ISO 61508 Development

SCSD Lecture 21 Mar 2024

Overview

- This lecture is based on IEC 61508-5 and -6
- IEC 61508-7 considers a range of techniques but many of those are outdated.
- IEC 61508 is focused on safety functions in controlling equipment
- It has however, had considerable influence on other software standards
- The focus is on safety functions whose typical mode of operation is:
  - To be called on infrequently by the control software or from other sources
  - This ”demand” should result in the system being moved into a safe state.
- So, the focus for safety function is the probability of failure on demand.
- IEC 61508 also considers systems with much more frequent demand or the need to operate continuously

Safety Integrity Level

- **Safety Integrity** is defined as “The probability of a **Safety Instrumented Function** (SIF) satisfactorily performing the required **safety functions** under all stated conditions within a stated period of time”.
- **Safety Integrity Level (SIL)**: levels 1-4 are categories of safety integrity used in specifying the safety integrity requirements of safety functions.

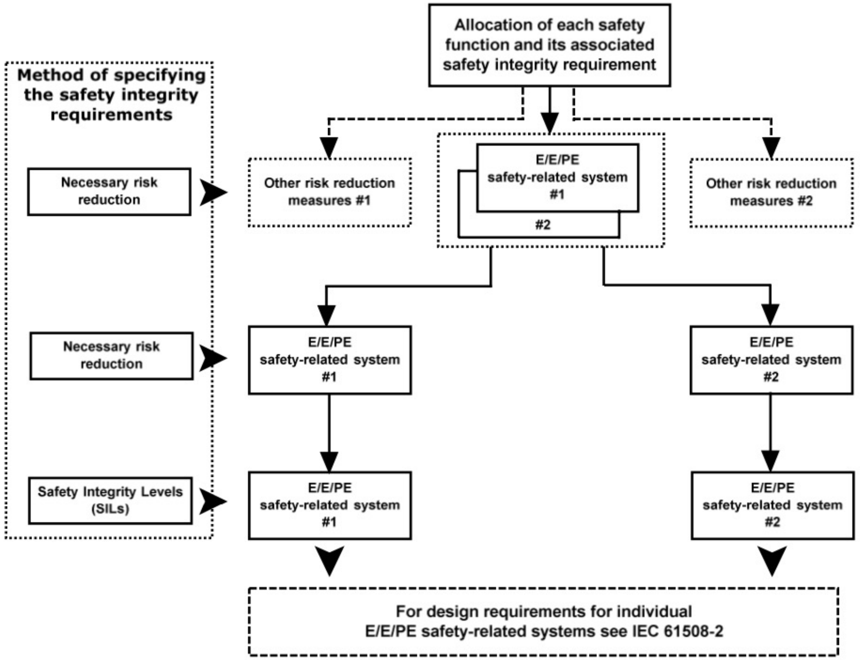
Definitions of SIL levels

Safety Integrity level	**Probability of Failure on Demand**	**Risk Reduction Factor**
SIL 4	10-5≥ PofD <10-4	100,000 to 10,000
SIL 3	10-4≥ PofD <10-3	10,000 to 1,000
SIL 2	10-3≥ PofD < 10-2	1,000 to 100
SIL 1	10-2≥ PofD < 10-1	100 to 10

Risk and Safety Integrity

- Risk is concerned with the overall likelihood and consequences of an event (for the whole system).
- Safety integrity applies to the electrical, electronic and programmable parts of the system together with other risk reduction measures.
- Safety integrity is a measure of the likelihood that these measures achieve the required risk reduction for the safety functions.
- Once tolerable risk has been identified (and the risk of operating the system without any risk reduction is identified) the necessary risk reduction will be determined and safety integrity requirements can be determined (and SIL level allocated to functions).

Safety Requirements


Allocation of each safety function and its associated safety integrity requirement
Method of specifying the safety integrity requirements
Necessary risk reduction
Other risk reduction measures #1
E/E/PE safety-related system #1
#2
Other risk reduction measures #2
Necessary risk reduction
E/E/PE safety-related system #1
E/E/PE safety-related system #2
Safety Integrity Levels (SILs)
E/E/PE safety-related system #1
E/E/PE safety-related system #2
For design requirements for individual E/E/PE safety-related systems see IEC 61508-2

NOTE 1 Safety integrity requirements are associated with each safety function before allocation (see 7.5.2.3 and 7.5.2.4 of IEC 61508-1).

NOTE 2 A safety function may be allocated across more than one safety-related system.

Figure A.7 – Allocation of safety requirements to the E/E/PE safety-related systems, and other risk reduction measures

Methods for Determining SIL requirements

- The following should be taken into account:
  1. the risk acceptance criteria that need to be met. Some of the techniques will not be suitable if it is required to demonstrate that risk has been reduced to as low as reasonably practicable;
  2. the mode of operation of the safety function. Some methods are only suitable for low demand mode;
  3. the knowledge and experience of the persons undertaking the SIL determination and what has been the traditional approach in the sector;
  4. the confidence needed that the resulting residual risk meets the criteria specified by the user organisation. Some of the methods can be linked back to quantified targets but some approaches are qualitative only;
  5. more than one method may be used. One method may be used for screening purposes followed by another more rigorous approach if the screening method shows the need for high safety integrity levels;
  6. the severity of the consequences. More rigorous methods may be selected for con- sequences that include multiple fatalities;
  7. whether common cause occurs between the E/E/PE safety related systems or between the E/E/PE safety related system and demand causes.

Methods for determining SILs

- ALARP
- Quantitative method
- Risk Graph
- Layer of Protection Analysis
- Hazardous event severity matrix

