IEC TS 61508-3-2

Edition 1.0 2024-08

TECHNICAL SPECIFICATION

Functional safety of electrical/electronic/programmable electronic safety-related systems –
Part 3-2: Requirements and guidance in the use of mathematical and logical techniques for establishing exact properties of software and its documentation

IEC TS 61508-3-2:2024-08(en)

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

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

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

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

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 It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

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

IEC Products & Services Portal - products.iec.ch
Discover our powerful search engine and read freely all the publications previews, graphical symbols and the glossary. With a subscription you will always have access to up to date content tailored to your needs.

Electropedia - www.electropedia.org
The world's leading online dictionary on electrotechnology, containing more than 22 500 terminological entries in English and French, with equivalent terms in 25 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC TS 61508-3-2

Edition 1.0 2024-08

TECHNICAL SPECIFICATION

Functional safety of electrical/electronic/programmable electronic safety-related systems –
Part 3-2: Requirements and guidance in the use of mathematical and logical techniques for establishing exact properties of software and its documentation

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 25.040.40

ISBN 978-2-8322-9565-6

CONTENTS

FOREWORD 3
INTRODUCTION 5
1 Scope 6
2 Normative references 6
3 Terms, definitions and abbreviations 6
3.1 Terms and definitions 6
3.2 Abbreviations 15
4 Conformance to this document 15
5 Formal safety requirements specification 16
6 Formal software architecture / Design specification 16
7 Higher-level programming languages: Selection of ESCL 17
8 Compilation to object code 17
9 Run-time errors and exceptions 17
10 Applicable techniques 18
Annex A (normative) Applicable Mathematical and Logical Techniques 19
Annex B (informative) Specific Mathematical and Logical Techniques 21
Annex C (informative) Properties assured by application of specific M&LT techniques 24
Annex D (informative) Software refinement from safety specification to Code 26
D.1 Transformation 26
D.2 Refinement 26
Bibliography 27

Table A.1 – M&LT Techniques 19
Table B.1 – Specific M&LT Techniques/Tools 21
Table C.1 – Properties Assured by M&LT Techniques 24

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

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

Part 3-2: Requirements and guidance in the use of mathematical and logical techniques for establishing exact properties of software and its documentation

FOREWORD

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

IEC TS 61508-3-2 has been prepared by subcommittee 65A: System aspects, of IEC technical committee 65: Industrial-process measurement, control and automation. It is a Technical Specification.

The text of this Technical Specification is based on the following documents:

Draft	Report on voting
65A/1113/DTS	65A/1143/RVDTS

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

The language used for the development of this Technical Specification is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available at www.iec.ch/members_experts/refdocs. The main document types developed by IEC are described in greater detail at www.iec.ch/publications.

A list of all parts in the IEC 61508 series, published under the general title *Functional safety of electrical/electronic/programmable electronic safety-related systems*, can be found on the IEC website.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under webstore.iec.ch in the data related to the specific document. At this date, the document will be

- reconfirmed,
- withdrawn, or
- revised.

INTRODUCTION

IEC 61508-1:2010 through IEC 61508-7:2010 forms the series of basic standards for the functional safety of electric, electronic and programmable electronic systems (E/E/PE systems). It covers the life cycle of these systems. The major part of the functionality of such systems is often implemented in software. IEC 61508-3:2010 sets software requirements.

IEC 61508-3:2010 Annex A (normative) and Annexes B and C (informative) contain tables listing various techniques and measures, and provide some guidance to the selection of such techniques for different safety integrity levels (SIL). It lists general categories and gives different levels of recommendation for these, such as "not recommended", "recommended" or "highly recommended", as well as more specific techniques for various phases of software development.

