

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Functional safety of electrical/electronic/programmable electronic
safety-related systems –
Part 7: Overview of techniques and measures**

**Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques
programmables relatifs à la sécurité –
Partie 7: Présentation de techniques et mesures**

WILSON



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2000 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester.

If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de la CEI ou du Comité national de la CEI du pays du demandeur.

Si vous avez des questions sur le copyright de la CEI ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de la CEI de votre pays de résidence.

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

About the IEC

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

About IEC publications

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

- Catalogue of IEC publications: www.iec.ch/searchpub

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 It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub

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

- Electropedia: www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv

If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch

Tel.: +41 22 919 02 11

Fax: +41 22 919 03 00

A propos de la CEI

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

A propos des publications CEI

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

- Catalogue des publications de la CEI: www.iec.ch/searchpub/cur_fut-f.htm

Le Catalogue en-ligne de la CEI vous permet d'effectuer des recherches en utilisant différents critères (numéro de référence, texte, comité d'études,...). Il donne aussi des informations sur les projets et les publications retirées ou remplacées.

- Just Published CEI: www.iec.ch/online_news/justpub

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

- Electropedia: www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 20 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International en ligne.

- Service Clients: www.iec.ch/webstore/custserv/custserv_entry-f.htm

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions, visitez le FAQ du Service clients ou contactez-nous:

Email: csc@iec.ch

Tél.: +41 22 919 02 11

Fax: +41 22 919 03 00

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Functional safety of electrical/electronic/programmable electronic
safety-related systems –
Part 7: Overview of techniques and measures**

**Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques
programmables relatifs à la sécurité –
Partie 7: Présentation de techniques et mesures**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX

XE

ICS 25.040.40; 35.240.50

ISBN 2-8318-5151-3

CONTENTS

	Page
FOREWORD.....	8
INTRODUCTION.....	10
Clause	
1 Scope.....	12
2 Normative references.....	14
3 Definitions and abbreviations.....	14
Annex A (informative) Overview of techniques and measures for E/E/PES: control of random hardware failures (see IEC 61508-2).....	15
A.1 Electrical.....	15
A.1.1 Failure detection by on-line monitoring	15
A.1.2 Monitoring of relay contacts	15
A.1.3 Comparator	15
A.1.4 Majority voter	16
A.1.5 Idle current principle (de-energised to trip).....	16
A.2 Electronic	16
A.2.1 Tests by redundant hardware.....	16
A.2.2 Dynamic principles.....	17
A.2.3 Standard test access port and boundary-scan architecture	17
A.2.4 Fail-safe hardware	17
A.2.5 Monitored redundancy.....	18
A.2.6 Electrical/electronic components with automatic check.....	18
A.2.7 Analogue signal monitoring.....	18
A.2.8 De-rating.....	19
A.3 Processing units.....	19
A.3.1 Self-test by software: limited number of patterns (one-channel).....	19
A.3.2 Self-test by software: walking bit (one-channel)	19
A.3.3 Self-test supported by hardware (one-channel)	19
A.3.4 Coded processing (one-channel)	20
A.3.5 Reciprocal comparison by software	20
A.4 Invariable memory ranges	20
A.4.1 Word-saving multi-bit redundancy (for example ROM monitoring with a modified Hamming code)	20
A.4.2 Modified checksum	21
A.4.3 Signature of one word (8-bit).....	21
A.4.4 Signature of a double word (16-bit)	21
A.4.5 Block replication (for example double ROM with hardware or software comparison).....	22
A.5 Variable memory ranges	22
A.5.1 RAM test "checkerboard" or "march".....	22
A.5.2 RAM test "walkpath".....	23
A.5.3 RAM test "galpat" or "transparent galpat".....	23
A.5.4 RAM test "Abraham"	24
A.5.5 One-bit redundancy (for example RAM monitoring with a parity bit).....	24
A.5.6 RAM monitoring with a modified Hamming code, or detection of data failures with error-detection-correction codes (EDC)	24
A.5.7 Double RAM with hardware or software comparison and read/write test	25

Clause	Page
A.6 I/O-units and interfaces (external communication)	25
A.6.1 Test pattern.....	25
A.6.2 Code protection.....	25
A.6.3 Multi-channel parallel output	26
A.6.4 Monitored outputs	26
A.6.5 Input comparison/voting	27
A.7 Data paths (internal communication)	27
A.7.1 One-bit hardware redundancy.....	27
A.7.2 Multi-bit hardware redundancy	27
A.7.3 Complete hardware redundancy	27
A.7.4 Inspection using test patterns	28
A.7.5 Transmission redundancy	28
A.7.6 Information redundancy	28
A.8 Power supply.....	28
A.8.1 Overvoltage protection with safety shut-off	28
A.8.2 Voltage control (secondary)	29
A.8.3 Power-down with safety shut-off	29
A.9 Temporal and logical program sequence monitoring.....	29
A.9.1 Watch-dog with separate time base without time-window	29
A.9.2 Watch-dog with separate time base and time-window.....	30
A.9.3 Logical monitoring of program sequence	30
A.9.4 Combination of temporal and logical monitoring of program sequences	30
A.9.5 Temporal monitoring with on-line check.....	30
A.10 Ventilation and heating.....	31
A.10.1 Temperature sensor.....	31
A.10.2 Fan control.....	31
A.10.3 Actuation of the safety shut-off via thermal fuse.....	31
A.10.4 Staggered message from thermo-sensors and conditional alarm.....	31
A.10.5 Connection of forced-air cooling and status indication	31
A.11 Communication and mass-storage	32
A.11.1 Separation of electrical energy lines from information lines	32
A.11.2 Spatial separation of multiple lines.....	32
A.11.3 Increase of interference immunity	32
A.11.4 Antivalent signal transmission	33
A.12 Sensors.....	33
A.12.1 Reference sensor.....	33
A.12.2 Positive-activated switch	33
A.13 Final elements (actuators)	33
A.13.1 Monitoring	33
A.13.2 Cross-monitoring of multiple actuators.....	34
A.14 Measures against the physical environment.....	34
Annex B (informative) Overview of techniques and measures for E/E/PES: avoidance of systematic failures (see IEC 61508-2 and IEC 61508-3).....	35
B.1 General measures and techniques.....	35
B.1.1 Project management.....	35
B.1.2 Documentation.....	36
B.1.3 Separation of safety-related systems from non-safety-related systems	37
B.1.4 Diverse hardware	37

Clause	Page
B.2 E/E/PES safety requirements specification	38
B.2.1 Structured specification	38
B.2.2 Formal methods	38
B.2.3 Semi-formal methods	39
B.2.3.1 General	39
B.2.3.2 Finite state machines/state transition diagrams	39
B.2.3.3 Time Petri nets	40
B.2.4 Computer-aided specification tools	40
B.2.4.1 General	40
B.2.4.2 Tools oriented towards no specific method	41
B.2.4.3 Model orientated procedure with hierarchical analysis	41
B.2.4.4 Entity models	41
B.2.4.5 Incentive and answer	42
B.2.5 Checklists	42
B.2.6 Inspection of the specification	43
B.3 E/E/PES design and development	43
B.3.1 Observance of guidelines and standards	43
B.3.2 Structured design	44
B.3.3 Use of well-tried components	45
B.3.4 Modularisation	45
B.3.5 Computer-aided design tools	46
B.3.6 Simulation	46
B.3.7 Inspection (reviews and analysis)	46
B.3.8 Walk-through	47
B.4 E/E/PES operation and maintenance procedures	47
B.4.1 Operation and maintenance instructions	47
B.4.2 User friendliness	48
B.4.3 Maintenance friendliness	48
B.4.4 Limited operation possibilities	48
B.4.5 Operation only by skilled operators	49
B.4.6 Protection against operator mistakes	49
B.4.7 (Not used)	49
B.4.8 Modification protection	49
B.4.9 Input acknowledgement	49
B.5 E/E/PES integration	50
B.5.1 Functional testing	50
B.5.2 Black-box testing	50
B.5.3 Statistical testing	51
B.5.4 Field experience	51
B.6 E/E/PES safety validation	52
B.6.1 Functional testing under environmental conditions	52
B.6.2 Interference surge immunity testing	53
B.6.3 (Not used)	53
B.6.4 Static analysis	53
B.6.5 Dynamic analysis	54

Clause	Page
B.6.6 Failure analysis	54
B.6.6.1 Failure modes and effects analysis	54
B.6.6.2 Cause consequence diagrams	55
B.6.6.3 Event tree analysis	55
B.6.6.4 Failure modes, effects and criticality analysis.....	55
B.6.6.5 Fault tree analysis	56
B.6.7 Worst-case analysis	56
B.6.8 Expanded functional testing	56
B.6.9 Worst-case testing	57
B.6.10 Fault insertion testing.....	57
Annex C (informative) Overview of techniques and measures for achieving software safety integrity (see IEC 61508-3)	58
C.1 General	58
C.2 Requirements and detailed design	58
C.2.1 Structured methods.....	58
C.2.1.1 General	58
C.2.1.2 CORE – Controlled Requirements Expression.....	59
C.2.1.3 JSD – Jackson System Development.....	59
C.2.1.4 MASCOT – Modular Approach to Software Construction, Operation and Test.....	60
C.2.1.5 Real-time Yourdon.....	60
C.2.1.6 SADT – Structured Analysis and Design Technique.....	61
C.2.2 Data flow diagrams	62
C.2.3 Structure diagrams.....	63
C.2.4 Formal methods.....	63
C.2.4.1 General	63
C.2.4.2 CCS – Calculus of Communicating Systems.....	64
C.2.4.3 CSP – Communicating Sequential Processes.....	64
C.2.4.4 HOL – Higher Order Logic.....	65
C.2.4.5 LOTOS.....	65
C.2.4.6 OBJ.....	65
C.2.4.7 Temporal logic.....	66
C.2.4.8 VDM, VDM++ – Vienna Development Method.....	67
C.2.4.9 Z.....	68
C.2.5 Defensive programming	69
C.2.6 Design and coding standards.....	70
C.2.6.1 General	70
C.2.6.2 Coding standards	70
C.2.6.3 No dynamic variables or dynamic objects	71
C.2.6.4 On-line checking during creation of dynamic variables or dynamic objects	71
C.2.6.5 Limited use of interrupts	71
C.2.6.6 Limited use of pointers	72
C.2.6.7 Limited use of recursion	72
C.2.7 Structured programming	72
C.2.8 Information hiding/encapsulation.....	73
C.2.9 Modular approach	74
C.2.10 Use of trusted/verified software modules and components	74