Quantitative Method

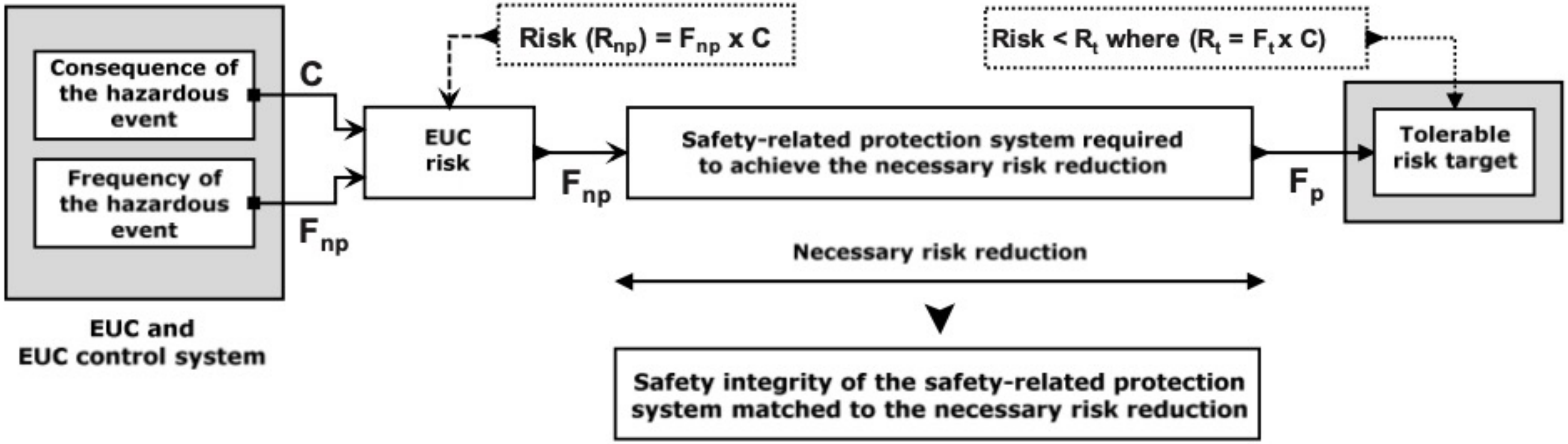
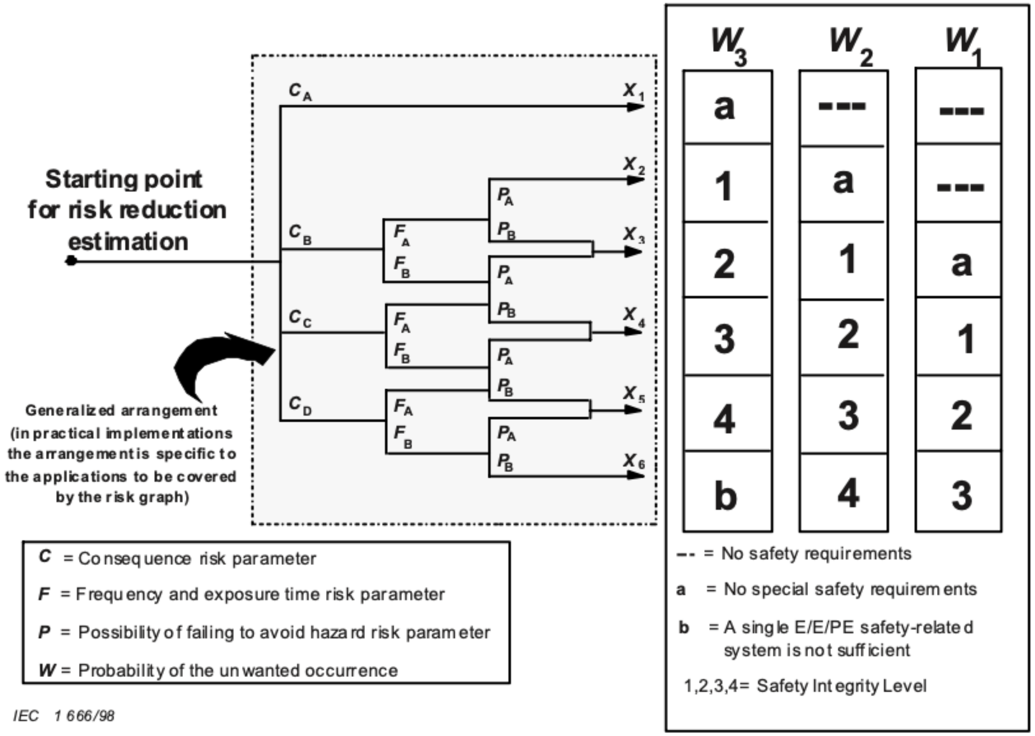

Risk (R_{np}) = F_{np} x C
Risk < R_t where (R_t = F_t x C)
Consequence of the hazardous event
Frequency of the hazardous event
C
F_{np}
EUC risk
F_{np}
Safety-related protection system required to achieve the necessary risk reduction
F_p
Tolerable risk target
EUC and EUC control system
Necessary risk reduction
Safety integrity of the safety-related protection system matched to the necessary risk reduction

Figure D.1 – Safety integrity allocation – example for safety-related protection system

Risk Graph


Starting point for risk reduction estimation
Generalized arrangement (in practical implementations the arrangement is specific to the applications to be covered by the risk graph)
C_A
C_B
C_C
C_D
F_A
F_B
P_A
P_B
X_1
X_2
X_3
X_4
X_5
X_6

W_3	W_2	W_1
a	---	---
1	a	---
2	1	a
3	2	1
4	3	2
b	4	3

C = Consequence risk parameter

F = Frequency and exposure time risk parameter

P = Possibility of failing to avoid hazard risk parameter

W = Probability of the unwanted occurrence

IEC 1 666/98

--- = No safety requirements

a = No special safety requirements

b = A single E/E/PE safety-related system is not sufficient

1,2,3,4= Safety Integrity Level

Figure E.1 – Risk Graph: general scheme

The parameter W captures the required risk reduction for the system.

