# IEC 61508 Certification of a Code Generator

**Tilman Glötzner**

ETAS GmbH, Germany, Tilman.Gloetzner@etas.com

**Keywords:** IEC 61508, ASCET Code Generator, Certification.

## Abstract

This paper describes the certification of the ASCET code generator to a safety standard, the IEC 61508.

A certificate confirms the compliance of a product, a process, or a service to set of requirements by an independent authority. For this paper, the requirements are taken from the IEC 61508.

First, the underlying idea of the standard is explained. The motivation for a certification is highlighted. Then, the process and milestones of a certification project are described. The outlook discusses the future of the IEC61508 and its industry specific daughter standard, the ISO26262. The ISO 26262 is currently drafted.

## 1 Overview

In the first section, the concept of the safety norm IEC 61508 is explained. Then, the benefits for IEC 61508 certified tools is discussed. In the next section, the tool to be certified and the process of the certification are explained. The paper closes with an outlook.

## 2 The IEC 61508

This section gives an overview of the IEC 61508, its intention, and its application. The term "safety integrity level", around which the norm revolves, is explained.

### 2.1 Introduction

The IEC 61508 is a European standard that addresses the development of embedded systems working in a safety related context. It offers a systematic approach to manage the safety inherent to a system. It is important to understand that the standard deals with "safety". "Safety" is not synonymic with "reliability". For example, a system that enters the safe state by taking itself offline if an anomaly is detected may be safe. If the safe state is entered very often however, the users will not perceive the system as very reliable.

The idea of the IEC 61508 is to identify potential hazards, evaluate their impact and occurrence probability, and derive safety functions from the identified hazards. The safety functions are designed to reduce the risk of damage or casualties due to system failure. The IEC 61508 advocates good engineering practices. It is not the intention of the standard to increase the system complexity, but reduce the risk of harm to an acceptable level.

Safety functions may fail as well. Consequently, the system may still enter a dangerous state. This is described by safety integrity level (SIL). The SIL defines the acceptable probability of a system causing harm.

Depending on the SIL to be achieved the IEC 61508 demands a set of measures to be used as part of the development cycle when implementing the safety functions. The measures target the embedded system, its hardware, its software, the development processes, and the operational procedures. They range from product perception until decommissioning and disposal.

### 2.2 The gist of the IEC 61508

The IEC 61508 offers a systematic approach to risk as illustrated in figure 1.

The occurrence probability of a hazard as well as the impact in case of its manifestation is evaluated. From that, a measure representing the actual risk is derived. The safety integrity level (SIL) determines the acceptable risk. If the risk inherent to a system exceeds the acceptable risk, the IEC 61508 demands the implementation of one or more safety functions that bring down the actual risk to a tolerable level. The standard also defines process measures and methods to be applied during the implementation of the safety functions. These are mainly "best engineering practices" which not only aim at realizing the safety functions correctly, but also at raising the general quality of a product as well as the development and maintenance processes.

A safety function is defined by 2 requirements:

- The functional requirement describes what the safety function does and is derived from a hazard analysis.
- The safety integrity requirement specifies the likelihood of a safety function being performed correctly.

The safety integrity level (SIL) represents a safety performance requirement for a system. As illustrated in table 1, the SIL is arranged in 4 discrete levels. Each level is

assigned to a probability range for a dangerous system failure, i.e. a system fails resulting in damage of property or casualties.
The SIL level that is required for an application ultimately depends on society, and is normally reflected in the legislative system. For passenger cars typically SIL2 or SIL3 are applied.