Clause	Page
C.3 Architecture design.....	75
C.3.1 Fault detection and diagnosis	75
C.3.2 Error detecting and correcting codes	76
C.3.3 Failure assertion programming.....	76
C.3.4 Safety bag.....	77
C.3.5 Software diversity (diverse programming)	77
C.3.6 Recovery block	78
C.3.7 Backward recovery.....	79
C.3.8 Forward recovery	79
C.3.9 Re-try fault recovery mechanisms.....	79
C.3.10 Memorising executed cases.....	80
C.3.11 Graceful degradation.....	80
C.3.12 Artificial intelligence fault correction	81
C.3.13 Dynamic reconfiguration	81
C.4 Development tools and programming languages.....	82
C.4.1 Strongly typed programming languages.....	82
C.4.2 Language subsets.....	82
C.4.3 Certified tools and certified translators.....	83
C.4.4 Tools and translators: increased confidence from use	83
C.4.4.1 Comparison of source program and executable code	84
C.4.5 Library of trusted/verified software modules and components.....	84
C.4.6 Suitable programming languages.....	85
C.5 Verification and modification	88
C.5.1 Probabilistic testing	88
C.5.2 Data recording and analysis.....	89
C.5.3 Interface testing	89
C.5.4 Boundary value analysis	89
C.5.5 Error guessing.....	90
C.5.6 Error seeding	90
C.5.7 Equivalence classes and input partition testing.....	91
C.5.8 Structure-based testing	91
C.5.9 Control flow analysis	92
C.5.10 Data flow analysis	93
C.5.11 Sneak circuit analysis.....	93
C.5.12 Symbolic execution	94
C.5.13 Formal proof.....	94
C.5.14 Complexity metrics.....	95
C.5.15 Fagan inspections	95
C.5.16 Walk-throughs/design reviews	96
C.5.17 Prototyping/animation	96
C.5.18 Process simulation.....	97
C.5.19 Performance requirements.....	97
C.5.20 Performance modelling	98
C.5.21 Avalanche/stress testing	98
C.5.22 Response timing and memory constraints	99
C.5.23 Impact analysis	99
C.5.24 Software configuration management.....	100

Clause	Page
C.6 Functional safety assessment	100
C.6.1 Decision tables (truth tables).....	100
C.6.2 Hazard and Operability Study (HAZOP)	100
C.6.3 Common cause failure analysis	102
C.6.4 Markov models.....	102
C.6.5 Reliability block diagrams.....	103
C.6.6 Monte-Carlo simulation	104
Annex D (informative) A probabilistic approach to determining software safety integrity for pre-developed software.....	105
D.1 General	105
D.2 Statistical testing formulae and examples of their use.....	106
D.2.1 Simple statistical test for low demand mode of operation.....	106
D.2.1.1 Prerequisites	106
D.2.1.2 Results	106
D.2.1.3 Example	106
D.2.2 Testing of an input space (domain) for a low demand mode of operation	106
D.2.2.1 Prerequisites	106
D.2.2.2 Results	106
D.2.2.3 Example	107
D.2.3 Simple statistical test for high demand or continuous mode of operation	107
D.2.3.1 Prerequisites	107
D.2.3.2 Results	107
D.2.3.3 Example	108
D.2.4 Complete test.....	108
D.2.4.1 Prerequisites	108
D.2.4.2 Results	108
D.2.4.3 Example	109
D.3 References	109
Bibliography	110
Index	112
Table C.1 – Recommendations for specific programming languages	87
Table D.1 – Necessary history for confidence to safety integrity levels	105
Table D.2 – Probabilities of failure for low demand mode of operation	106
Table D.3 – Mean distances of two test points.....	107
Table D.4 – Probabilities of failure for high demand or continuous mode of operation	108
Table D.5 – Probability of testing all program properties.....	109

INTERNATIONAL ELECTROTECHNICAL COMMISSION

FUNCTIONAL SAFETY OF ELECTRICAL/ELECTRONIC/ PROGRAMMABLE ELECTRONIC SAFETY-RELATED SYSTEMS –

Part 7: Overview of techniques and measures

FOREWORD

- 1) The IEC (International Electrotechnical Commission) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of the 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, the IEC publishes International Standards. 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. The 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 the 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 National Committees.
- 3) The documents produced have the form of recommendations for international use and are published in the form of standards, technical specifications, technical reports or guides and they are accepted by the National Committees in that sense.
- 4) In order to promote international unification, IEC National Committees undertake to apply IEC International Standards transparently to the maximum extent possible in their national and regional standards. Any divergence between the IEC Standard and the corresponding national or regional standard shall be clearly indicated in the latter.
- 5) The IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with one of its standards.
- 6) Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. The IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 61508-7 has been prepared by subcommittee 65A: System aspects, of IEC technical committee 65: Industrial-process measurement and control.

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

FDIS	Report on voting
65A/293/FDIS	65A/299/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 3.

Annexes A, B, C and D are for information only.

IEC 61508 consists of the following parts, under the general title *Functional safety of electrical/electronic/programmable electronic safety-related systems*:

- Part 1: General requirements
- Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems
- Part 3: Software requirements
- Part 4: Definitions and abbreviations
- Part 5: Examples of methods for the determination of safety integrity levels
- Part 6: Guidelines on the application of IEC 61508-2 and IEC 61508-3
- Part 7: Overview of techniques and measures

The committee has decided that the contents of this publication will remain unchanged until 2006. At this date, the publication will be

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

Withdrawn

INTRODUCTION

Systems comprised of electrical and/or electronic components have been used for many years to perform safety functions in most application sectors. Computer-based systems (generically referred to as programmable electronic systems (PESs)) are being used in all application sectors to perform non-safety functions and, increasingly, to perform safety functions. If computer system technology is to be effectively and safely exploited, it is essential that those responsible for making decisions have sufficient guidance on the safety aspects on which to make those decisions.

This International Standard sets out a generic approach for all safety lifecycle activities for systems comprised of electrical and/or electronic and/or programmable electronic components (electrical/electronic/programmable electronic systems (E/E/PESs)) that are used to perform safety functions. This unified approach has been adopted in order that a rational and consistent technical policy be developed for all electrically based safety-related systems. A major objective is to facilitate the development of application sector standards.

In most situations, safety is achieved by a number of protective systems which rely on many technologies (for example mechanical, hydraulic, pneumatic, electrical, electronic, programmable electronic). Any safety strategy must therefore consider not only all the elements within an individual system (for example sensors, controlling devices and actuators) but also all the safety-related systems making up the total combination of safety-related systems. Therefore, while this International Standard is concerned with electrical/electronic/programmable electronic (E/E/PE) safety-related systems, it may also provide a framework within which safety-related systems based on other technologies may be considered.

It is recognised that there is a great variety of E/E/PES applications in a variety of application sectors and covering a wide range of complexity, hazard and risk potentials. In any particular application, the exact prescription of safety measures will be dependent on many factors specific to the application. This International Standard, by being generic, will enable such a prescription to be formulated in future application sector International Standards.

This International Standard

- considers all relevant overall, E/E/PES and software safety lifecycle phases (for example, from initial concept, through design, implementation, operation and maintenance to decommissioning) when E/E/PESs are used to perform safety functions;
- has been conceived with a rapidly developing technology in mind; the framework is sufficiently robust and comprehensive to cater for future developments;
- enables application sector International Standards, dealing with safety-related E/E/PESs, to be developed; the development of application sector international standards, within the framework of this standard, should lead to a high level of consistency (for example, of underlying principles, terminology, etc.) both within application sectors and across application sectors; this will have both safety and economic benefits;
- provides a method for the development of the safety requirements specification necessary to achieve the required functional safety for E/E/PE safety-related systems;

- uses safety integrity levels for specifying the target level of safety integrity for the safety functions to be implemented by the E/E/PE safety-related systems;
- adopts a risk-based approach for the determination of the safety integrity level requirements;
- sets numerical target failure measures for E/E/PE safety-related systems which are linked to the safety integrity levels;
- sets a lower limit on the target failure measures, in a dangerous mode of failure, that can be claimed for a single E/E/PE safety-related system; for E/E/PE safety-related systems operating in
 - a low demand mode of operation, the lower limit is set at an average probability of failure of 10^{-5} to perform its design function on demand;
 - a high demand or continuous mode of operation, the lower limit is set at a probability of a dangerous failure of 10^{-9} per hour;

NOTE A single E/E/PE safety-related system does not necessarily mean a single-channel architecture.

- adopts a broad range of principles, techniques and measures to achieve functional safety for E/E/PE safety-related systems, but does not rely on the concept of fail-safe, which may be of value when the failure modes are well defined and the level of complexity is relatively low – the concept of fail-safe was considered inappropriate because of the full range of complexity of E/E/PE safety-related systems that are within the scope of the standard.

Withdrawal

FUNCTIONAL SAFETY OF ELECTRICAL/ELECTRONIC/ PROGRAMMABLE ELECTRONIC SAFETY-RELATED SYSTEMS –

Part 7: Overview of techniques and measures

1 Scope

1.1 This part of IEC 61508 contains an overview of various safety techniques and measures relevant to IEC 61508-2 and IEC 61508-3.

NOTE The references should be considered as basic references to methods and tools or as examples, and may not represent the state of the art.

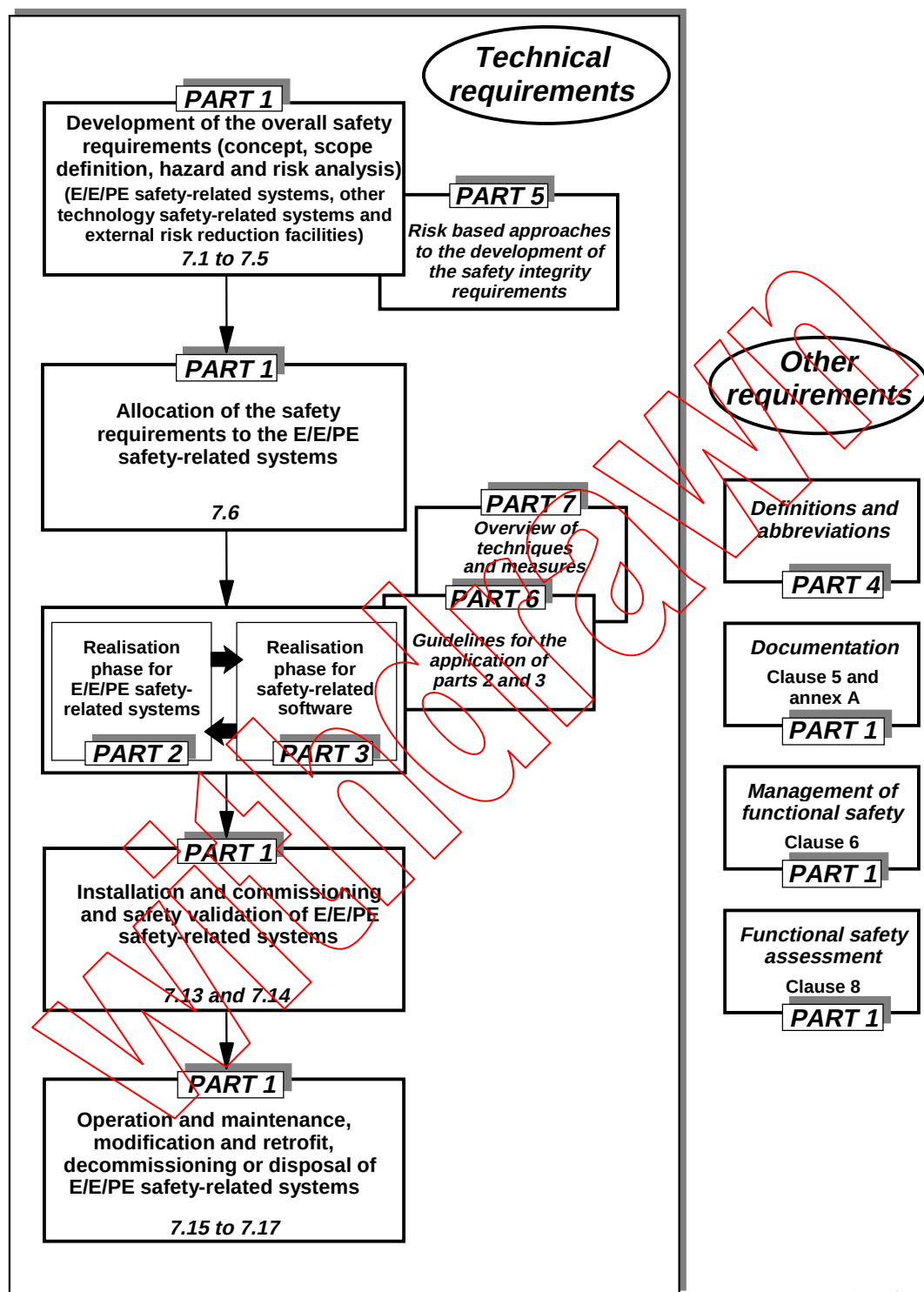
1.2 IEC 61508-1, IEC 61508-2, IEC 61508-3 and IEC 61508-4 are basic safety publications, although this status does not apply in the context of low-complexity E/E/PE safety-related systems (see 3.4.4 of IEC 61508-4). As basic safety publications, they are intended for use by technical committees in the preparation of standards in accordance with the principles contained in IEC Guide 104 and ISO/IEC Guide 51. IEC 61508 is also intended for use as a stand-alone standard.