Layer of Protection Analysis

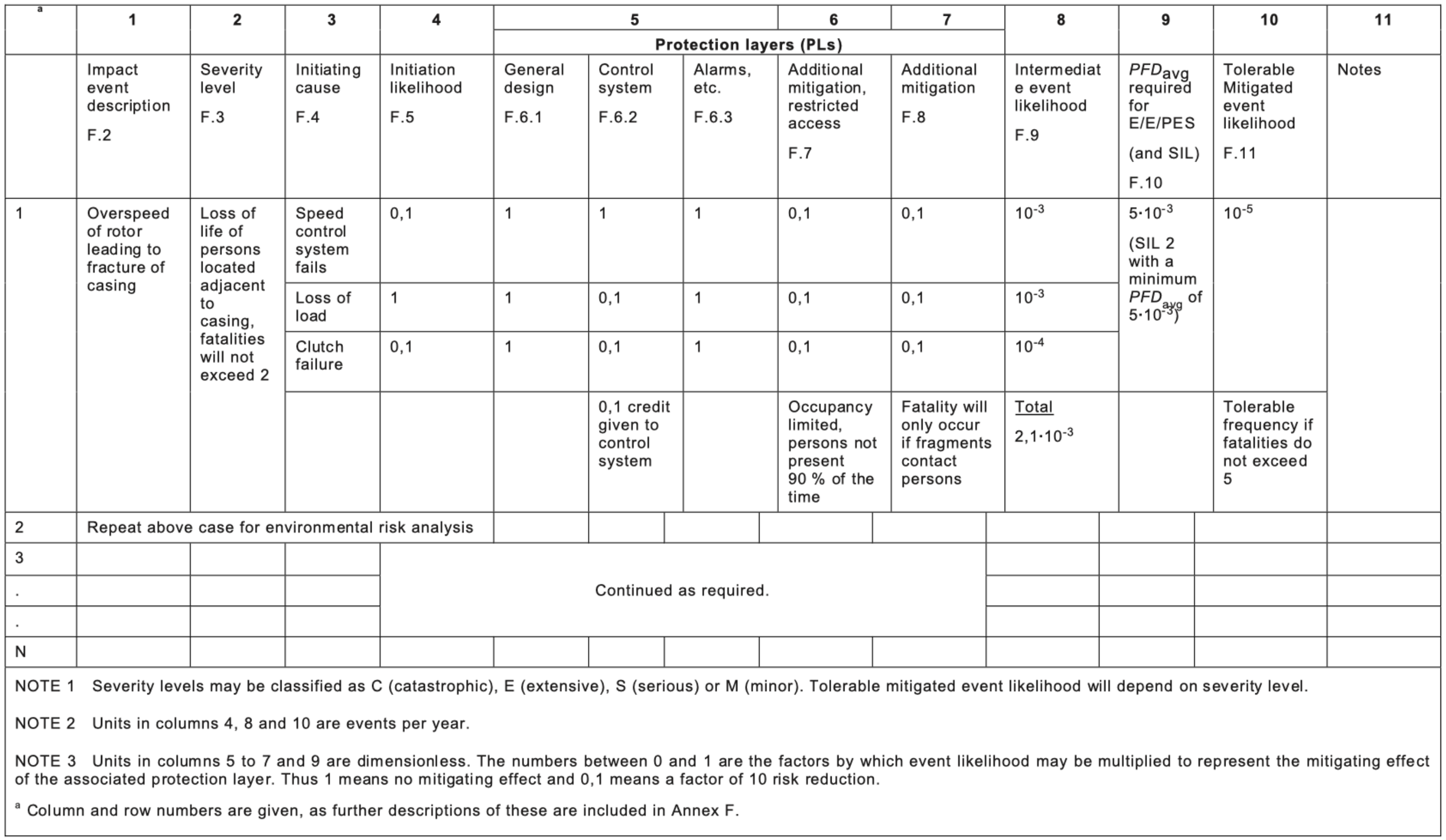


Table F.1 – LOPA report

a	1	2	3	4	5			6	7	8	9	10	11
					Protection layers (PLs)								
	Impact event description F.2	Severity level F.3	Initiating cause F.4	Initiation likelihood F.5	General design F.6.1	Control system F.6.2	Alarms, etc. F.6.3	Additional mitigation, restricted access F.7	Additional mitigation F.8	Intermediate event likelihood F.9	*PFD*avg required for E/E/PES (and SIL) F.10	Tolerable Mitigated event likelihood F.11	Notes
1	Overspeed of rotor leading to fracture of casing	Loss of life of persons located adjacent to casing, fatalities will not exceed 2	Speed control system fails	0,1	1	1	1	0,1	0,1	10^{-3}	$5\cdot10^{-3}$ (SIL 2 with a minimum PFD_{avg} of $5\cdot10^{-3}$)	10^{-5}	
			Loss of load	1	1	0,1	1	0,1	0,1	10^{-3}			
			Clutch failure	0,1	1	0,1	1	0,1	0,1	10^{-4}			
						0,1 credit given to control system		Occupancy limited, persons not present 90 % of the time	Fatality will only occur if fragments contact persons	Total $2{,}1\cdot10^{-3}$		Tolerable frequency if fatalities do not exceed 5	
2	Repeat above case for environmental risk analysis												
3				Continued as required.									
.													
.													
N													

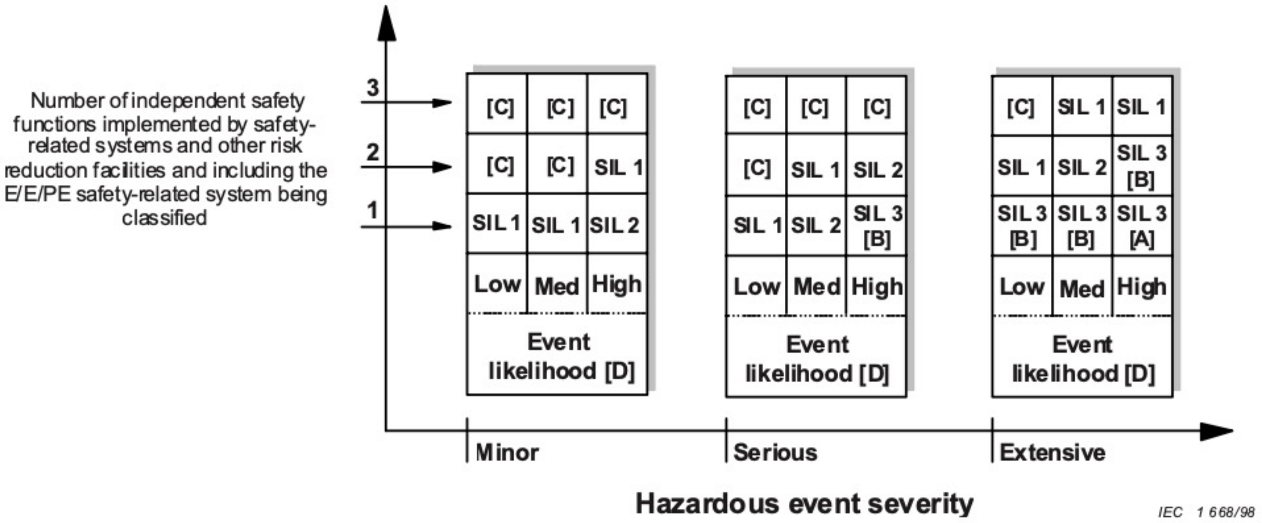
NOTE 1 Severity levels may be classified as C (catastrophic), E (extensive), S (serious) or M (minor). Tolerable mitigated event likelihood will depend on severity level.

NOTE 2 Units in columns 4, 8 and 10 are events per year.

NOTE 3 Units in columns 5 to 7 and 9 are dimensionless. The numbers between 0 and 1 are the factors by which event likelihood may be multiplied to represent the mitigating effect of the associated protection layer. Thus 1 means no mitigating effect and 0,1 means a factor of 10 risk reduction.

[a] Column and row numbers are given, as further descriptions of these are included in Annex F.

Hazardous Event Severity Matrix


Number of independent safety functions implemented by safety-related systems and other risk reduction facilities and including the E/E/PE safety-related system being classified
3
2
1
Event likelihood [D]
Minor
Serious
Extensive
Hazardous event severity
IEC 1 668/98

	Minor			Serious			Extensive		
3	[C]	[C]	[C]	[C]	[C]	[C]	[C]	SIL 1	SIL 1
2	[C]	[C]	SIL 1	[C]	SIL 1	SIL 2	SIL 1	SIL 2	SIL 3 [B]
1	SIL 1	SIL 1	SIL 2	SIL 1	SIL 2	SIL 3 [B]	SIL 3 [B]	SIL 3 [B]	SIL 3 [A]
	Low	Med	High	Low	Med	High	Low	Med	High

[A] One SIL 3 E/E/PE safety function does not provide sufficient risk reduction at this risk level. Additional risk reduction measures are required.

[B] One SIL 3 E/E/PE safety function may not provide sufficient risk reduction at this risk level. Hazard and risk analysis is required to determine whether additional risk reduction measures are necessary.

[C] An independent E/E/PE safety function is probably not required.

[D] Event likelihood is the likelihood that the hazardous event occurs without any safety function or other risk reduction measure.

