Introduction to 61508

Nicholas Mc Guire
Distributed & Embedded Systems Lab
Lanzhou, China
safety@osadl.org, mcguire@lzu.edu.cn

61508 intro

Focus:

- Managing complexity
- Provide Methodology of achieving tolerable risk
- Specify well defined procedures
- Focus on generic aspects
- Says nothing about certification !

61508 is a basic safety standard - it is the basis for a number of application sector specific standards.

What is 61508 ?

- procedural safety life-cylce -> safety case
- risk based approach
- generic safety life-cycle specification

A major objective is to facilitate the development of application sector specific standards [61508-1 Introduction]

61508 scope

- Functional Safety
- Focus on a monolitic safety case
- Slanted towards systems of low complexity
- Targeting design and specification faults
- Limited concidereations for human factors [61508-1 1.2 Note 2]

61508 constraints

- System Level
- global safety context
- somewhat hardware centric
- no notion of failure mode (fail-safe/fail operational)
- slant towards low complexity PES

61508 Context

61508 - reduce risk to an acceptable level:

- Social
- Economic
- Regulatory (National)

61508 must be reinterpreted in the specific context of its application - don't limit this to the technical aspects only!

61508 Principle

chieving acceptable level of risk:

- Risk Assessmen
- Risk Reduction

No system is risk free

61508 Flow

```
-> Indentify potential hazards
  -> Map to risks
    -> Derive SIL requirement(s) for system
      -> Apply appropriate methods
        -> Justify risk mitigation
```

This path is followed by all derived standards inside a framework for functional safety management. Note that it does not depend on the compomemt being COTS/OSS or bespoke.

Types of Safety

- Reactive Safety
  - fault detection
  - fault reaction
  - self-check requirements
- Composit Safety
  - fault tollerance
  - detectability through isolation
  - availability issues

61508 does not concider fail-safe/fail-operational - it is a very generic process.

Functional Safety Architecture

- Redundancy/Replication
  - 2oo2/NooM
  - relevance of random errors (types)
  - SIL level [61508-3 7.4.2.8 Note]
- Diversity
  - types of diversity
  - effects of diversity
  - safety case based on diversity [61508-6 Appendix E]

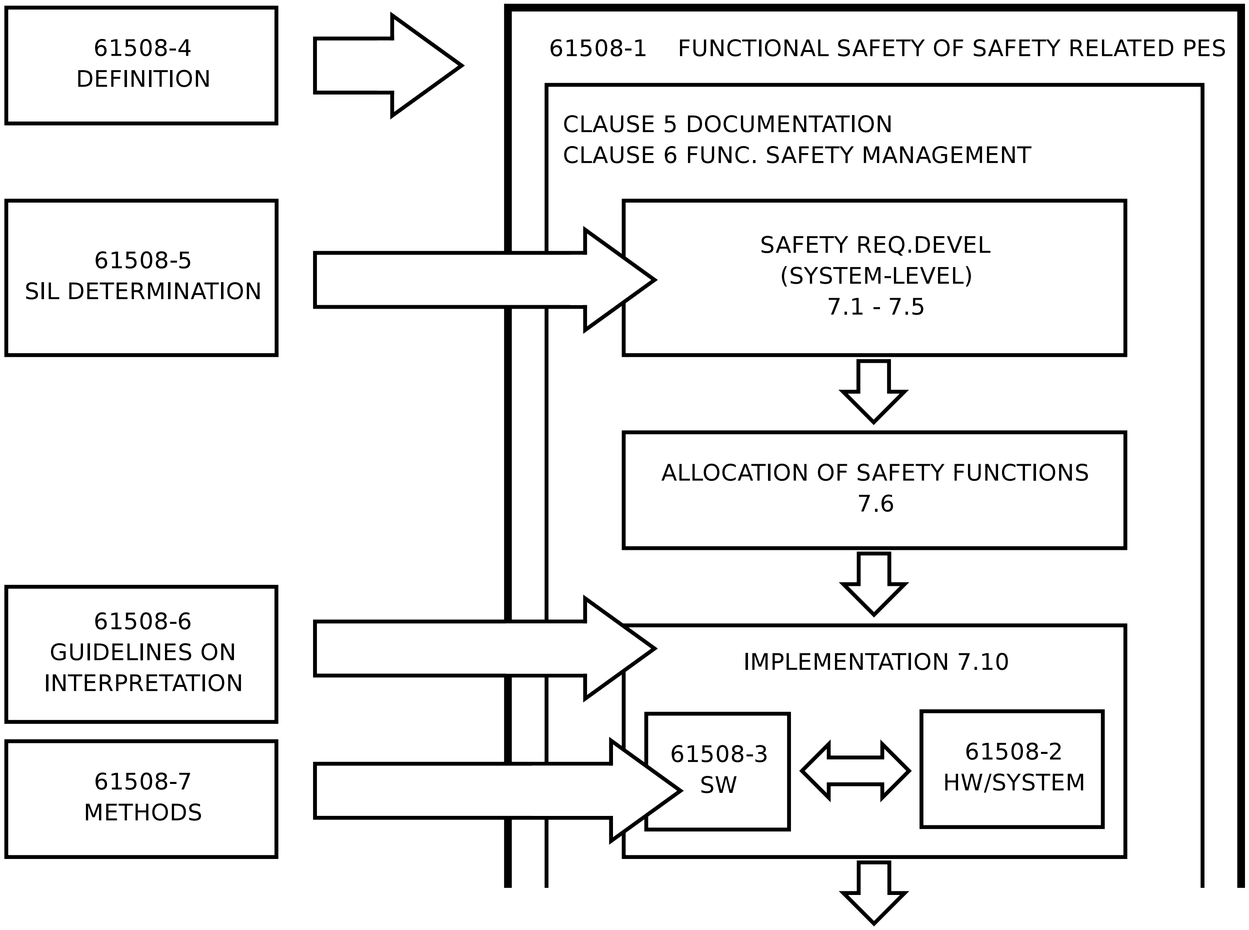
The architecture is an essential component in SW safety.