One of the responsibilities of a technical committee is, wherever applicable, to make use of basic safety publications in the preparation of its own publications. In this context, the requirements, test methods or test conditions of this basic safety publication will not apply unless specifically referred to or included in the publications prepared by those technical committees.

NOTE 1 The functional safety of an E/E/PE safety-related system can only be achieved when all related requirements are met. Therefore it is important that all related requirements are carefully considered and adequately referenced.

NOTE 2 In the USA and Canada, until the proposed process sector implementation of IEC 61508 (i.e. IEC 61511) is published as an international standard in the USA and Canada, existing national process safety standards based on IEC 61508 (i.e. ANSI/ISA S84.01-1996) can be applied to the process sector instead of IEC 61508.

1.3 Figure 1 shows the overall framework for parts 1 to 7 of this standard and indicates the role that IEC 61508-7 plays in the achievement of functional safety for E/E/PE safety-related systems.



IEC 225/2000

Figure 1 – Overall framework of IEC 61508

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of IEC 61508. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of IEC 61508 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

IEC 61508-1:1998, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 1: General requirements*

IEC 61508-2, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems*¹⁾

IEC 61508-3:1998, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 3: Software requirements*

IEC 61508-4:1998, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 4: Definitions and abbreviations of terms*

IEC 61508-5:1998, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 5 Examples of methods for the determination of safety integrity levels*

IEC 61508-6, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 6: Guidelines on the application of IEC 61508-2 and IEC 61508-3*¹⁾

IEC Guide 104:1997, *The preparation of safety publications and the use of basic safety publications and group safety publications*

IEC/ISO Guide 51:1990, *Guidelines for the inclusion of safety aspects in standards*

3 Definitions and abbreviations

For the purposes of this part of IEC 61508, the definitions and abbreviations given in IEC 61508-4 apply.

¹⁾ To be published.

Annex A (informative)

Overview of techniques and measures for E/E/PES: control of random hardware failures (see IEC 61508-2)

A.1 Electrical

Global objective: To control failures in electromechanical components.

A.1.1 Failure detection by on-line monitoring

NOTE This technique/measure is referenced in tables A.2, A.3, A.7 and A.14 to A.19 of IEC 61508-2.

Aim: To detect failures by monitoring the behaviour of the E/E/PE safety-related system in response to the normal (on-line) operation of the equipment under control (EUC).

Description: Under certain conditions, failures can be detected using information about (for example) the time behaviour of the EUC. For example, if a switch, which is part of the E/E/PE safety-related system, is normally actuated by the EUC, then if the switch does not change state at the expected time, a failure will have been detected. It is not usually possible to localise the failure.

A.1.2 Monitoring of relay contacts

NOTE This technique/measure is referenced in tables A.2 and A.15 of IEC 61508-2.

Aim: To detect failures (for example welding) of relay contacts.

Description: Forced contact (or positively guided contact) relays are designed so that their contacts are rigidly linked together. Assuming there are two sets of changeover contacts, *a* and *b*, if the normally open contact, *a*, welds, the normally closed contact, *b*, cannot close when the relay coil is next de-energised. Therefore, the monitoring of the closure of the normally closed contact *b* when the relay coil is de-energised may be used to prove that the normally open contact *a* has opened. Failure of normally closed contact *b* to close indicates a failure of contact *a*, so the monitoring circuit should ensure a safe shut-down, or ensure that shut-down is continued, for any machinery controlled by contact *a*.

References:

Zusammenstellung und Bewertung elektromechanischer Sicherheitsschaltungen für Verriegelungseinrichtungen. F. Kreutzkamp, W. Hertel, Sicherheitstechnisches Informations- und Arbeitsblatt 330212, BIA-Handbuch. 17. Lfg. X/91, Erich Schmidt Verlag, Bielefeld.

Anlagensicherung mit Mitteln der MSR-Technik. G. Strohrman, Oldenburg, 1983.

A.1.3 Comparator

NOTE This technique/measure is referenced in tables A.2, A.3, A.4 of IEC 61508-2.

Aim: To detect, as early as possible, (non-simultaneous) failures in an independent processing unit or in the comparator.

Description: The signals of independent processing units are compared cyclically or continuously by a hardware comparator. The comparator may itself be externally tested, or it may use self-monitoring technology. Detected differences in the behaviour of the processors lead to a failure message.

A.1.4 Majority voter

NOTE This technique/measure is referenced in tables A.2, A.3 and A.4 of IEC 61508-2.

Aim: To detect and mask failures in one of at least three hardware channels.

Description: A voting unit using the majority principle (2 out of 3, 3 out of 3, or m out of n) is used to detect and mask failures. The voter may itself be externally tested, or it may use self-monitoring technology.

References:

Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

Anlagensicherung mit Mitteln der MSR-Technik. Praxis der Sicherheitstechnik, Vol 1, Dechema, 1988.

Sicherung von Anlagen der Verfahrenstechnik mit Mitteln der Mess-, Steuerungs- und Regelungstechnik. VDI/VDE Blatt 1 to 5, 1984 to 1988.

A.1.5 Idle current principle (de-energised to trip)

NOTE This technique/measure is referenced in tables A.2, A.9, A.14 and A.15 of IEC 61508-2.

Aim: To execute the safety function if power is cut or lost.

Description: The safety function is executed if the contacts are open and no current flows. For example, if brakes are used to stop a dangerous movement of a motor, the brakes are opened by closing contacts in the safety-related system and are closed by opening the contacts in the safety-related system.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

A.2 Electronic

Global objective: To control failure in solid-state components.

A.2.1 Tests by redundant hardware

NOTE This technique/measure is referenced in tables A.3, A.16, A.17 and A.19 of IEC 61508-2.

Aim: To detect failures using hardware redundancy, i.e. using additional hardware not required to implement the process functions.

Description: Redundant hardware can be used to test at an appropriate frequency the specified safety functions. This approach is normally necessary for realising A.1.1 or A.2.2.

Reference: DIN V VDE 0801: Grundsätze für Rechner in Systemen mit Sicherheitsaufgaben (Principles for Computers in Safety-Related Systems), Beuth-Verlag, Berlin, 1990.

A.2.2 Dynamic principles

NOTE This technique/measure is referenced in table A.3 of IEC 61508-2.

Aim: To detect static failures by dynamic signal processing.

Description: A forced change of otherwise static signals (internally or externally generated) helps to detect static failures in components. This technique is often associated with electromechanical components.

Reference: Elektronik in der Sicherheitstechnik. H. Jürs, D. Reinert, Sicherheitstechnisches Informations- und Arbeitsblatt 330220, BIA-Handbuch, Erich-Schmidt Verlag, Bielefeld, 1993.

A.2.3 Standard test access port and boundary-scan architecture

NOTE This technique/measure is referenced in tables A.3, A.16 and A.19 of IEC 61508-2.

Aim: To control and observe what happens at each pin of an IC.

Description: Boundary-scan test is an IC design technique which increases the testability of the IC by resolving the problem of how to gain access to the circuit test points within it. In a typical boundary-scan IC, comprised of core logic and input and output buffers, a shift-register stage is placed between the core logic and the input and output buffers adjacent to each IC pin. Each shift-register stage is contained in a boundary-scan cell. The boundary-scan cell can control and observe what happens at each input and output pin of an IC, via the standard test access port. Internal testing of the IC core logic is accomplished by isolating the on-chip core logic from stimuli received from surrounding components, and then performing an internal self-test. These tests can be used to detect failures in the IC.

Reference: IEEE 1149.1:1990, Standard Test Access Port and Boundary-Scan Architecture.

A.2.4 Fail-safe hardware

NOTE This technique/measure is referenced in table A.3 of IEC 61508-2.

Aim: To put a system into a safe state if a failure occurs.

Description: In hard-wired systems, a unit is said to operate in a fail-safe manner if

- a defined set of faults will lead to a safe condition, and
- they are detected.

EXAMPLE The defined set of faults could include stuck-at faults, stuck-open, short circuits within and between components and directed short circuits.

References:

Dependability of Critical Computer Systems 1. F. J. Redmill, Elsevier Applied Science, 1988, ISBN 1-85166-203-0.

Elektronik in der Sicherheitstechnik. H. Jürs, D. Reinert, Sicherheitstechnisches Informations- und Arbeitsblatt 330220, BIA-Handbuch, Erich-Schmidt Verlag, Bielefeld, 1993.

A.2.5 Monitored redundancy

NOTE This technique/measure is referenced in table A.3 of IEC 61508-2.

Aim: To detect failure, by providing several functional units, by monitoring the behaviour of each of these to detect failures, and by initiating a transition to a safe condition if any discrepancy in behaviour is detected.

Description: The safety function is executed by at least two hardware channels. The outputs of these channels are monitored and a safe condition is initiated if a fault is detected (i.e. if the output signals from all channels are not identical).

References:

Dependability of Critical Computer Systems 1. F. J. Redmill, Elsevier Applied Science, 1988, ISBN 1-85166-203-0.

Elektronik in der Sicherheitstechnik. H. Jürs, D. Reinert, Sicherheitstechnisches Informations- und Arbeitsblatt 330220, BIA-Handbuch, Erich-Schmidt Verlag, Bielefeld, 1993.

A.2.6 Electrical/electronic components with automatic check

NOTE This technique/measure is referenced in table A.3 of IEC 61508-2.

Aim: To detect faults by periodic checking of the safety functions.

Description: The hardware is tested before starting the process, and is tested repeatedly at suitable intervals. The EUC continues to operate only if each test is successful.

References:

Dependability of Critical Computer Systems 1. F. J. Redmill, Elsevier Applied Science, 1988, ISBN 1-85166-203-0.

Elektronik in der Sicherheitstechnik. H. Jürs, D. Reinert, Sicherheitstechnisches Informations- und Arbeitsblatt 330220, BIA-Handbuch, Erich-Schmidt Verlag, Bielefeld, 1993.

A.2.7 Analogue signal monitoring

NOTE This technique/measure is referenced in tables A.3 and A.14 of IEC 61508-2.

Aim: To improve confidence in measured signals.

Description: Wherever there is a choice, analogue signals are used in preference to digital on/off states. For example, trip or safe states are represented by analogue signal levels, usually with signal level tolerance monitoring. The technique provides continuity monitoring and a higher level of confidence in the transmitter, reducing the necessary proof-test frequency of the transmitter sensing function. External interfaces, for example impulse lines, will also require testing.

Reference: UKOOA Guidelines for Instrument-Based Systems, UK Offshore Operators Association Limited, December 1995.

A.2.8 De-rating

Aim: To increase the reliability of hardware components.

Description: Hardware components are operated at levels which are guaranteed by the design of the system to be well below the maximum specification ratings. De-rating is the practice of ensuring that under all normal operating circumstances, components are operated well below their maximum stress levels.