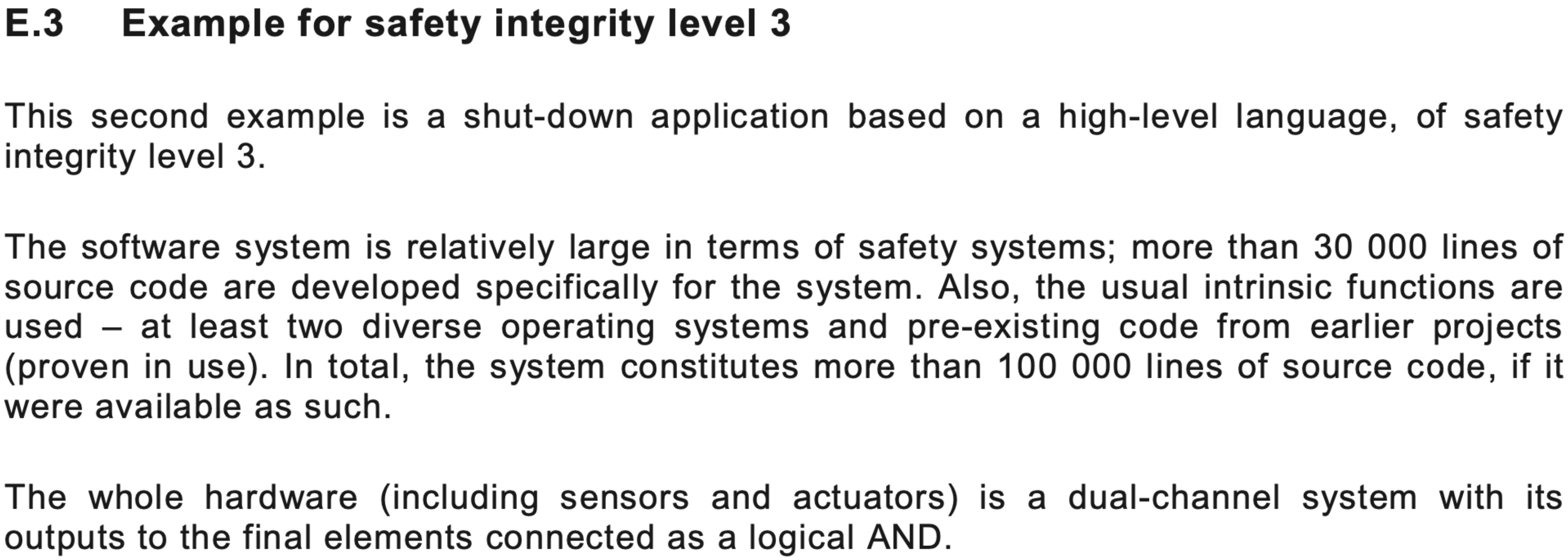
[E] Event likelihood and the total number of independent protection layers are defined in relation to the specific application.

Figure G.1 – Hazardous event severity matrix – example (illustrates general principles only)

Development Process

- Each of the tables in IEC 61508-3 are populated for the application.
- What follows is the example in IEC 61508-6 for a SIL3 system
- Each of the stages in the development process spells out the use of particular techniques.

Example introduction



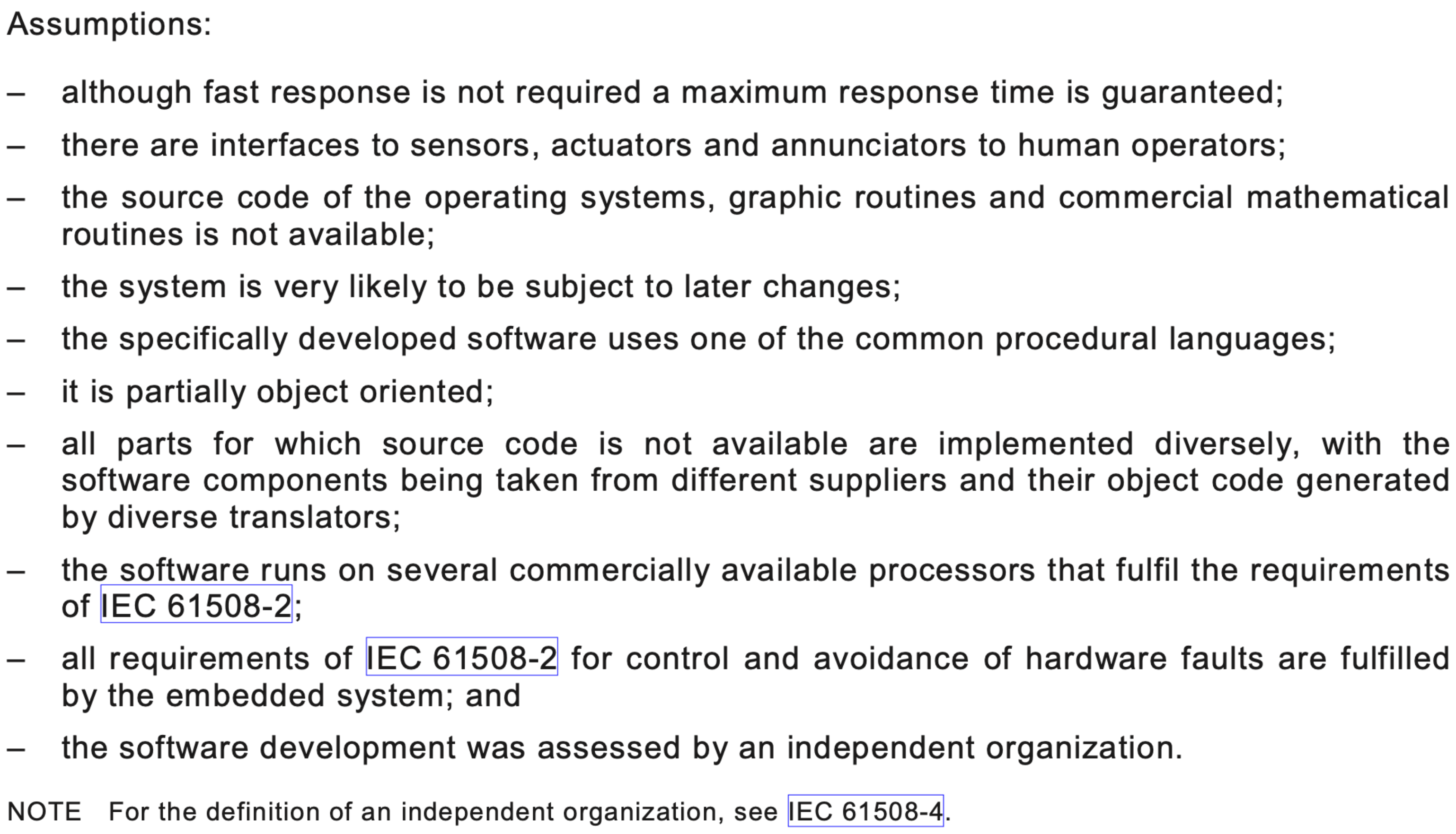
E.3 Example for safety integrity level 3

This second example is a shut-down application based on a high-level language, of safety integrity level 3.

The software system is relatively large in terms of safety systems; more than 30 000 lines of source code are developed specifically for the system. Also, the usual intrinsic functions are used – at least two diverse operating systems and pre-existing code from earlier projects (proven in use). In total, the system constitutes more than 100 000 lines of source code, if it were available as such.

The whole hardware (including sensors and actuators) is a dual-channel system with its outputs to the final elements connected as a logical AND.

Example assumptions



Assumptions:

- although fast response is not required a maximum response time is guaranteed;
- there are interfaces to sensors, actuators and annunciators to human operators;
- the source code of the operating systems, graphic routines and commercial mathematical routines is not available;
- the system is very likely to be subject to later changes;
- the specifically developed software uses one of the common procedural languages;
- it is partially object oriented;
- all parts for which source code is not available are implemented diversely, with the software components being taken from different suppliers and their object code generated by diverse translators;
- the software runs on several commercially available processors that fulfil the requirements of IEC 61508-2;
- all requirements of IEC 61508-2 for control and avoidance of hardware faults are fulfilled by the embedded system; and
- the software development was assessed by an independent organization.

