et routage est fournie dans un format dépendant de la technologie micro-électronique. La topologie physique étant alors connue, les temps de propagation peuvent être affinés selon la résistance et la capacité de chaque chemin. Ces informations sont utilisées pour rétro-annoter la description, afin de la simuler sur le banc de test avec des temps de propagation réalistes pour les cellules et les interconnexions.

Le fournisseur de la technologie micro-électronique indique les temps de propagation des cellules de sa bibliothèque, dans des formats comme VHDL-VITAL (IEEE 1076.4). La liste d'interconnexions est rétro-annotée avec ces informations, qui sont ainsi prises en compte dans la simulation « post-réalisation ».

En plus de la vérification par « simulation post-réalisation », des outils d'analyse statique permettent de vérifier les temps de propagation (STA), ou l'équivalence entre différents niveaux de description.

B.4 Types de circuits intégrés spécifiques

B.4.1 Considérations générales

La technologie évoluant et offrant de nombreuses variantes de circuits intégrés spécifiques, la présente norme traite des principes et non des détails spécifiques de chaque variante.

Cet article donne une vue d'ensemble des principales variantes disponibles (note: leurs noms ne sont pas toujours employés de façon cohérente dans l'industrie).

D'un point de vue théorique, toute fonction calculable peut être réalisée avec un seul type de porte élémentaire bien choisie telle que "NAND" ("*A nand B*" signifie "*non (A et B)*". Par conséquent, la gamme de fonctions qui peuvent être réalisées à l'intérieur d'un circuit donné dépend essentiellement de la taille de celui-ci (nombre de portes) et de sa connectivité interne qui permet une utilisation plus ou moins efficace des portes.

B.4.2 PAL (Logique à réseau programmable, Programmable Array Logic)

Un PAL est un petit circuit organisé en un réseau OU/ET permettant de réaliser des équations logiques se présentant comme des sommes de produits telles que: *sortie = (A et B et non C) ou (non B et non C) ou (D).*

Les PAL sont personnalisés en configurant des connexions, typiquement en faisant fondre des fusibles ou parfois en configurant des commutateurs reprogrammables.

La structure ET est programmable: l'expression du produit avant programmation est: (*A et non A et B et non B et C et non C, etc.)*, où chaque terme correspond à un fusible. En fonction des exigences fonctionnelles, les termes inutiles sont supprimés en faisant fondre les fusibles correspondants, pour générer par exemple (*A et non C*).

La structure OU est fixe: les entrées du "OU" sont un nombre fixe de produits programmables comme ci-dessus, par exemple *(A et non C) ou (A et non B) ou (D)*.

Des langages de bas niveau tels que PALASM sont typiquement utilisés pour personnaliser les PAL: le concepteur saisit les équations logiques à réaliser, et l'outil les traduit en une carte de fusibles. Ces langages ne permettent pas de créer des descriptions comportementales comme en VHDL ou Verilog.

Les circuits PAL offrent typiquement quelques entrées et sorties (par exemple 10 entrées, 8 sorties), et sont équivalents au plus à quelques centaines de portes. En raison de cette taille limitée, ils sont hors du domaine d'application de la présente norme.

B.4.3 PLD, CPLD (Réseau logique programmable [complexe])

Les PLD et CPLD sont équivalents à de multiples PAL interconnectés, mais de nouvelles familles peuvent offrir des fonctionnalités supplémentaires.

Comme les PAL, ils sont basés sur une somme de produits de structure fixe, le trajet du signal de l'entrée à la sortie est donc fixé et les temps de propagation sont assez constants. Bien sûr, si des fonctionnalités supplémentaires comme des chemins de retour ou des logiques spécialisées sont incluses, cette propriété peut être perdue.

La taille des CPLD atteint l'équivalent de plusieurs dizaines de milliers de portes.

B.4.4 FPGA

Un FPGA comprend un grand nombre de blocs logiques programmables pouvant chacun réaliser une logique combinatoire et un stockage, des interconnexions programmables entre blocs et des plots d'entrée/sortie (de direction, impédance, tension et type de mémorisation typiquement programmables). Des canaux spécifiques sont souvent fournis pour les signaux critiques tels que les horloges. Un FPGA peut également contenir des blocs logiques spécialisés tels que mémoires, cœurs de processeur, des interfaces normalisées, etc.