A.3 Processing units

Global objective: To recognise failures which lead to incorrect results in processing units.

A.3.1 Self-test by software: limited number of patterns (one-channel)

NOTE This technique/measure is referenced in table A.4 of IEC 61508-2.

Aim: To detect, as early as possible, failures in the processing unit.

Description: The hardware is built using standard techniques which do not take any special safety requirements into account. The failure detection is realised entirely by additional software functions which perform self-tests using at least two complementary data patterns (for example 55hex and AAhex).

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.3.2 Self-test by software: walking bit (one-channel)

NOTE This technique/measure is referenced in table A.4 of IEC 61508-2.

Aim: To detect, as early as possible, failures in the physical storage (for example registers) and instruction decoder of the processing unit.

Description: The failure detection is realised entirely by additional software functions which perform self-tests using a data pattern (for example walking-bit pattern) which tests the physical storage (data and address registers) and the instruction decoder. However, the diagnostic coverage is only 90 %.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.3.3 Self-test supported by hardware (one-channel)

NOTE This technique/measure is referenced in table A.4 of IEC 61508-2.

Aim: To detect, as early as possible, failures in the processing unit, using special hardware that increases the speed and extends the scope of failure detection.

Description: Additional special hardware facilities support self-test functions to detect failure. For example, this could be a hardware unit which cyclically monitors the output of a certain bit pattern according to the watch-dog principle.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.3.4 Coded processing (one-channel)

NOTE This technique/measure is referenced in table A.4 of IEC 61508-2.

Aim: To detect, as early as possible, failures in the processing unit.

Description: Processing units can be designed with special failure-recognising or failure-correcting circuit techniques. So far, these techniques have been applied only to relatively simple circuits and are not widespread; however, future developments should not be excluded.

References:

The Coded Microprocessor Certification. P. Ozello, Proc. SAFECOMP '92, 185-190, 1992.

Vital Coded Microprocessor Principles and Application for Various Transit Systems. P. Forin, IFAC Control Computers Communications in Transportation, 79-84, 1989.

Le Processeur Codé: un nouveau concept appliqué à la sécurité des systèmes de transports. Gabriel, Martin, Wartski, Revue Générale des chemins de fer, No. 6, June 1990.

A.3.5 Reciprocal comparison by software

NOTE This technique/measure is referenced in table A.4 of IEC 61508-2.

Aim: To detect, as early as possible, failures in the processing unit, by dynamic software comparison.

Description: Two processing units exchange data (including results, intermediate results and test data) reciprocally. A comparison of the data is carried out using software in each unit and detected differences lead to a failure message.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.4 Invariable memory ranges

Global objective: The detection of information modifications in the invariable memory.

A.4.1 Word-saving multi-bit redundancy (for example ROM monitoring with a modified Hamming code)

NOTE See also A.5.6 and C.3.2. This technique/measure is referenced in table A.5 of IEC 61508-2.

Aim: To detect all single-bit failures, all two-bit failures, some three-bit failures, and some all-bit failures in a 16-bit word.

Description: Every word of memory is extended by several redundant bits to produce a modified Hamming code with a Hamming distance of at least 4. Every time a word is read, checking of the redundant bits can determine whether or not a corruption has taken place. If a difference is found, a failure message is produced. The procedure can also be used to detect addressing failures, by calculating the redundant bits for the concatenation of the data word and its address.

References:

Error detecting and error correcting codes. R. W. Hamming, The Bell System Technical Journal 29 (2), 147-160, 1950.

Prüfbare und korrigierbare Codes. W. W. Peterson, München, Oldenburg, 1967.

A.4.2 Modified checksum

NOTE This technique/measure is referenced in table A.5 of IEC 61508-2.

Aim: To detect all odd-bit failures, i.e. approximately 50 % of all possible bit failures.

Description: A checksum is created by a suitable algorithm which uses all the words in a block of memory. The checksum may be stored as an additional word in ROM, or an additional word may be added to the memory block to ensure that the checksum algorithm produces a predetermined value. In a later memory test, a checksum is created again using the same algorithm, and the result is compared with the stored or defined value. If a difference is found, a failure message is produced.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.4.3 Signature of one word (8-bit)

NOTE This technique/measure is referenced in table A.5 of IEC 61508-2.

Aim: To detect all one-bit failures and all multi-bit failures within a word, as well as approximately 99,6 % of all possible bit failures.

Description: The contents of a memory block is compressed (using either hardware or software) using a cyclic redundancy check (CRC) algorithm into one memory word. A typical CRC algorithm treats the whole contents of the block as byte-serial or bit-serial data flow, on which a continued polynomial division is carried out using a polynomial generator. The remainder of the division represents the compressed memory contents – it is the "signature" of the memory – and is stored. The signature is computed once again in later tests and compared with one already stored. A failure message is produced if there is a difference.

References:

Calculating an error checking character in software. S. Vasa, Computer Design, 5, 1976.

Berechnung von Fehlererkennungswahrscheinlichkeiten bei Signaturregistern. D. Leisegang, Elektronische Rechenanlagen 24, H. 2, S. 55-61, 1982.

A.4.4 Signature of a double word (16-bit)

NOTE This technique/measure is referenced in table A.5 of IEC 61508-2.

Aim: To detect all one-bit failures and all multi-bit failures within a word, as well as approximately 99,998 % of all possible bit failures.

Description: This procedure calculates a signature using a cyclic redundancy check (CRC) algorithm, but the resulting value is at least two words in size. The extended signature is stored, recalculated and compared as in the single-word case. A failure message is produced if there is a difference between the stored and recalculated signatures.

References:

Signaturanalyse in der Datenverarbeitung. D. Leisengang, M. Wagner, Elektronik 32, H. 21, S. 67-72, 1983.

Signaturregister für selbsttestende ICs. B. Könemann, J. Mucha, G. Zwiehoff, Größtintegration/ NTG-Fachtagung Baden-Baden, S. 109-112, April 1977.

A.4.5 Block replication (for example double ROM with hardware or software comparison)

NOTE This technique/measure is referenced in table A.5 of IEC 61508-2.

Aim: To detect all bit failures.

Description: The address space is duplicated in two memories. The first memory is operated in the normal manner. The second memory contains the same information and is accessed in parallel to the first. The outputs are compared and a failure message is produced if a difference is detected. In order to detect certain kinds of bit errors, the data must be stored inversely in one of the two memories and inverted once again when read.

References:

Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

Computers can now perform vital safety functions safely. Otto Berg von Linde, Railway Gazette International, Vol. 135, No. 11, 1979.

A.5 Variable memory ranges

Global objective: Detecting failures during addressing, writing, storing and reading.

A.5.1 RAM test "checkerboard" or "march"

NOTE This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect predominantly static bit failures.

Description: A checker-board type pattern of 0s and 1s is written into the cells of a bit-oriented memory. The cells are then inspected in pairs to ensure that the contents are the same and correct. The address of the first cell of such a pair is variable and the address of the second cell of the pair is formed by inverting bitwise the first address. In the first run, the address range of the memory is run towards higher addresses from the variable address, and in a second run towards lower addresses. Both runs are then repeated with an inverted pre-assignment. A failure message is produced if any difference occurs.

In a RAM test "march" the cells of a bit-oriented memory are initialised by a uniform bit stream. In the first run, the cells are inspected in ascending order: each cell is checked for the correct contents and its contents are inverted. The background, which is created in the first run, is treated in a second run in descending order and in the same manner. Both first runs are repeated with an inverted pre-assignment in a third or fourth run. A failure message is produced if a difference occurs.

References:

Memory testing. W. G. Fee, LSI Testing (Tutorial at the COMPCON 77 in San Francisco), IEEE Computer Society, W. G. Fee (ed.), 81-88, 1978.

Memory testing. P. Rosenfield, Electronics and Power, H. 1, P. 26-31, 1979.

Halbleiterspeicher-Testfolgen. Th. John, E. Schaefer, Elektronikpraxis, H. 6, 18-26 and H. 7, 10-14, 1980.

A.5.2 RAM test "walkpath"

NOTE This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect static and dynamic bit failures, and cross-talk between memory cells.

Description: The memory range to be tested is initialised by a uniform bit stream. The first cell is then inverted and the remaining memory area is inspected to ensure that the background is correct. After this, the first cell is re-inverted to return it to its original value, and the whole procedure is repeated for the next cell. A second run of the "wandering bit model" is carried out with an inverse background pre-assignment. A failure message is produced if a difference occurs.

References:

Memory testing. W. G. Fee, LSI Testing (Tutorial at the COMPCON 77 in San Francisco), IEEE Computer Society, W. G. Fee (ed.), 81-88, 1978.

Techniques for testing the microprocessor family. W. Barraclough, A. Chiang, W. Sohl, Proceedings of the IEEE 64 (6), 943-950, 1976.

A.5.3 RAM test "galpat" or "transparent galpat"

NOTE This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect static bit failures and a large proportion of dynamic couplings.

Description: In the RAM test "galpat", the chosen range of memory is first initialised uniformly (i.e. all 0s or all 1s). The first memory cell to be tested is then inverted and all the remaining cells are inspected to ensure that their contents are correct. After every read access to one of the remaining cells, the inverted cell is also checked. This procedure is repeated for each cell in the chosen memory range. A second run is carried out with the opposite initialisation. Any difference produces a failure message.

The "transparent galpat" test is a variation on the above procedure: instead of initialising all cells in the chosen memory range, the existing contents are left unchanged and signatures are used to compare the contents of sets of cells. The first cell to be tested in the chosen range is selected, and the signature S1 of all remaining cells in the range is calculated and stored. The cell to be tested is then inverted and the signature S2 of all the remaining cells is recalculated. (After every read access to one of the remaining cells, the inverted cell is also checked.) S2 is compared with S1, and any difference produces a failure message. The cell under test is re-inverted to re-establish the original contents, and the signature S3 of all the remaining cells is recalculated and compared with S1. Any difference produces a failure message. All memory cells in the chosen range are tested in the same manner.

References:

Entwurf von Selbsttestprogrammen für Mikrocomputer. E. Maehle, Microcomputing. Berichte der Tagung III/79 des German Chapter of the ACM, W. Remmele, H. Schecher, (ed.), Stuttgart, Teubner, 204-216, 1979.

Periodischer Selbsttest einer mikroprocessorgesteuerten Sicherheitsschaltung. U. Stinnesbek, Diplomarbeit am Institut für theoretische Elektrotechnik der RWTH Aachen 1980.

A.5.4 RAM test "Abraham"

NOTE This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect all stuck-at and coupling failures between memory cells.

Description: The proportion of faults detected exceeds that of the RAM test "galpat". The number of operations required to perform the entire memory test is about $30n$, where n is the number of cells in the memory. The test can be made transparent for use during the operating cycle by partitioning the memory and testing each partition in different time segments.

Reference: Efficient Algorithms for Testing Semiconductor Random-Access Memories. R. Nair, S. M. Thatte, J. A. Abraham, IEEE Trans. Comput. C-27 (6), 572-576, 1978.

A.5.5 One-bit redundancy (for example RAM monitoring with a parity bit)

NOTE This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect 50 % of all possible bit failures in the memory range tested.

Description: Every word of the memory is extended by one bit (the parity bit) which completes each word to an even or odd number of logical 1s. The parity of the data word is checked each time it is read. If the wrong number of 1s is found, a failure message is produced. The choice of even or odd parity should be made such that, whichever of the zero word (nothing but 0s) and the one word (nothing but 1s) is the more unfavourable in the event of a failure, then that word is not a valid code. Parity can also be used to detect addressing failures, when the parity is calculated for the concatenation of the data word and its address.