NOTE For the definition of an independent organization, see IEC 61508-4.

Requirements

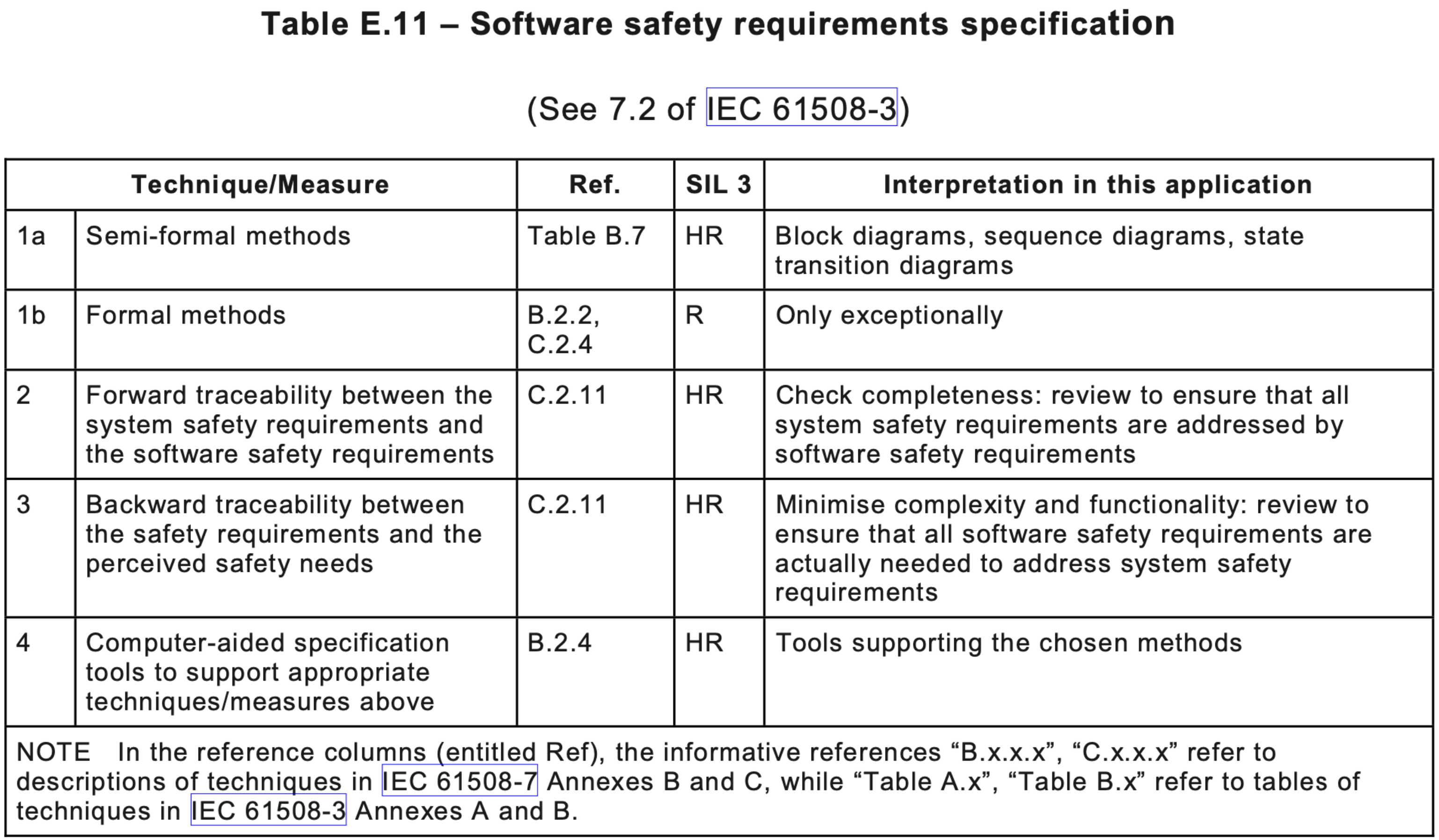


Table E.11 – Software safety requirements specification

(See 7.2 of IEC 61508-3)

Technique/Measure		Ref.	SIL 3	Interpretation in this application
1a	Semi-formal methods	Table B.7	HR	Block diagrams, sequence diagrams, state transition diagrams
1b	Formal methods	B.2.2, C.2.4	R	Only exceptionally
2	Forward traceability between the system safety requirements and the software safety requirements	C.2.11	HR	Check completeness: review to ensure that all system safety requirements are addressed by software safety requirements
3	Backward traceability between the safety requirements and the perceived safety needs	C.2.11	HR	Minimise complexity and functionality: review to ensure that all software safety requirements are actually needed to address system safety requirements
4	Computer-aided specification tools to support appropriate techniques/measures above	B.2.4	HR	Tools supporting the chosen methods

NOTE In the reference columns (entitled Ref), the informative references “B.x.x.x”, “C.x.x.x” refer to descriptions of techniques in IEC 61508-7 Annexes B and C, while “Table A.x”, “Table B.x” refer to tables of techniques in IEC 61508-3 Annexes A and B.