These techniques and measures are a mix of generic and specific. The phrase "Formal Methods" as used in IEC 61508-3 refers to the use of mathematical and logical techniques for specifying, assessing, designing and verifying software. Today, such methods are available for specifying requirements, for the assessment of the design, for checking source code and object code and for the derivation of test suites, and for monitoring the correct operation of software at runtime. In this document, we refer to these methods by using the description as mathematical and logical techniques (M< sometimes doubled as M&LT techniques). Some of the M&LT techniques in this document are not restricted to software development, being equally applicable to other digital-system-based engineering technologies. None of the M&LT techniques are limited to the domain of safety-related software systems, although in this document only safety-related applications of M&LT techniques are explicitly addressed.

Use of the recommended methods of IEC 61508-3:2010, Annexes A, B and C do not rule out, for example, susceptibility of the software to run-time failure. State of the art in software development enables various types of run-time failures to be ruled out through rigorous development of the software. It is possible using techniques identified here to assure freedom from many types of software run-time failures.

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

Part 3-2: Requirements and guidance in the use of mathematical and logical techniques for establishing exact properties of software and its documentation

1 Scope

This Technical Specification, part of the IEC 61508 series, covers the general assurance of dependable software used in critical operational-technology (OT) which is running on hardware devices which are specified as part of the OT application. It is particularly aimed at safety-related software which is being developed according to the E/E/PE software functional safety standard IEC 61508-3; in particular, the development of the software follows a Formal Safety Requirements Specification. Successful use of some or all of the assurance points specified in this document enhances the confidence that a particular piece of safety-related software meets the requirements of the SIL of the safety function which it (partially or fully) implements, and thereby increases the systematic capability of the software.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61508-3:2010, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 3: Software requirements*

IEC 61508-4:2010, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 4: Definitions and abbreviations*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at http://www.electropedia.org/
- ISO Online browsing platform: available at http://www.iso.org/obp

3.1.1
abstract interpretation
<of a computer program> static analysis of a program on abstract program states or abstract machine states that provides sound results for a given property, i.e., that never reports the property to hold if it does not hold

3.1.2
assurance point
<in software development using M&LT> triple, consisting of software and/or documentation (S1), another SW and/or documentation (S2), and a property P jointly of S1 and S2 such that P(S1.S2) can be formally mathematically proved

Note 1 to entry: Although a mathematical proof is formally possible at an assurance point, such a proof can be too complex, or require too many resources, to be given in its entirety and reliably checked, say by an assessor.

3.1.3
automated prover
automated theorem prover
computer program which performs inference in a formal logic between sentences of a formal language

3.1.4
automated proving
using an automated prover

3.1.5
characteristic function
<of a set or a relation> function (of variable domain) and codomain {0,1} such that its value is one when its argument belongs to the set or relation, and its value is 0 when its argument does not belong to the set or relation

3.1.6
code generator
automatic code generator
software which effects the transformation of a high-level language program or a specification into a common third- or fourth-generation-language program

3.1.7
coding standard
programming-language subsetting

<in M&LT techniques> restrictions on the constructs with which a program can be written in a high-level programming language

Note 1 to entry: The purpose of a coding standard which restricts programming-language constructs that may be used is to assure an unambiguous semantics to a program written according to the coding standard.

Note 2 to entry: A typical coding standard (n.b., the singular version of this phrase is uncommon) will ensure that known causes of unreliable behaviour in program code are avoided, e.g., pointer variables are proscribed; undefined or compiler-variable language features are avoided. Coding standards for programs written in a language without strong data typing might well ensure that the anticipated or specified range of input or output data is explicitly checked at input or output.

Note 3 to entry: The term coding standards in general use often refers to further properties of code than subsetting.

3.1.8
compilation
translation operation that translates executable source code level (ESCL) into object code (OC)

Note 1 to entry: The definition explicitly mentions ESCL, a concept used in this document, but not generally where compilers are used and compilation is practiced.

3.1.9
completeness
<of a formal language with a logic with respect to a given semantic property> quality of a formal language with an associated logic and a formal semantics that holds with respect to a given semantic property if every sentence of the language having this property can be shown to have this property through inference in the associated logic

Note 1 to entry: An algorithm is complete with respect to a property if it always proves the property if it holds.