Reference: Integrierte Digitalbausteine. K. Reiß, H. Liedl, W. Spichall, Berlin, 1970.

A.5.6 RAM monitoring with a modified Hamming code, or detection of data failures with error-detection-correction codes (EDC)

NOTE See also A.4.1 and C.3.2. This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect all odd-bit failures, all two-bit failures, some three-bit and some multi-bit failures.

Description: Every word of the memory is extended by several redundant bits to produce a modified Hamming code with a Hamming distance of at least 4. Every time a word is read, one can determine whether a corruption has taken place by checking the redundant bits. If a difference is found, a failure message is produced. The procedure can also be used to detect addressing failure, when the redundant bits are calculated for the concatenation of the data word and its address.

References:

Error detecting and error correcting codes. R. W. Hamming, The Bell System Technical Journal 29 (2), 147-160, 1950.

Prüfbare und korrigierbare Codes. W. W. Peterson, München, Oldenburg, 1967.

A.5.7 Double RAM with hardware or software comparison and read/write test

NOTE This technique/measure is referenced in table A.6 of IEC 61508-2.

Aim: To detect all bit failures.

Description: The address space is duplicated in two memories. The first memory is operated in the normal manner. The second memory contains the same information and is accessed in parallel to the first. The outputs are compared and a failure message is produced if a difference is detected. In order to detect certain kinds of bit errors, the data must be stored inversely in one of the two memories and inverted once again when read.

References:

Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

Computers can now perform vital safety functions safely. Otto Berg von Linde, Railway Gazette International, Vol. 135, No. 11, 1979.

A.6 I/O-units and interfaces (external communication)

Global objective: To detect failures in input and output units (digital, analogue, serial or parallel) and to prevent the sending of inadmissible outputs to the process.

A.6.1 Test pattern

NOTE This technique/measure is referenced in tables A.7, A.14 and A.15 of IEC 61508-2.

Aim: To detect static failures (stuck-at failures) and cross-talk.

Description: This is a dataflow-independent cyclical test of input and output units. It uses a defined test pattern to compare observations with the corresponding expected values. The test pattern information, the test pattern reception, and test pattern evaluation must all be independent of each other. The EUC should not be inadmissibly influenced by the test pattern.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.6.2 Code protection

NOTE This technique/measure is referenced in tables A.7, A.16, A.17 and A.19 of IEC 61508-2.

Aim: To detect random hardware and systematic failures in the input/output dataflow.

Description: This procedure protects the input and output information from both systematic and random hardware failures. Code protection provides dataflow-dependent failure detection of the input and output units, based on information redundancy and/or time redundancy. Typically, redundant information is superimposed on input and/or output data. This then provides a means to monitor the correct operation of the input or output circuits. Many techniques are possible, for example a carrier frequency signal may be superimposed on the output signal of a sensor. The logic unit may then check for the presence of the carrier frequency or redundant code bits may be added to an output channel to allow the monitoring of the validity of a signal passing between the logic unit and final actuator.

Reference: Standard input/output tests and monitoring procedures – Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.6.3 Multi-channel parallel output

NOTE This technique/measure is referenced in table A.7 of IEC 61508-2.

Aim: To detect random hardware failures (stuck-at failures), failures caused by external influences, timing failures, addressing failures, drift failures and transient failures.

Description: This is a dataflow-dependent multi-channel parallel output with independent outputs for the detection of random hardware failures. Failure detection is carried out via external comparators. If a failure occurs, the EUC is switched off directly. This measure is only effective if the dataflow changes during the diagnostic test interval.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.6.4 Monitored outputs

NOTE This technique/measure is referenced in table A.7 of IEC 61508-2.

Aim: To detect individual failures, failures caused by external influences, timing failures, addressing failures, drift failures (for analogue signals) and transient failures.

Description: This is a dataflow-dependent comparison of outputs with independent inputs to ensure compliance with a defined tolerance range (time, value). A detected failure cannot always be related to the defective output. This measure is only effective if the dataflow changes during the diagnostic test interval.

References:

Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

MSR-Schutzeinrichtungen. Anforderungen und Massnahmen zur gesicherten Funktion. DIN V 19251, February 1995.

A.6.5 Input comparison/voting

NOTE This technique/measure is referenced in tables A.7 and A.14 of IEC 61508-2.

Aim: To detect individual failures, failures caused by external influences, timing failures, addressing failures, drift failures (for analogue signals) and transient failures.

Description: This is a dataflow-dependent comparison of independent inputs to ensure compliance with a defined tolerance range (time, value). There will be 1 out of 2, 2 out of 3 or better redundancy. This measure is only effective if the dataflow changes during the diagnostic test interval.

References:

Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

MSR-Schutzeinrichtungen. Anforderungen und Massnahmen zur gesicherten Funktion. DIN V 19251, February 1995.

A.7 Data paths (internal communication)

Global objective: To detect failures caused by a defect in the information transfer.

A.7.1 One-bit hardware redundancy

NOTE This technique/measure is referenced in table A.8 of IEC 61508-2.

Aim: To detect all odd-bit failures, i.e. 50 % of all the possible bit failures in the data stream.

Description: The bus is extended by one line (bit) and this additional line (bit) is used to detect failures by parity checking.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.7.2 Multi-bit hardware redundancy

NOTE This technique/measure is referenced in table A.8 of IEC 61508-2.

Aim: To detect failures during the communication on the bus and in serial transmission links.

Description: The bus is extended by two or more lines (bits) and these additional lines (bits) are used in order to detect failures by Hamming code techniques.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.7.3 Complete hardware redundancy

NOTE This technique/measure is referenced in table A.8 of IEC 61508-2.

Aim: To detect failures during the communication by comparing the signals on two buses.

Description: The bus is doubled and the additional lines (bits) are used in order to detect failures.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.7.4 Inspection using test patterns

NOTE This technique/measure is referenced in table A.8 of IEC 61508-2.

Aim: To detect static failures (stuck-at failure) and cross-talk.

Description: This is a dataflow-independent cyclical test of data paths. It uses a defined test pattern to compare observations with the corresponding expected values.

The test pattern information, the test pattern reception, and test pattern evaluation must all be independent of each other. The EUC should not be inadmissibly influenced by the test pattern.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.7.5 Transmission redundancy

NOTE This technique/measure is referenced in table A.8 of IEC 61508-2.

Aim: To detect transient failures in bus communication.

Description: The information is transferred several times in sequence. The repetition is effective only against transient failures.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.7.6 Information redundancy

NOTE This technique/measure is referenced in table A.8 of IEC 61508-2.

Aim: To detect failures in bus communication.

Description: Data is transmitted in blocks, together with a calculated checksum for each block. The receiver then re-calculates the checksum of the received data and compares the result with the received checksum.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.8 Power supply

Global objective: To detect or tolerate failures caused by a defect in the power supply.

A.8.1 Overvoltage protection with safety shut-off

NOTE This technique/measure is referenced in table A.9 of IEC 61508-2.

Aim: To protect the safety-related system against overvoltage.

Description: Overvoltage is detected early enough that all outputs can be switched to a safe condition by the power-down routine or there is a switch-over to a second power unit.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

A.8.2 Voltage control (secondary)

NOTE This technique/measure is referenced in table A.9 of IEC 61508-2.

Aim: To monitor the secondary voltages and initiate a safe condition if the voltage is not in its specified range.

Description: The secondary voltage is monitored and a power-down is initiated, or there is a switch-over to a second power unit, if it is not in its specified range.

A.8.3 Power-down with safety shut-off

NOTE This technique/measure is referenced in table A.9 of IEC 61508-2.

Aim: To shut off the power with all safety critical information stored.

Description: Overvoltage or undervoltage is detected early enough so that the internal state can be saved in non-volatile memory (if necessary), and so that all outputs can be set to a safe condition by the power-down routine, or that all outputs can be switched to a safe condition by the power-down routine, or there is a switch-over to a second power unit.

A.9 Temporal and logical program sequence monitoring

NOTE This group of techniques and measures is referenced in tables A.16, A.17 and A.19 of IEC 61508-2.

Global objective: To detect a defective program sequence. A defective program sequence exists if the individual elements of a program (for example software modules, subprograms or commands) are processed in the wrong sequence or period of time, or if the clock of the processor is faulty.

A.9.1 Watch-dog with separate time base without time-window

NOTE This technique/measure is referenced in tables A.10 and A.12 of IEC 61508-2.

Aim: To monitor the behaviour and the plausibility of the program sequence.

Description: External timing elements with a separate time base (for example watch-dog timers) are periodically triggered to monitor the computer's behaviour and the plausibility of the program sequence. It is important that the triggering points are correctly placed in the program. The watch-dog is not triggered at a fixed period, but a maximum interval is specified.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.9.2 Watch-dog with separate time base and time-window

NOTE This technique/measure is referenced in tables A.10 and A.12 of IEC 61508-2.

Aim: To monitor the behaviour and the plausibility of the program sequence.

Description: External timing elements with a separate time base (for example watch-dog timers) are periodically triggered to monitor the computer's behaviour and the plausibility of the program sequence. It is important that the triggering points are correctly placed in the program. A lower and upper limit is given for the watch-dog timer. If the program sequence takes a longer or shorter time than expected, emergency action is taken.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.9.3 Logical monitoring of program sequence

NOTE This technique/measure is referenced in tables A.10 and A.12 of IEC 61508-2.

Aim: To monitor the correct sequence of the individual program sections.

Description: The correct sequence of the individual program sections is monitored using software (counting procedure, key procedure) or using external monitoring facilities. It is important that the checking points are placed in the program correctly.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.9.4 Combination of temporal and logical monitoring of program sequences

NOTE This technique/measure is referenced in tables A.10 and A.12 of IEC 61508-2.

Aim: To monitor the behaviour and the correct sequence of the individual program sections.

Description: A temporal facility (for example a watch-dog timer) monitoring the program sequence is retriggered only if the sequence of the program sections is also executed correctly.

Reference: Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

A.9.5 Temporal monitoring with on-line check

NOTE This technique/measure is referenced in tables A.10 and A.12 of IEC 61508-2.

Aim: To detect faults in the temporal monitoring.

Description: The temporal monitoring is checked at start-up, and a start is only possible if the temporal monitoring operates correctly. For example, a heat sensor could be checked by a heated resistor at start-up.

A.10 Ventilation and heating

NOTE This group of techniques and measures is referenced in tables A.17 and A.19 of IEC 61508-2.

Global objective: To control failures in the ventilation or heating, and/or their monitoring, if this is safety-related.

A.10.1 Temperature sensor

NOTE This technique/measure is referenced in table A.11 of IEC 61508-2.

Aim: To detect over- or under-temperature before the system begins to operate outside specification.

Description: A temperature sensor monitors temperature at the most critical points of the E/E/PES. Before the temperature leaves the specified range, emergency action is taken.

A.10.2 Fan control

NOTE This technique/measure is referenced in table A.11 of IEC 61508-2.

Aim: To detect incorrect operation of the fans.

Description: The fans are monitored for correct operation. If a fan is not working properly, maintenance (or ultimately, emergency) action is taken.

A.10.3 Actuation of the safety shut-off via thermal fuse

NOTE This technique/measure is referenced in table A.11 of IEC 61508-2.

Aim: To shut off the safety-related system before the system works outside of its thermal specification.

Description: A thermal fuse is used to shut off the safety-related system. For a PES, the shut-off is introduced by a power-down routine which stores all information necessary for emergency action.