Architecture

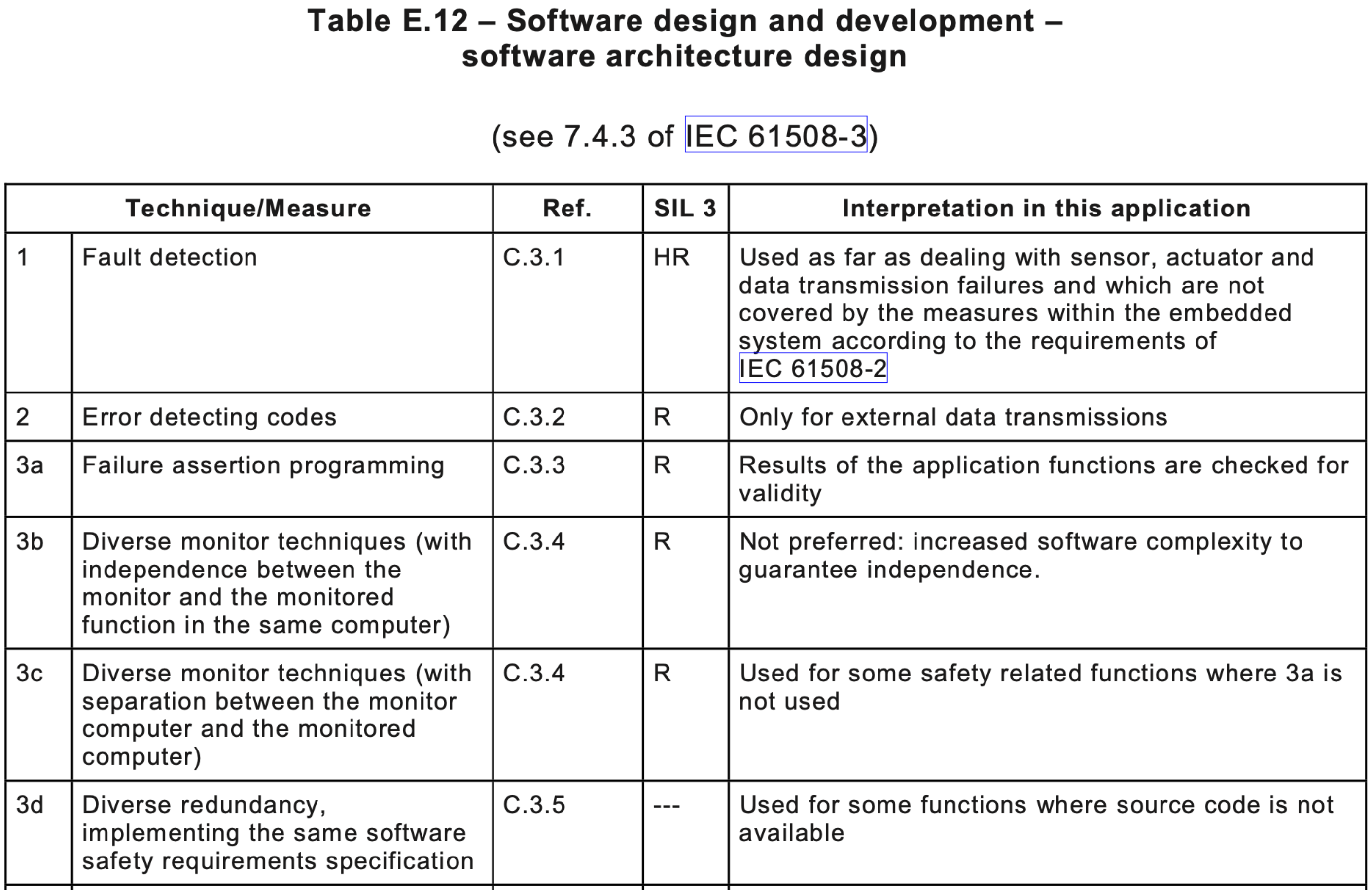


Table E.12 – Software design and development – software architecture design

(see 7.4.3 of IEC 61508-3)

Technique/Measure		Ref.	SIL 3	Interpretation in this application
1	Fault detection	C.3.1	HR	Used as far as dealing with sensor, actuator and data transmission failures and which are not covered by the measures within the embedded system according to the requirements of IEC 61508-2
2	Error detecting codes	C.3.2	R	Only for external data transmissions
3a	Failure assertion programming	C.3.3	R	Results of the application functions are checked for validity
3b	Diverse monitor techniques (with independence between the monitor and the monitored function in the same computer)	C.3.4	R	Not preferred: increased software complexity to guarantee independence.
3c	Diverse monitor techniques (with separation between the monitor computer and the monitored computer)	C.3.4	R	Used for some safety related functions where 3a is not used
3d	Diverse redundancy, implementing the same software safety requirements specification	C.3.5	---	Used for some functions where source code is not available

Support Tools

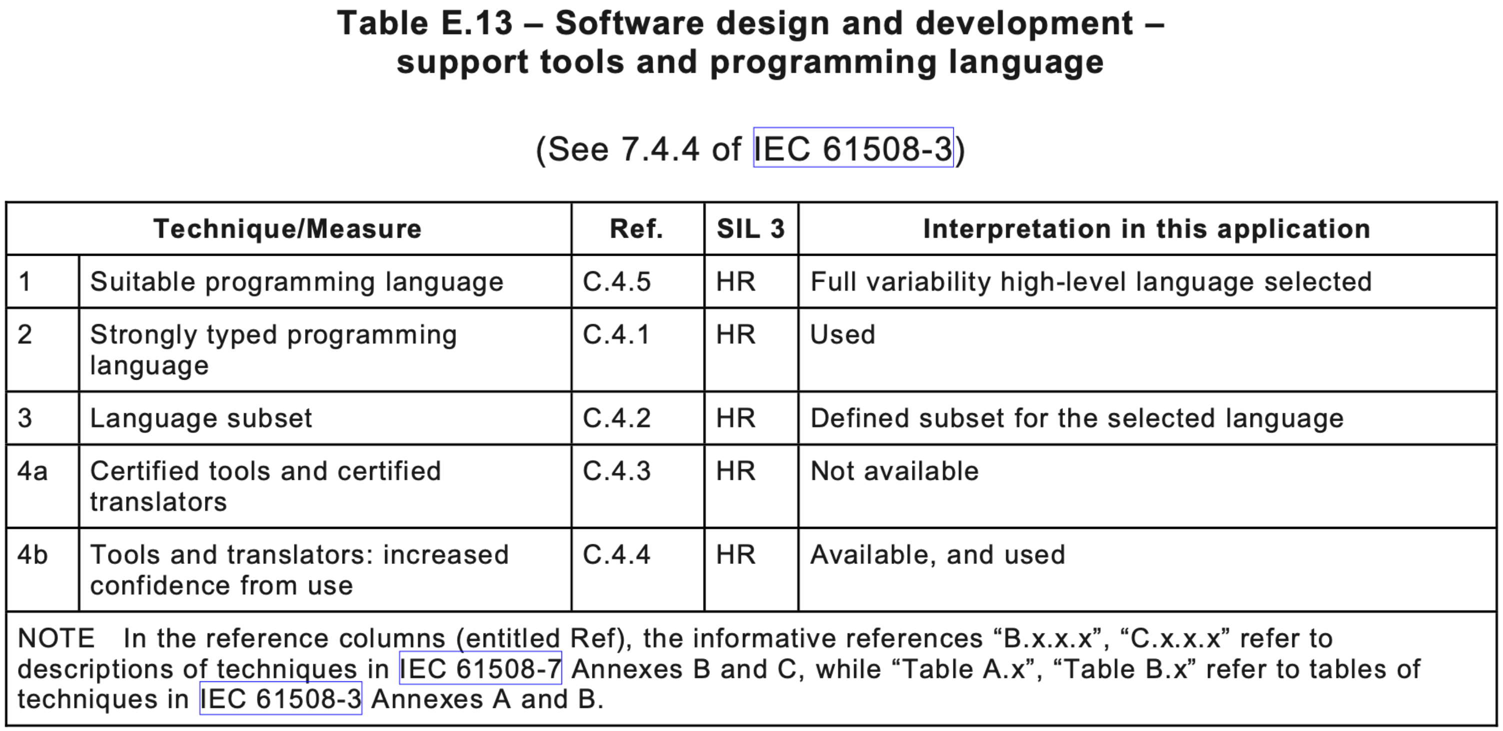


Table E.13 – Software design and development – support tools and programming language

(See 7.4.4 of IEC 61508-3)

Technique/Measure		Ref.	SIL 3	Interpretation in this application
1	Suitable programming language	C.4.5	HR	Full variability high-level language selected
2	Strongly typed programming language	C.4.1	HR	Used
3	Language subset	C.4.2	HR	Defined subset for the selected language
4a	Certified tools and certified translators	C.4.3	HR	Not available
4b	Tools and translators: increased confidence from use	C.4.4	HR	Available, and used

NOTE In the reference columns (entitled Ref), the informative references “B.x.x.x”, “C.x.x.x” refer to descriptions of techniques in IEC 61508-7 Annexes B and C, while “Table A.x”, “Table B.x” refer to tables of techniques in IEC 61508-3 Annexes A and B.