L'équivalence des FPGA en nombre de portes n'est pas significative, car leur complexité et leurs structures diverses rendent difficile la prévision du nombre de blocs nécessaires pour réaliser une fonction donnée. Certains FPGA comprennent des centaines de milliers de blocs programmables, des centaines d'entrées/sorties, et sont constitués de milliards de transistors.

Les FPGA peuvent conserver leurs fonctions (« configurations ») par des moyens comme:

a) SRAM (la configuration est volatile, copiée au départ à partir d'une mémoire externe),
b) mémoire flash (la configuration est stockée dans des éléments de mémoire interne non volatiles mais reprogrammables),
c) anti-fusible (la configuration est permanente; ces circuits sont programmables une fois).

La sensibilité de la configuration à un SEU et au rayonnement neutron/alpha est forte les SRAM, faible pour les mémoires flash, et très faible pour les anti-fusibles.

B.4.5 Réseau de portes, ou circuit intégré prédiffusé

Le fournisseur de circuits intégrés prépare des circuits standards dont tous les transistors sont fabriqués mais non interconnectés. La fonction spécifique à réaliser est synthétisée par une interconnexion spécifique de ces transistors.

Cette approche implique des coûts non récurrents pour la fabrication des masques spécifiques aux couches métalliques (interconnexions), mais elle permet un coût unitaire inférieur à celui des FPGA car elle n'utilise pas de silicium pour le circuit de programmation. Cependant, cette technologie semble de plus en plus remplacée par les FPGA.

B.4.6 Circuits pré-caractérisés (standard cells)

Le fournisseur propose une technologie micro-électronique avec laquelle il conçoit un ensemble fixé de cellules, telles que des portes combinatoires, bascules, additionneurs, compteurs, etc. Ces cellules ont des caractéristiques connues, comme la surface, le courant d'entrée, la capacité et le temps de propagation. Elles sont conçues de façon à avoir la même hauteur et des largeurs éventuellement différentes, et peuvent donc être placées sur la puce en rangées pour faciliter le routage et l'alimentation.

Les caractéristiques fonctionnelles et physiques des cellules sont décrites dans la bibliothèque technologique, fournie au concepteur du système d'I&C. Cette bibliothèque est utilisée pour la synthèse logique (voir Article B.3) qui transforme la description RTL en liste d'interconnexions de ces cellules, qui sont ensuite placées sur la puce et interconnectées. Après les vérifications fonctionnelles et technologiques, les masques nécessaires pour produire les circuits intégrés sont fabriqués et la production peut démarrer.

Cette approche implique des coûts non récurrents supérieurs à ceux des réseaux de portes car tous les masques sont spécifiques, mais elle permet un coût unitaire inférieur car la taille de la puce correspond exactement aux besoins. La disponibilité de différentes cellules de chaque type, optimisant différents aspects comme la vitesse, la surface ou la consommation, permet une meilleure optimisation de chaque partie de la conception, sous le contrôle du concepteur d'I&C au moyen d'outils associés aux HDL.

B.4.7 ASIC entièrement sur mesure (Full custom ASIC, raw ASIC)

Cette technologie implique une conception spécifique de tous les aspects du circuit intégré, jusqu'au niveau des transistors, avec des outils spécifiques. Cela entraîne des coûts non récurrents très élevés qui nécessitent de grands volumes pour se justifier économiquement. Ces circuits ne sont pas dans le domaine d'application de la présente norme.

Bibliographie

CEI 60780, *Centrales nucléaires – Equipements électriques de sûreté – Qualification*

CEI 61226, *Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Classement des fonctions d'instrumentation et de contrôle-commande*

CEI 62342, *Centrales nucléaires de puissance – Systèmes d'instrumentation et de contrôle-commande importants pour la sûreté – Gestion du vieillissement*

ISO 9001, *Systèmes de management de la qualité – Exigences*

ISO/IEC 25000, *Ingénierie du logiciel – Exigences de qualité du produit logiciel et évaluation (SQuaRE)*

_____

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

3, rue de Varembé
PO Box 131
CH-1211 Geneva 20
Switzerland

Tel: + 41 22 919 02 11
Fax: + 41 22 919 03 00
info@iec.ch
www.iec.ch