A.10.4 Staggered message from thermo-sensors and conditional alarm

NOTE This technique/measure is referenced in table A.11 of IEC 61508-2.

Aim: To indicate that the safety-related system is working outside its thermal specification.

Description: The temperature is monitored and an alarm is raised if the temperature is outside of a specified range.

A.10.5 Connection of forced-air cooling and status indication

NOTE This technique/measure is referenced in table A.11 of IEC 61508-2.

Aim: To prevent overheating by forced-air cooling.

Description: The temperature is monitored and forced-air cooling is introduced if the temperature is higher than a specified limit. The user is informed of the status.

A.11 Communication and mass-storage

Global objective: To control failures during communication with external sources and mass-storage.

A.11.1 Separation of electrical energy lines from information lines

NOTE This technique/measure is referenced in table A.13 of IEC 61508-2.

Aim: To minimise cross-talk induced by high currents in the information lines.

Description: Electrical energy supply lines are separated from the lines carrying the information. The electrical field which could induce voltage spikes on the information lines decreases with distance.

A.11.2 Spatial separation of multiple lines

NOTE This technique/measure is referenced in tables A.13 and A.17 of IEC 61508-2.

Aim: To minimise cross-talk induced by high currents in multiple lines.

Description: Lines carrying duplicated signals are separated from each other. The electrical field which could induce voltage spikes on the multiple lines decreases with the distance. This measure also reduces common cause failures.

A.11.3 Increase of interference immunity

NOTE This technique/measure is referenced in tables A.13, A.17 and A.19 of IEC 61508-2.

Aim: To minimise electromagnetic interference on the safety-related system.

Description: Design techniques such as shielding and filtering are used to increase the interference immunity of the safety-related system to electromagnetic disturbances which may be radiated or conducted on power or signal lines, or result from electrostatic discharges.

References:

IEC 61000-5-2/TR3:1997, *Electromagnetic compatibility (EMC) – Part 5: Installation and mitigation guidelines – Section 2: Earthing and cabling*.

Noise Reduction Techniques in Electronic Systems. H. W. Ott, John Wiley Interscience, 2nd Edition, 1988.

EMC for Product Designers. Tim Williams, Newnes, 1992, ISBN 0-7506-1264-9.

Grounding and Shielding Techniques in Instrumentation. John Wiley & Sons, New York, 1986.

Principles and Techniques of Electromagnetic Compatibility. C. Christopoulos, CRC Press, 1995.

Gestaltung von Maschinensteuerungen unter Berücksichtigung der elektromagnetischen Verträglichkeit. F. Börner, Sicherheitstechnisches Informations- und Arbeitsblatt 330260, BIA-Handbuch. 20. Lfg. V/93, Erich Schmidt Verlag, Bielefeld.

A.11.4 Antivalent signal transmission

NOTE This technique/measure is referenced in tables A.13 and A.17 of IEC 61508-2.

Aim: To detect the same induced voltages in multiple signal transmission lines.

Description: All duplicated information is transmitted with antivalent signals (for example logic 1 and 0). Common cause failures (for example by electromagnetic emission) can be detected by an antivalent comparator.

Reference: Elektronik in der Sicherheitstechnik. H. Jürs, D. Reinert, Sicherheitstechnisches Informations- und Arbeitsblatt 330220, BIA-Handbuch. 20. Lfg. V/93, Erich Schmidt Verlag, Bielefeld.

A.12 Sensors

Global objective: To control failures in the sensors of the safety-related system.

A.12.1 Reference sensor

NOTE This technique/measure is referenced in table A.14 of IEC 61508-2.

Aim: To detect the incorrect operation of a sensor.

Description: An independent reference sensor is used to monitor the operation of a process sensor. All input signals are checked at suitable time intervals by the reference sensor to detect failures of the process sensor.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

A.12.2 Positive-activated switch

NOTE This technique/measure is referenced in table A.14 of IEC 61508-2.

Aim: To open a contact by a direct mechanical connection between switch cam and contact.

Description: A positive-activated switch opens its normally closed contacts by a direct mechanical connection between switch cam and contact. This ensures that whenever the switch cam is in the operated position, the switch contacts must be open.

Reference: Verriegelung beweglicher Schutzeinrichtungen. F. Kreutzkamp, K. Becker, Sicherheitstechnisches Informations- und Arbeitsblatt 330210, BIA-Handbuch. 1. Lfg. IX/85, Erich Schmidt Verlag, Bielefeld.

A.13 Final elements (actuators)

Global objective: To control failures in the final elements in the safety-related system.

A.13.1 Monitoring

NOTE This technique/measure is referenced in table A.15 of IEC 61508-2.

Aim: To detect the incorrect operation of an actuator.

Description: The operation of the actuator is monitored (for example by the positively activated contacts of a relay, see monitoring of relay contacts in A.1.2). The redundancy introduced by this monitoring can be used to trigger emergency action.

References:

Zusammenstellung und Bewertung elektromechanischer Sicherheitsschaltungen für Verriegelungseinrichtungen. F. Kreutzkamp, W. Hertel, Sicherheitstechnisches Informations- und Arbeitsblatt 330212, BIA-Handbuch. 17. Lfg. X/91, Erich Schmidt Verlag, Bielefeld.

Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

A.13.2 Cross-monitoring of multiple actuators

NOTE This technique/measure is referenced in table A.15 of IEC 61508-2.

Aim: To detect faults in actuators by comparing the results.

Description: Each multiple actuator is monitored by a different hardware channel. If a discrepancy occurs, emergency action is taken.

A.14 Measures against the physical environment

NOTE This technique/measure is referenced in table A.17 of IEC 61508-2

Aim: To prevent influences of the physical environment (water, dust, corrosive substances) causing failures.

Description: The enclosure of the equipment is designed to withstand the expected environment.

Reference: IEC 60529:1989, *Degrees of protection provided by enclosures (IP Code)*.

Annex B (informative)

Overview of techniques and measures for E/E/PES: avoidance of systematic failures (see IEC 61508-2 and IEC 61508-3)

NOTE Many techniques in this annex are applicable to software but have not been duplicated in annex C.

B.1 General measures and techniques

B.1.1 Project management

NOTE This technique/measure is referenced in tables B.1 to B.6 of IEC 61508-2.

Aim: To avoid failures by adoption of an organisational model and rules and measures for development and testing of safety-related systems.

Description: The most important and best measures are

- the creation of an organisational model, especially for quality assurance (see standards such as the series ISO 9000-1 to ISO 9004-1 or similar) which is set down in a quality assurance handbook; and
- the establishment of regulations and measures for the creation and validation of safety-related systems in cross-project and project-specific guidelines.

A number of important basic principles are set down in the following:

- definition of a design organisation:
 - tasks and responsibilities of the organisational units,
 - authority of the quality assurance departments,
 - independence of quality assurance (internal inspection) from development;
- definition of a sequence plan (activity models):
 - determination of all activities which are relevant during execution of the project including internal inspections and their scheduling,
 - project update;
- definition of a standardised sequence for an internal inspection:
 - planning, execution and checking of the inspection (inspection theory),
 - releasing mechanisms for subproducts,
 - the safekeeping of repeat inspections;
- configuration management:
 - administration and checking of versions,
 - detection of the effects of modifications,
 - consistency inspections after modifications;
- introduction of a quantitative assessment of quality assurance measures:
 - requirement acquisition,
 - failure statistics;
- introduction of computer-aided universal methods, tools and training of personnel.

References:

IEEE: Software Engineering Standards. IEEE/Wiley-Interscience, New York, 1987.

Dependability of Critical Computer Systems 1. F. J. Redmill, Elsevier Applied Science, 1988, ISBN 1-85166-203-0.

Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

B.1.2 Documentation

NOTE 1 This technique/measure is referenced in tables B.1 to B.6 of IEC 61508-2.

NOTE 2 See also clause 5 and annex A of IEC 61508-1.

Aim: To avoid failures and facilitate assessment of system safety, by documenting each step during development.

Description: The operational capacity and safety, as well as the care taken in development by all parties involved, has to be demonstrated during assessment. In order to be able to show the development care, and in order to guarantee the verification of the evidence of safety at any time, special importance is given to the documentation.

Important common measures are the introduction of guidelines and computer aid, i.e.

- guidelines, which
 - specify a grouping plan;
 - ask for checklists for the contents; and
 - determine the form of the document;
- administration of the documentation with the help of a computer-aided and organised project library.

Individual measures are:

- separation in the documentation
 - of the requirements,
 - of the system (user-documentation) and
 - of the development (including internal inspection);
- grouping of the development documentation according to the safety lifecycle;
- definition of standardised documentation modules, from which the documents can be compiled;
- clear identification of the constituent parts of the documentation;
- formalised revision update;
- selection of clear and intelligible means of description:
 - formal notation for determinations;
 - natural language for introductions, justifications and representations of intentions;
 - graphical representations for examples;
 - semantic definition of graphical elements; and
 - directories of specialised words.

References:

IEC 61506:1997, *Industrial-process measurement and control – Documentation of application software*.

EWICS European Workshop on Industrial Computer Systems, TC 7: Safety Related Computers – Software Development and Systems Documentation. Verlag TÜV Rheinland, Köln, 1985.

Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

Entwicklungstechnik sicherheitsverantwortlicher Software in der Eisenbahn-Signaltechnik. U. Feucht, Informatik-Fachberichte 86, Springer Verlag, Berlin, 184-195, 1984.

Richtlinie zur Erstellung und Prüfung sicherheitsrelevanter Software. K. Grimm, G. Heiner, Informatik Fachberichte 86, Springer Verlag, Berlin, 277-288, 1984.

B.1.3 Separation of safety-related systems from non-safety-related systems

NOTE This technique/measure is referenced in tables B.1 and B.6 of IEC 61508-2.

Aim: To prevent the non-safety-related part of the system from influencing the safety-related part in undesired ways.

Description: In the specification it should be decided whether a separation of the safety-related systems and non-safety-related systems is possible. Clear specifications should be written for the interfacing of the two parts. A clear separation reduces the effort for testing the safety-related systems.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

B.1.4 Diverse hardware

NOTE This technique/measure is referenced in tables A.16, A.17 and A.19 of IEC 61508-2.

Aim: To detect systematic failures during operation of the EUC, using diverse components with different rates and types of failures.

Description: Different types of components are used for the diverse channels of a safety-related system. This reduces the probability of common cause failures (for example overvoltage, electromagnetic interference), and increases the probability of detecting such failures.

Existence of different means of performing a required function, for example different physical principles, offer other ways of solving the same problem. There are several types of diversity. Functional diversity employs the use of different approaches to achieve the same result.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

B.2 E/E/PES safety requirements specification

Global objective: To produce a specification which is, as far as possible, complete, free from mistakes, free from contradiction, and simple to verify.

B.2.1 Structured specification

NOTE This technique/measure is referenced in tables B.1 and B.6 of IEC 61508-2.

Aim: To reduce complexity by creating a hierarchical structure of partial requirements. To avoid interface failures between the requirements.

Description: This technique structures the functional specification into partial requirements such that the simplest possible, visible relations exist between the latter. This analysis is successively refined until small clear partial requirements can be distinguished. The result of the final refinement is a hierarchical structure of partial requirements which provide a framework for the specification of the complete requirements. This method emphasises the interfaces of the partial requirements and is particularly effective for avoiding interface failures.

References:

Structured Analysis and System Specification. T. De Marco, Yourdon Press, Englewood Cliffs, 1979.

ESA PSS 05-02, Guide to the user requirements definition phase, European Space Agency, 1989.

B.2.2 Formal methods

NOTE 1 See C.2.4 for details of specific formal methods.