Detailed Design

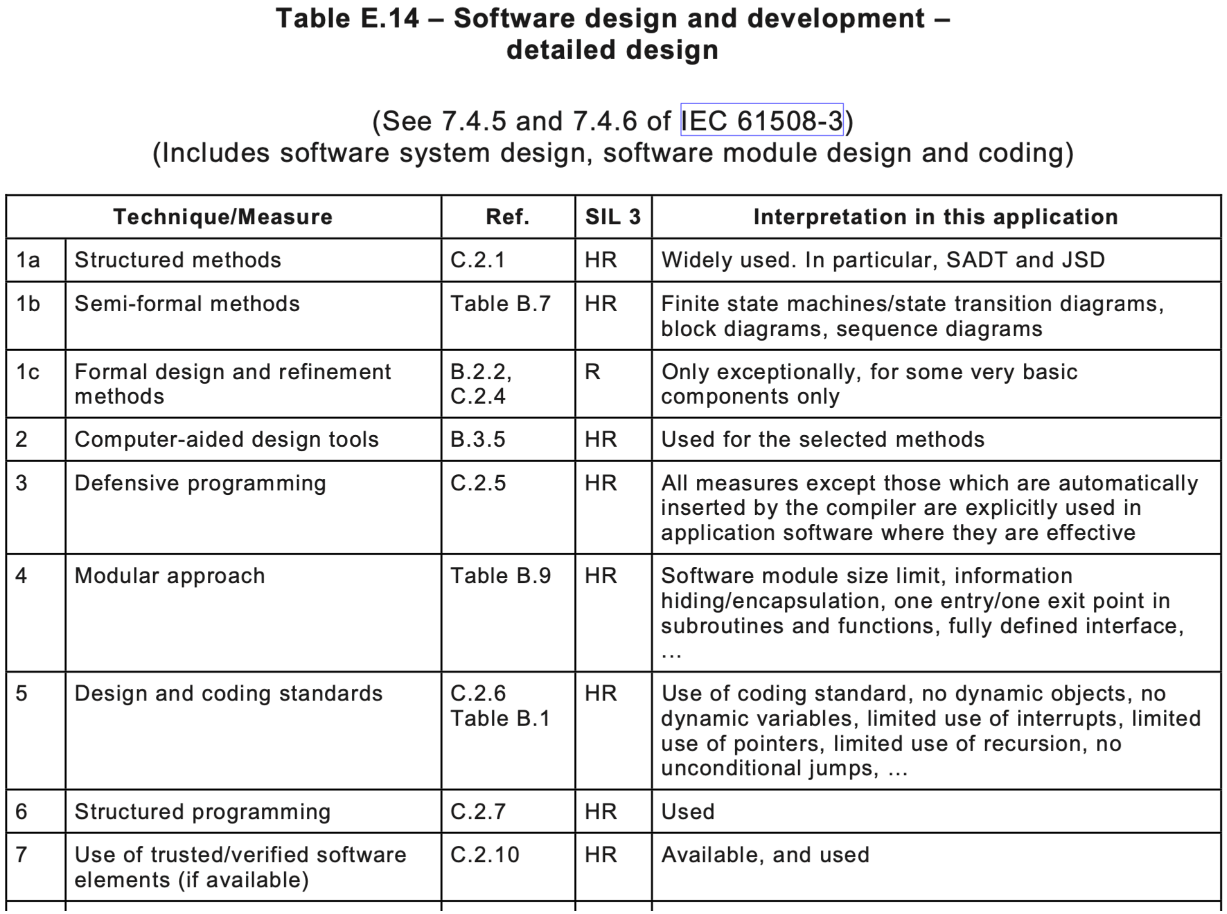


Table E.14 – Software design and development – detailed design

(See 7.4.5 and 7.4.6 of IEC 61508-3)
(Includes software system design, software module design and coding)

Technique/Measure		Ref.	SIL 3	Interpretation in this application
1a	Structured methods	C.2.1	HR	Widely used. In particular, SADT and JSD
1b	Semi-formal methods	Table B.7	HR	Finite state machines/state transition diagrams, block diagrams, sequence diagrams
1c	Formal design and refinement methods	B.2.2, C.2.4	R	Only exceptionally, for some very basic components only
2	Computer-aided design tools	B.3.5	HR	Used for the selected methods
3	Defensive programming	C.2.5	HR	All measures except those which are automatically inserted by the compiler are explicitly used in application software where they are effective
4	Modular approach	Table B.9	HR	Software module size limit, information hiding/encapsulation, one entry/one exit point in subroutines and functions, fully defined interface, …
5	Design and coding standards	C.2.6 Table B.1	HR	Use of coding standard, no dynamic objects, no dynamic variables, limited use of interrupts, limited use of pointers, limited use of recursion, no unconditional jumps, …
6	Structured programming	C.2.7	HR	Used
7	Use of trusted/verified software elements (if available)	C.2.10	HR	Available, and used