61508 Structure (SW)

- 61508-4 Definitions (global)
- Clause 5/6 Doc/Management (global)
- 7.1-7.5 Development of Requirements (<- Part 5)
- 7.6 Allocation of safety functions
- 7.10 Implementation -> 61508-2,61508-3
- 61508-6 guide to application of 61508-2/3
- 61508-7 Methods

61508 is not a simple standard as it has strong horizontal and vertical linkingof clauses.

61508 Structure (SW)



61508-4
DEFINITION

61508-5
SIL DETERMINATION

61508-6
GUIDELINES ON
INTERPRETATION

61508-7
METHODS

61508-1 FUNCTIONAL SAFETY OF SAFETY RELATED PES

CLAUSE 5 DOCUMENTATION
CLAUSE 6 FUNC. SAFETY MANAGEMENT

SAFETY REQ.DEVEL
(SYSTEM-LEVEL)
7.1 - 7.5

ALLOCATION OF SAFETY FUNCTIONS
7.6

IMPLEMENTATION 7.10

61508-3
SW

61508-2
HW/SYSTEM

61508-3 Overview

Functional Safety of software for E/E/PES safety related systems

- tightly coupled to 61508-2
- Software is never maintained only modified
- anything using an OS or libraries qualifies as high-complexity
- *"Previously developed software"* == COTS

61508-4/5/6/7

- 4 - Definitions
- 5 - Guidance in assigning SIL
- 6 - Guidance in applying 61508-2, 61508-3
- 7 - Approved methodologies

61508 provides the methods necessary to handle COTS - notably
61508-6 Appendix E gives a COTS based SIL3 system concept overview !

Safety Management

- Problem: Failure -> Error -> Hazard
- 61508: Hazard indentification is the key to fundtional safety management in 61508 and derived standards.

The main worry of functional safety management is to preserve consistency - this is in my opinion the main reason for the dominance of the monolitic safety model.

SIL Safety Integrity Level

Descrete level (one out of a posible four) for specifying the safety integrity requirements of the safety functions to be allocated to the E/E/PE safety related systems, where safety integrity level 4 has the highest level of safety integrity and safety integrity level 1 has the lowest [61508-4 3.5.6]

- risk reduction needed to achive acceptable risk level
- defined for continuous and low demand mode
- Note not all derived standards define SIL4 !
- Safety Integrity Level can be assigned by quantitative or qualitative methods [61508-5].

Safety-related (component)

- Safety-related electronic control system:
  Electronic control system of a machine whos failure can result in the imediate increase of the risk(s) [62061 3.2.4]
- Subsystem:
  Entity of the top-level architectural design of the SRECS where a failure where a failure of any subsystem will result in the failure of a safety related control function. [62061 3.2.5]
- Module:
  routine, discrete component or a functional set of encapsulated routines or descrete components belonging together [61508-4 3.3.6]

Diversity

- Types of diversity
  - software: N-version programming
  - hardware: different HW-platforms
  - usage: diverseity of access
  - temporal: diversity of env-state
- level of diversity
  - specification (procedural vs. rule-driven)
  - Selection (i.e. libraries, servers)
  - implementation (i.e. languages)
  - integration (i.e. compiler, generators)

Diversity can address some of the COTS wories of 61508.

Proven-in-use

- Evidence - not specified clearly
- ”Standard or previously developed software” [61508-3 7.4.2.11]
- ”Increased confidence from use” [61508-3 A3 4b)]

Especially the deried standards have a quite varying interpretation of evidence and multiple terms for COTS.

Softwares role in safety

As far as practicalbe the design shall minimize the safety-related part of the software.[61508-3 7.4.2.6]

62061 constraints on software:

- Architectural constraints (HW) [62061 - 6.7.6]
- Probability of dangorous random HW failures [62061 - 6.7.8]
- Requiremenst for systematic SIL (SILCL) [62061 - 6.7.9]

62061 is a 61508 derivative for the Machine sector.

”previously developed software”

- if standard or previously developed software is to be used as part of the design then it shall be clearly indentified. [61508-3 7.4.2.11]
- The software suitability in satisfying the specification of requirements for software safety (see 7.2) shall be justified. [61508-3 7.4.2.11]
- Suitability shall be based uppon evidence of satisfactory opperation in a similar application ... [61508-3 7.4.2.11]
- or having been subject to the same verification and validation proceedures as would be expected for any newly developed software [61508-3 7.4.2.11]
- Constraints from the previous software environment shall be evaluated [61508-3 7.4.2.11]