NOTE 2 This technique/measure is referenced in tables B.1 and B.6 of IEC 61508-2.

Aim: To express a specification unambiguously and consistently, so that mistakes and omissions can be detected.

Description: Formal methods provide a means of developing a description of a system at some stage in its specification or design. The resulting description takes a mathematical form and can be subjected to mathematical analysis to detect various classes of inconsistency or incorrectness. Moreover, the description can in some cases be analysed by a machine with a rigour similar to the syntax checking of a source program by a compiler, or animated to display various aspects of the behaviour of the system described. Animation can give extra confidence that the system meets the real requirement as well as the formally specified requirement, because it improves human recognition of the specified behaviour.

A formal method will generally offer a notation (generally some form of discrete mathematics being used), a technique for deriving a description in that notation, and various forms of analysis for checking a description for different correctness properties.

Starting from a mathematically formal specification, the design can be transformed by a series of step-wise refinements to a logic circuit design.

References:

Dependability of Critical Computer Systems 3. P. G. Bishop et al, Elsevier Applied Science, 1990, ISBN 1-85166-544-7.

HOL: A Machine Orientated Formulation of Higher Order Logic. M. Gordon, University of Cambridge Technical Report Number 68, 1985.

B.2.3 Semi-formal methods

Aim: To express parts of a specification unambiguously and consistently, so that some mistakes and omissions can be detected.

NOTE This technique/measure is referenced in tables B.1, B.2 and B.6 of IEC 61508-2 and in tables A.1, A.2 and A.4 of IEC 61508-3.

B.2.3.1 General

Aim: To prove that the design meets its specification.

Description: Semi-formal methods provide a means of developing a description of a system at some stage in its development, i.e. specification, design or coding. The description can in some cases be analysed by machine or animated to display various aspects of the system behaviour. Animation can give extra confidence that the system meets the real requirement as well as the specified requirement.

Two semi-formal methods are described in the following subclauses.

B.2.3.2 Finite state machines/state transition diagrams

NOTE This technique/measure is referenced in tables B.5 and B.7 of IEC 61508-3.

Aim: To model, specify or implement the control structure of a system.

Description: Many systems can be described in terms of their states, their inputs, and their actions. Thus when in state S1, on receiving input I a system might carry out action A and move to state S2. By describing a system's actions for every input in every state we can describe a system completely. The resulting model of the system is called a finite state machine. It is often drawn as a so-called state transition diagram showing how the system moves from one state to another, or as a matrix in which the dimensions are state and input, and the matrix cells contain the action and new state resulting from receiving the input when in the given state.

Where a system is complicated or has a natural structure this can be reflected in a layered finite state machine.

A specification or design expressed as a finite state machine can be checked for

- completeness (the system must have an action and new state for every input in every state);
- consistency (only one state change is described for each state/input pair); and
- reachability (whether or not it is possible to get from one state to another by any sequence of inputs).

These are important properties for critical systems. Tools to support these checks are easily developed. Algorithms also exist that allow the automatic generation of test cases for verifying a finite state machine implementation or for animating a finite state machine model.

Reference: Introduction to the theory of Finite State Machines. A. Gill, McGraw-Hill, 1962.

B.2.3.3 Time Petri nets

NOTE This technique/measure is referenced in tables B.5 and B.7 of IEC 61508-3.

Aim: To model relevant aspects of the system behaviour and to assess and possibly improve safety and operational requirements through analysis and re-design.

Description: Petri nets belong to a class of graph theoretic models which are suitable for representing information and control flow in systems that exhibit concurrency and have asynchronous behaviour.

A Petri net is a network of places and transitions. The places may be "marked" or "unmarked". A transition is "enabled" when all the input places to it are marked. When enabled, it is permitted (but not obliged) to "fire". If it fires, the input places to the transition become unmarked, and each output place from the transition is marked instead.

The potential hazards can be represented as particular states (markings) in the model. The Petri net model can be extended to allow for timing features of the system. Although "classical" Petri nets concentrate on control flow aspects, several extensions have been proposed to incorporate data flow into the model.

References:

Petri nets: Properties, analysis and applications. T. Murata, Proc. IEEE 77 (4), 541-580, April 1989.

Safety analysis using Petri nets. N. G. Leveson, J. L. Stolzy, Proc. 15th Ann. Int. Symp on Fault-Tolerant Computing, 358-363, IEEE, 1985.

Using Petri nets for safety analysis of unmanned Metro system. M. El Koursi, P. Ozello, Proc. SAFECOMP '92, 135-139, Pergamon Press, 1992.

Net theory and applications. W. Brauer (ed.), Lecture Notes in Computer Science, Vol. 84, Springer Verlag, 1980.

Petri net theory and modelling of systems. J. L. Peterson, Prentice Hall, 1981.

A tool for requirements specification and analysis of real time software based on timed Petri nets. S. Bologna, F. Pisacane, C. Ghezzi, D. Mandrioli, Proc. SAFECOMP 88, 9-11 November 1988. Fulda, Fed. Rep. of Germany, 1988.

B.2.4 Computer-aided specification tools

NOTE This technique/measure is referenced in tables B.1 and B.6 of IEC 61508-2 and in tables A.1 and A.2 of IEC 61508-3.

B.2.4.1 General

Aim: To use formal specification techniques to facilitate automatic detection of ambiguity and completeness.

Description: The technique produces a specification in the form of a database that can be automatically inspected to assess consistency and completeness. The specification tool can animate various aspects of the specified system to the user. In general, the technique supports not only the creation of the specification but also of the design and other phases of the project lifecycle. Specification tools can be classified according to the following subclauses.

B.2.4.2 Tools oriented towards no specific method

Aim: To help the user write a good specification by providing prompts and links between related parts.

Description: The specification tool takes over some routine work from the user and supports the project management. It does not enforce any particular specification methodology. The relative independence with regard to method allows users a great deal of freedom but also gives them little of the specialised support necessary when creating specifications. This makes familiarisation with the system more difficult.

Reference: Integrierte Rechnerunterstützung für Entwicklung, Projektmanagement und Produktverwaltung mit EPOS. R. Lauber, P. Lempp, Elektron. Rechenanlagen 27, Heft 2, 68-74, 1985.

B.2.4.3 Model orientated procedure with hierarchical analysis

Aim: To help the user write a good specification by ensuring consistency between descriptions of actions and data at various levels of abstraction.

Description: This method gives a functional representation of the desired system (structured analysis) at various levels of abstraction (degree of precision). The analysis at various levels acts on both actions and data. Assessment of ambiguity and completeness is possible between hierarchical levels as well as between two functional units (modules) on the same level.

Reference: Structured Analysis for Requirement Definition. D. T. Ross, K. E. Schomann jr, IEEE Trans. on SE, January 1977.

B.2.4.4 Entity models

Aim: To help the user write a good specification by focusing on entities within the system and relationships between them.

Description: The desired system is described as a collection of objects and their relationships. The tool enables one to determine which relationships can be interpreted by the system. In general, the relationships permit a description of the hierarchical structure of the objects, the data flow, the relationships between the data, and which data are subject to certain manufacturing processes. The classical procedure has been extended for process control applications. Inspection capabilities and support for the user depend on the variety of relationships illustrated. On the other hand, a large number of representation possibilities makes the application of this technique complex.

References:

PSL/PSA Computer-aided Technique for Structured Documentation and Analysis of Information Processing. D. Teichroew, E. A. Hershey, IEEE Trans on SE, Jan 1977.

Computer Aided Software Development. D. Teichroew, E. A. Hershey, Y. Lamamoto, Beitrag in: Verfahren und Hilfsmittel für Spezifikation und Entwurf von Prozeßautomatisierungssystemen. Hommel (ed.), Bericht KfK-PDV 154, Kernforschungszentrum Karlsruhe, 1978.

PCSL und ESPRESO – zwei Ansätze zur Formalisierung der Prozessrechner Software-spezifikation. J. Ludewig, GI-Fachtagung Prozessrechner 1981, Informatik-Fachberichte Bd. 39, Springer Verlag, Berlin, 1981.

B.2.4.5 Incentive and answer

Aim: To help the user write a good specification by identifying stimulus-response relationships.

Description: The relationships between the objects of the system are specified in a notation of "incentives" and "answers". A simple and easily expanded language is used which contains language elements which represent objects, relationships, characteristics and structures.

References:

A Requirements Engineering Methodology for Real-time Processing Requirements. M. W. Alford, IEEE Trans on SE, January 1977.

The Specification System X-SPEX – Introduction and Experiences. G. Dahll, J. Lahti, Proc. SAFECOMP '83, 111-118.

B.2.5 Checklists

NOTE This technique/measure is referenced in tables B.1, B.2 and B.6 of IEC 61508-2 and in tables A.10 and B.8 of IEC 61508-3.

Aim: To draw attention to, and manage critical appraisal of, all important aspects of the system by safety lifecycle phase, ensuring comprehensive coverage without laying down exact requirements.

Description: A set of questions to be answered by the person performing the checklist. Many of the questions are of a general nature and the assessor must interpret them as seems most appropriate to the particular system being assessed. Checklists can be used for all phases of the overall, E/E/PES and software safety lifecycles and are particularly useful as a tool to aid the functional safety assessment.

To accommodate wide variations in systems being validated, most checklists contain questions which are applicable to many types of system. As a result, there may well be questions in the checklist being used which are not relevant to the system being dealt with and which should be ignored. Equally there may be a need, for a particular system, to supplement the standard checklist with questions specifically directed at the system being dealt with.

In any case it should be clear that the use of checklists depends critically on the expertise and judgement of the engineer selecting and applying the checklist. As a result, the decisions taken by the engineer, with regard to the checklist(s) selected, and any additional or superfluous questions, should be fully documented and justified. The objective is to ensure that the application of the checklists can be reviewed and that the same results will be achieved unless different criteria are used.

The object in completing a checklist is to be as concise as possible. When extensive justification is necessary this should be done by reference to additional documentation. Pass, fail and inconclusive, or some similar restricted set of responses should be used to document the results for each question. This conciseness greatly simplifies the procedure of reaching an overall conclusion as to the results of the checklist assessment.

References:

IEC 61346 (all parts), *Industrial systems, installations and equipment and industrial products – Structuring principles and reference designation*.

IEC 60880:1986, *Software for computers in the safety systems of nuclear power stations*.

Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

Programmable Electronic Systems (PES) in Safety Related Application. Health and Safety Executive, UK, 1987.

Dependability of Critical Computer Systems 2, F. J. Redmill, Elsevier Applied Science, 1989, ISBN 1-85166-381-9.

The Art of Software Testing. G. Myers, Wiley & Sons, New York, 1979.

B.2.6 Inspection of the specification

NOTE This technique/measure is referenced in tables B.1 and B.6 of IEC 61508-2.

Aim: To avoid incompleteness and contradiction in the specification.

Description: Inspection is a general technique in which various qualities of a specification document are assessed by an independent team. The inspection team puts questions to the creator, who must answer them satisfactorily. The examination should (if possible) be carried out by a team that was not involved in the creation of the specification. The required degree of independence is determined by the safety integrity levels demanded of the system. The independent inspectors should be able to reconstruct the operational function of the system in an indisputable manner without referring to any further specifications. They must also check that all relevant safety and technical aspects in the operational and organisational measures are covered. This procedure has proved itself to be very effective in practice.

References:

The Art of Software Testing. G. Myers, Wiley & Sons, New York, 1979.

IEC 61160:1992, *Formal design review*.

B.3 E/E/PES design and development

Global objective: To produce a stable design of the safety-related system in conformance with the specification.