3.1.10
compositional
<of a formal semantics> taking as input just a sentence of a formal language (and not, say, any indication of context, say the reference of indexicals) and which constructs a transcription purely using the parse tree of the sentence

3.1.11
computable
recursive
<computable-function theory> turing-computable

Note 1 to entry: Recursive is used here in the sense in which this term is used in Turing computability and recursive function theory [1][1], [2].

Note 2 to entry: There are in the mathematical literature other notions of “computable” than Turing-computable. Some are known to be equivalent to Turing-computable, but for some the question is open. This definition thus disambiguates use of the term “computable”. Similarly, in computer science and system engineering the term “recursive” has variable meanings; this definition disambiguates use of the term.

Note 3 to entry: The term “decidable” is often used to refer to properties; the term “computable” to functions. A property is decidable if and only if its characteristic function is computable.

Note 4 to entry: “Turing computable” is a concept which is usually defined over many tens of pages of textbooks on recursion theory, often using many subsidiary concepts [1] [2]. There lacks a short definition to use here.

Note 5 to entry: There are other notions of computability in logic and computer science which are not, or not known to be, reducible to Turing-computability.

3.1.12
consistency
property of a collection of sentences of a formal language that they are not contradictory

3.1.13
contradictory
property of a collection of sentences of a formal language when their renderings are mutually exclusive, that is, they cannot all hold or be realised at the same time in the same structure

3.1.14
decidable
having a Turing-computable characteristic function

[1] Numbers in square brackets refer to the Bibliography.

3.1.15
element
part of a subsystem comprising a single component or any group of components that performs one or more element safety functions

Note 1 to entry: An element may comprise hardware and/or software.

Note 2 to entry: A typical element is a sensor, programmable controller or final element

[SOURCE: IEC 61508-4:2010, 3.4.5]

3.1.16
element safety function
that part of a safety function which is implemented by an element

[SOURCE: IEC 61508-4:2010, 3.5.3]

3.1.17
executable source code level
ESCL
source code at which the exact behaviour of the software is completely described

3.1.18
formal inference
<in a logic> derivation of a sentence from premises using the explicit formal rules of inference of the logic

3.1.19
formal language
<in M&LT> language with a defined syntax which is parsable without exception by means of digital computation, and such that it is computable whether a given sequence of symbols forms a sentence of the language or not

Note 1 to entry: The term “sentence” is used in this document for a member of a formal language, but in many formal languages other terms are more appropriately used. When a formal language consists of strings of symbols simpliciter, as do formal languages in formal language theory, automata theory, and formal logic, then a well-formed member is called a sentence, except in formal logic, where it is called a well-formed formula. In programming languages, a well-formed member is called a valid program, and in specification languages, a well-formed member is called a valid specification. In diagrammatic languages, a well-formed member would be called a valid diagram.

Note 2 to entry: This notion of formal language is that used in logic, linguistics, mathematics, formal language theory, automata theory and theory of compilation. Formal languages such as (engineering and software) specification languages also often come with a preferred formal semantics (e.g., Z, TLA) and the use of the term in such software engineering contexts often implicitly includes the formal semantics.

Note 3 to entry: In mathematics and automata theory, the term formal language denotes just a set of strings of symbols from a defined symbol set, with no constraints upon how these strings are formed. However, all the formal languages used in M&LT satisfy the conditions given in the definition.

Note 4 to entry: In formal logic, the term “language” (without the prefix “formal”) customarily refers to the set of non-logical symbols. The set of logical symbols of a formal language of logic is customarily taken to be determined, although it varies between first-order logic and higher-order logics.

3.1.20
formal logic
propositional logic, predicate logic, higher-order logic, combinatory logic, modal logic, non-classical logic, other mathematical language structure based on a formal language which has a notion of formal inference, consistency and contradiction

3.1.21
formal proof
<in a formal logic, of a sentence from a given set of sentences> written formal inference of the sentence in the formal logic, using as premises the sentences in the given set