Test

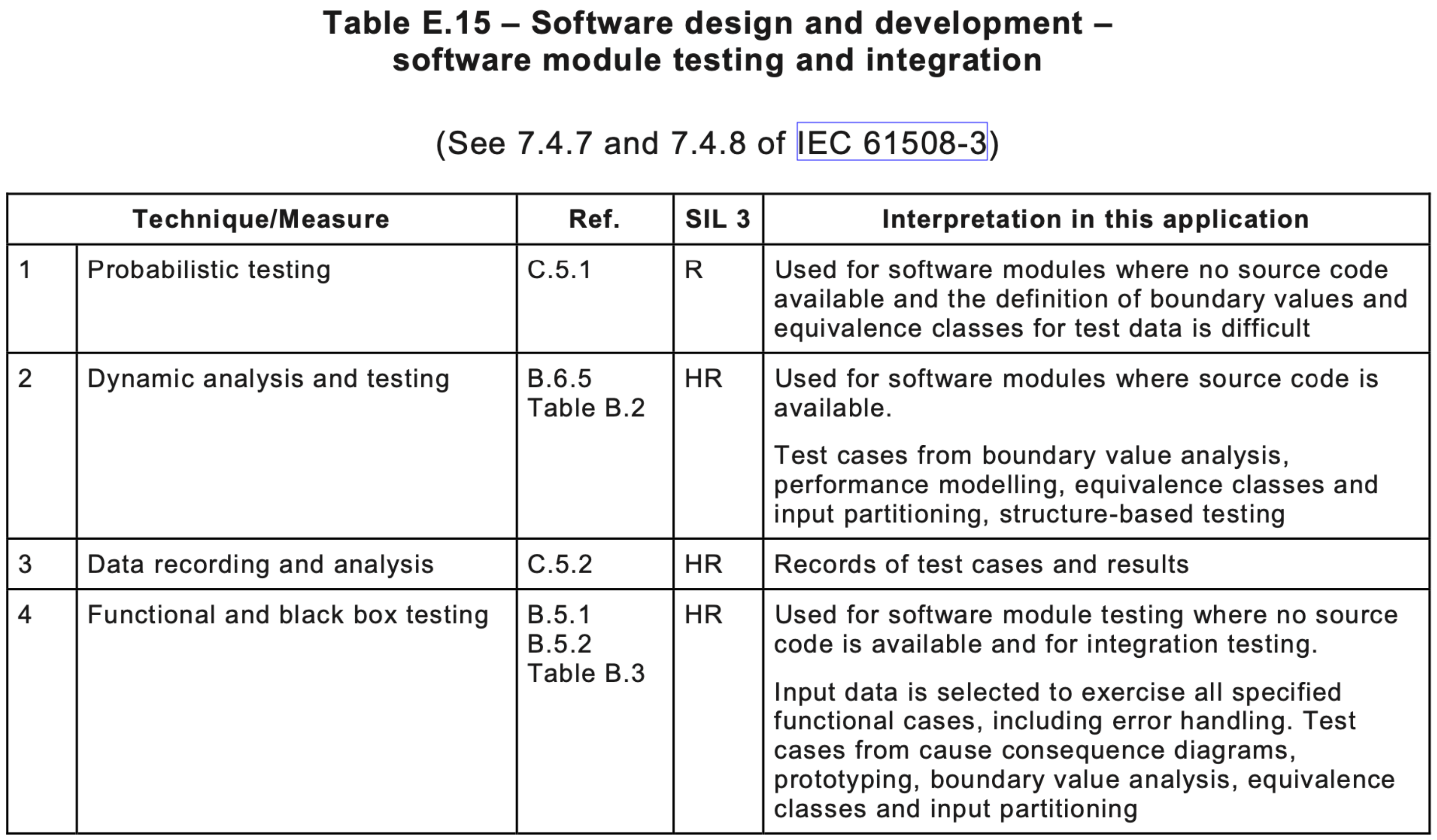


Table E.15 – Software design and development – software module testing and integration

(See 7.4.7 and 7.4.8 of IEC 61508-3)

Technique/Measure		Ref.	SIL 3	Interpretation in this application
1	Probabilistic testing	C.5.1	R	Used for software modules where no source code available and the definition of boundary values and equivalence classes for test data is difficult
2	Dynamic analysis and testing	B.6.5 Table B.2	HR	Used for software modules where source code is available. Test cases from boundary value analysis, performance modelling, equivalence classes and input partitioning, structure-based testing
3	Data recording and analysis	C.5.2	HR	Records of test cases and results
4	Functional and black box testing	B.5.1 B.5.2 Table B.3	HR	Used for software module testing where no source code is available and for integration testing. Input data is selected to exercise all specified functional cases, including error handling. Test cases from cause consequence diagrams, prototyping, boundary value analysis, equivalence classes and input partitioning

Integration

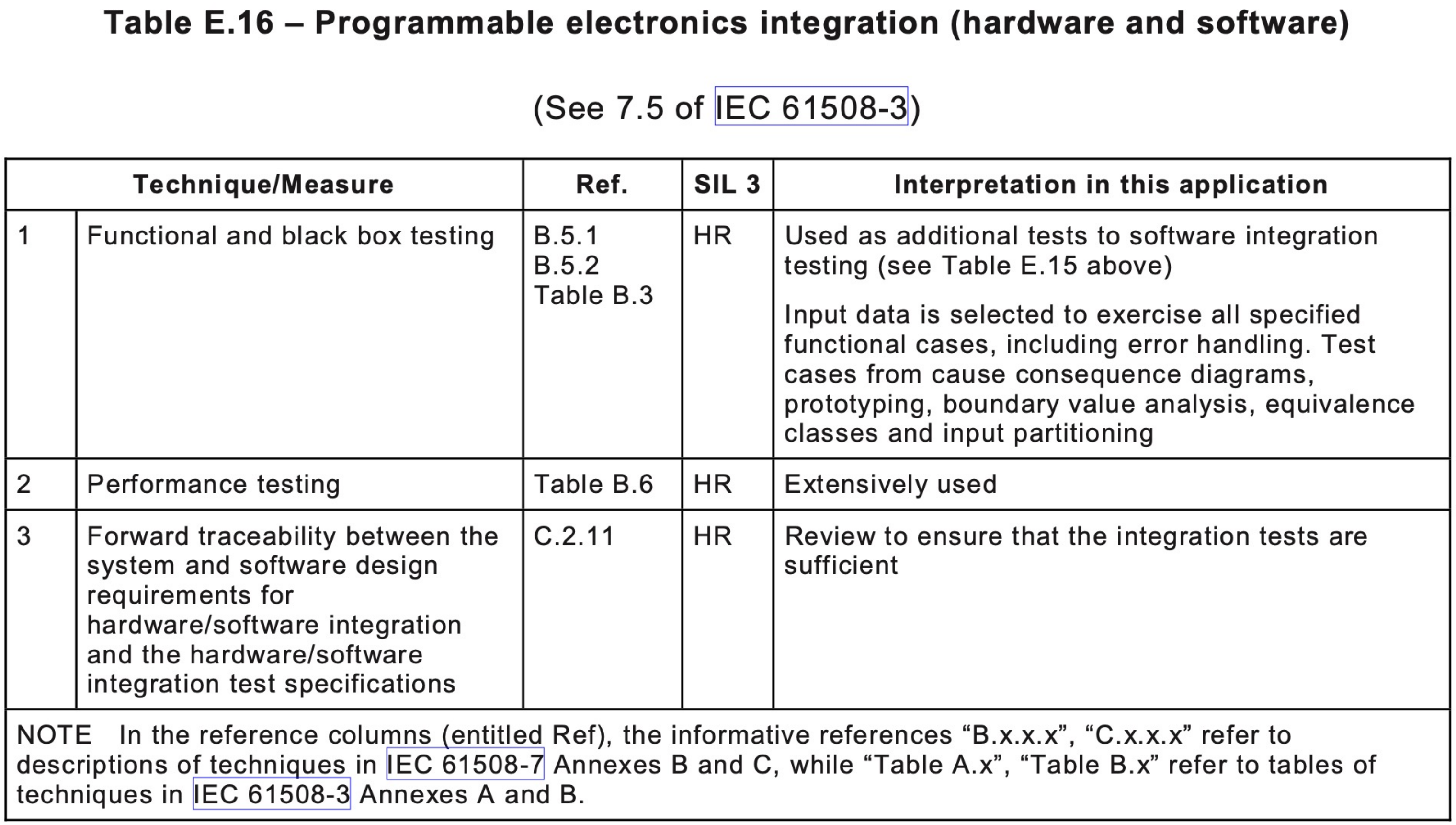


Table E.16 – Programmable electronics integration (hardware and software)

(See 7.5 of IEC 61508-3)

Technique/Measure		Ref.	SIL 3	Interpretation in this application
1	Functional and black box testing	B.5.1 B.5.2 Table B.3	HR	Used as additional tests to software integration testing (see Table E.15 above) Input data is selected to exercise all specified functional cases, including error handling. Test cases from cause consequence diagrams, prototyping, boundary value analysis, equivalence classes and input partitioning
2	Performance testing	Table B.6	HR	Extensively used
3	Forward traceability between the system and software design requirements for hardware/software integration and the hardware/software integration test specifications	C.2.11	HR	Review to ensure that the integration tests are sufficient

NOTE In the reference columns (entitled Ref), the informative references “B.x.x.x”, “C.x.x.x” refer to descriptions of techniques in IEC 61508-7 Annexes B and C, while “Table A.x”, “Table B.x” refer to tables of techniques in IEC 61508-3 Annexes A and B.

Summary

- Safety Integrity
- SILs
- Determining SILs
- Using SILs to determine development process