B.3.1 Observance of guidelines and standards

NOTE This technique/measure is referenced in table B.2 of IEC 61508-2.

Aim: To observe application sector standards (not specified in this standard).

Description: Guidelines should be complied with during the design of the safety-related system. These guidelines should firstly lead to safety-related systems which are practically free from failures, and secondly facilitate the subsequent safety validation. They can be universally valid, specific to a project, or specific only to a single phase.

References:

EWICS European Workshop on Industrial Computer Systems, TC 7: Safety Related Computers – Software Development and Systems Documentation. Verlag TÜV Rheinland, Köln, 1985.

Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

Deutsche Bundesbahn: Richtlinien-Entwürfe 42500 to 42550 für das Handbuch "Grundsätze zur technischen Zulassung in der Signal- und Nachrichtentechnik". Bundesbahn-Zentralamt München, August 1987.

Richtlinie zur Erstellung und Prüfung sicherheitsrelevanter Software. K. Grimm, G. Heiner, Informatik Fachberichte 86, Springer Verlag, Berlin, 277-288, 1984.

B.3.2 Structured design

NOTE This technique/measure is referenced in tables B.2 and B.6 of IEC 61508-2.

Aim: To reduce complexity by creating a hierarchical structure of partial requirements. To avoid interface failures between the requirements. To simplify verification.

Description: When designing the hardware, specific criteria or methods should be used. For example, the following might be required:

- a hierarchically structured circuit design;
- use of manufactured and tested circuit parts.

Similarly, when designing the software, the use of structure charts enables an unambiguous structure of the software modules to be created. This structure shows how the modules relate to each other, the precise data which passes between modules, and the precise controls that exist between modules.

References:

Structured Development for Real-Time Systems (3 Volumes). P. T. Yourdon, P. T. Yourdon Press, 1985.

Software Design for Real-time Systems. J. E. Cooling, Chapman and Hall, 1991.

Essential Systems Analysis. St. M. McMenamin, F. Palmer, Yourdon Inc, 1984.

The Use of Structured Methods in the Development of Large Software-Based Avionic Systems. D. J. Hatley, Proceedings DASC, Baltimore, 1984.

An Overview of JSD, J. R. Cameron, IEEE Trans SE-12 No. 2, February 1986.

System Development. M. Jackson, Prentice-Hall, 1983.

MASCOT 3 User Guide. MASCOT Users Forum, RSRE, Malvern, England, 1987.

Structured Development for Real-Time Systems (3 Volumes). P. T. Ward, S. J. Mellor, Yourdon Press, 1985.

Structured Analysis for Requirements Definition, D. T. Ross, K. E. Schoman Jr, IEEE Trans. Software Eng, Vol. SE-3, 6-15, 1977.

Structured Analysis (SA): A language for communicating ideas. D. T. Ross, IEEE Trans. Software Eng, Vol. SE-3 (1), 16-34.

Applications and Extensions of SADT. D. T. Ross, Computer, 25-34, April 1985.

Structured Analysis and Design Technique – Application on Safety Systems. W. Heins, Risk Assessment and Control Courseware, Module B1, chapter 11, Delft University of Technology, Safety Science Group, PO Box 5050, 2600 GB Delft, Netherlands, 1989.

IEC 61346 (all parts), *Industrial systems, installations and equipment and industrial products – Structuring principles and reference designations*.

B.3.3 Use of well-tried components

NOTE This technique/measure is referenced in tables B.2 and B.6 of IEC 61508-2.

Aim: To reduce the risk of numerous first time and undetected faults by the use of components with specific characteristics.

Description: The selection of well-tried components is carried out by the manufacturer, with regard to safety according to the reliability of the components (for example the use of operationally tested physical units to meet high safety requirements, or the storing of safety-related programs in safe memories only). The safety of memories can refer to unauthorised access as well as environmental influences (electromagnetic compatibility, radiation, etc) and the response of the components in the event of a failure occurring.

References:

Reliability Testing for Industrial use. W. T. Greenwood, Computer 10 (7), 26-30, 1977.

Independent Test Labs: Caveat Emptor. E. Rubinstein, IEEE Spectrum, 14 (6), 44-50, 1977.

Microcomputers in safety technique – an aid to orientation for developer and manufacturer. H. Hölscher, J. Rader, Verlag TÜV Rheinland, Köln, 1986, ISBN 3-88585-315-9.

IEC 61163-1:1995, *Reliability stress screening – Part 1: Repairable items manufactured in lots*.

Zuverlässigkeit elektronischer Komponenten. T. Bajenescu, VDE-Verlag, Berlin, 1985.

B.3.4 Modularisation

NOTE This technique/measure is referenced in tables B.2 and B.6 of IEC 61508-2.

Aim: To reduce complexity and avoid failures, related to interfacing between subsystems.

Description: Every subsystem, at all levels of the design, is clearly defined and is of restricted size (only a few functions). The interfaces between subsystems are kept as simple as possible and the cross-section (i.e. shared data, exchange of information) is minimised. The complexity of individual subsystems is also restricted.

References:

EWICS European Workshop on Industrial Computer Systems, TC 7: Safety Related Computers – Software Development and Systems Documentation. TÜV Rheinland, Köln, 1985.

The Art of Software Testing. G. J. Myers, Wiley & Sons, New York, 1979.

Software Reliability – Principles and Practices. G. J. Myers, Wiley-Interscience, New York, 1976.

Software Design for Real-time Systems. J. E. Cooling, Chapman and Hall, 1991.

B.3.5 Computer-aided design tools

NOTE This technique/measure is referenced in tables B.2 and B.6 of IEC 61508-2 and in table A.4 of IEC 61508-3.

Aim: To carry out the design procedure more systematically. To include appropriate automatic construction elements which are already available and tested.

Description: Computer-aided design tools (CAD) should be used during the design of both hardware and software when available and justified by the complexity of the system. The correctness of such tools should be demonstrated by specific testing, by an extensive history of satisfactory use, or by independent verification of their output for the particular safety-related system that is being designed.

References:

Verification – The Practical Problems. J. T. Webb and D. J. Mannering, SARSS 87, November 1987, Altrincham, England, Elsevier Applied Science, 1987, ISBN 1-85166-167-0.

An Experience in Design and Validation of Software for a Reactor Protection System. S. Bologna, E. de Agostino et al, IFAC Workshop, SAFECOMP 1979, Stuttgart, 16-18 May 1979, Pergamon Press, 1979.

B.3.6 Simulation

NOTE This technique/measure is referenced in tables B.2, B.5 and B.6 of IEC 61508-2.

Aim: To carry out a systematic and complete inspection of an electrical/electronic circuit, of both the functional performance and the correct dimensioning of the components.

Description: The function of the safety-related system circuit is simulated on a computer via a software behavioural model. Individual components of the circuit each have their own simulated behaviour, and the response of the circuit in which they are connected is examined by looking at the marginal data of each component.

References:

Proc. Working Conference on Prototyping. Namur, October 1983, Budde et al, Springer Verlag, 1984.

Using an executable specification language for an information system. S. Urban et al, IEEE Trans Software Engineering, Vol. SE-11 No. 7, July 1985.

Verification and validation of Real-time Software. W. J. Quirk (ed.), Springer Verlag, Berlin, 1985.

VDI-Gemeinschaftsausschuß Industrielle Systemtechnik: Software-Zuverlässigkeit. VDI-Verlag, 1993.

B.3.7 Inspection (reviews and analysis)

NOTE This technique/measure is referenced in tables B.2 and B.6 of IEC 61508-2.

Aim: To reveal discrepancies between the specification and implementation.

Description: Specified functions of the safety-related system are examined and evaluated to ensure that the safety-related system conforms to the requirements given in the specification. Any points of doubt concerning the implementation and use of the product are documented so they may be resolved. In contrast to a walk-through, the author is passive and the inspector is active during the inspection procedure.

References:

The Art of Software Testing. G. J. Myers, Wiley & Sons, New York, 1979.

Dependability of Critical Computer Systems 3. P. G. Bishop et al, Elsevier Applied Science, 1990, ISBN 1-85166-544-7.

VDI-Gemeinschaftsausschuß Industrielle Systemtechnik: Software-Zuverlässigkeit. VDI-Verlag, 1993.

ANSI/IEEE Std. 1028:1997, IEEE Standard for software reviews and audits.

B.3.8 Walk-through

NOTE This technique/measure is referenced in table B.6 of IEC 61508-2.

Aim: To reveal discrepancies between the specification and implementation.

Description: Specified functions of the safety-related system draft are examined and evaluated to ensure that the safety-related system complies with the requirements given in the specification. Doubts and potential weak points concerning the realisation and use of the product are documented so that they may be resolved. In contrast to an inspection, the author is active and the inspector is passive during the walk-through.

References:

Dependability of Critical Computer Systems 3. P. G. Bishop et al, Elsevier Applied Science, 1990, ISBN 1-85166-544-7.

Methodisches Testen von Programmen. G. J. Myers, Oldenbourg Verlag, München, Wien, 1987.

VDI-Gemeinschaftsausschuß Industrielle Systemtechnik: Software-Zuverlässigkeit. VDI-Verlag, 1993.

ANSI/IEEE Std. 1028:1997, IEEE Standard for software reviews and audits.

B.4 E/E/PES operation and maintenance procedures

Global objective: To develop procedures which help to avoid failures during the operation and maintenance of the safety-related system.

B.4.1 Operation and maintenance instructions

NOTE This technique/measure is referenced in table B.4 of IEC 61508-2.

Aim: To avoid mistakes during operation and maintenance of the safety-related system.

Description: User instructions contain essential information on how to use and how to maintain the safety-related system. In special cases, these instructions will also include examples on how to install the safety-related system in general. All instructions must be easily understood. Figures and schematics should be used to describe complex procedures and dependencies.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.

B.4.2 User friendliness

NOTE This technique/measure is referenced in table B.4 of IEC 61508-2.

Aim: To reduce complexity during operation of the safety-related system.

Description: The correct operation of the safety-related system may depend to some degree on human operation. By considering the relevant system design and the design of the workplace, the safety-related system developer must ensure that

- the need for human intervention is restricted to an absolute minimum;
- the necessary intervention is as simple as possible;
- the potential for harm from operator error is minimised;
- the intervention facilities and indication facilities are designed according to ergonomic requirements;
- the operator facilities are simple, well labelled and intuitive to use;
- the operator is not overstrained, even in extreme situations;
- training on intervention procedures and facilities is adapted to the level of knowledge and motivation of the trainee user.

B.4.3 Maintenance friendliness

NOTE This technique/measure is referenced in table B.4 of IEC 61508-2.

Aim: To simplify maintenance procedures of the safety-related system and to design the necessary means for effective diagnosis and repair.

Description: Preventive maintenance and repair is often carried out under difficult circumstances and under pressure from deadlines. Therefore, the safety-related system developer should ensure that

- safety-related maintenance measures are necessary as seldom as possible or even, ideally, not necessary at all;
- sufficient, sensible and easy-to-handle diagnosis tools are included for those repairs that are unavoidable – tools should include all necessary interfaces;
- if separate diagnosis tools have to be developed or obtained, then these should be available on time.

B.4.4 Limited operation possibilities

NOTE This technique/measure is referenced in tables B.4 and B.6 of IEC 61508-2.

Aim: To reduce the operation possibilities for the normal user.

Description: This approach reduces the operation possibilities by

- limiting the operation within special operating modes, for example by key switches;
- limiting the number of operating elements;
- limiting the number of generally possible operating modes.

Reference: Guidelines for Safe Automation of Chemical Processes. CCPS, AIChE, New York, 1993.