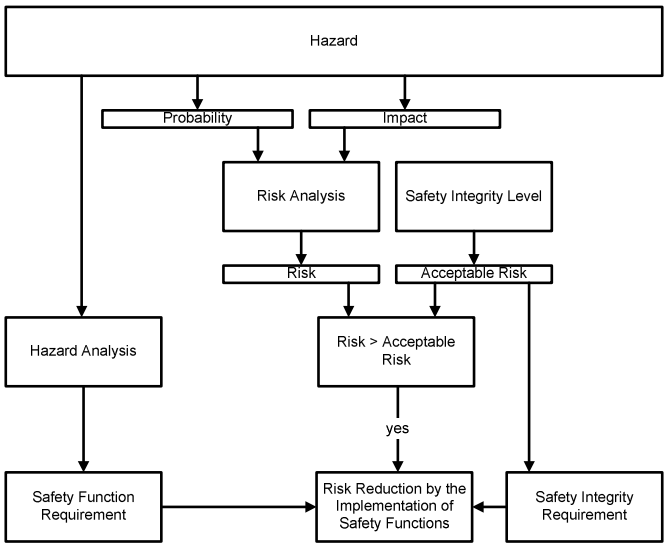

Figure 1: The gist of the IEC 61508.

| SIL | Probability of a dangerous system failure per hour | Acceptable impact |
|---|---|---|
| SIL 1 | $10^{-6} – 10^{-5}$ | Light injuries of one or several persons, damage or loss of property |
| SIL 2 | $10^{-7} – 10^{-6}$ | Heavy injuries of one or several persons, one fatality |
| SIL 3 | $10^{-8} – 10^{-7}$ | Several fatalities |
| SIL 4 | $10^{-9} – 10^{-8}$ | Many fatalities (catastrophe) |

Table 1: The Safety Integrity Level (SIL).

## 3 The rational for a certification

Developing a system or product to make it certifiable, and then certifying it, is a lengthy and costly venture. This section explains the motivation of OEMs to request certified tools or product components from their tool suppliers, and to certify the systems build from them.

### 3.1 Effort reduction of system certification by using certified components

By using certified components, part of the certification effort is shifted from the OEM to his supplier. The supplier of a component is normally in the position to conduct certification more easily, faster and more thoroughly than the customer. He has access to all necessary product and processes related documents, and an intimate knowledge about the implementation. It remains for the manufacturer to verify as part of his system certification that he is using the integrated components within the constraints specified by their respective certificates. This can for instance be documented in a safety manual that describes the tested configurations of a component.

A certified code generator, i.e., the transformation function from model to code, produces components that can more easily be certified than manual code, and reduces the certification effort of the system significantly.

- Code reviews of automatically generated code can be largely shifted from source code reviews to the abstract model level. The model level is concise and easier to understand for the human reviewer. The effort of source code review of large projects in particular is reduced drastically.

- Module- and module integration tests, static code analysis, and formal reviews of the source code become at least partially superfluous. (Of course they need still to be verified and validated on the model level.)

- An automatic code generator excludes manual implementation mistakes, and immediately improves the quality.

- Design or requirements changes impact the re-verification of the model only. The automatic code generation makes the code consistent to the model at the push of a button. The transformation from model to code is certified, and hence the generated source code does not need to be scrutinized for re-verification. The model represents the only source. This lowers the certification effort for changes and shortens the overall software development process.

It should be noted that a certified code generator does not automatically produce error free code.

- If the model representing a function is not up to its requirements, the generated code will reflect that.

- Timing issues for instance are not automatically resolved. The user will have to verify that the code meets its real-time requirements.

Consequently, the OEM is still responsible for overall product and its proper function. The user of a certified tool can however have good faith that the code generator will always produce the same code for a given model and set of translator options.

### 3.2 Quality increase of product and processes

The application of the IEC 61508 leads to improved products and processes. A certification represents an audit by an independent organization thereby enforcing the consequent application of the standard.

The IEC 61508 represents the state of the art in the development of safety relevant systems. A certification is a powerful and visible proof that everything that is technical possible and economically justifiably has been done to avert harm from the users of a product. It is expected that a certificate will support the side of an OEM in product liability lawsuits or in case of charges of culpable negligence.

### 3.3 Anticipation of the legislators

Unlike in the nuclear or aerospace industry, there was only a limited need for safety standards in the field of automotive electronics. Safety functions were implemented mechanically. Today, the trend to implement also safety relevant functions electronically using software is unbroken. This leads to increased system complexity and previously unseen modes of systemic failures. The legislators have not reacted yet. In Germany for instance, a qualification of an ESP function is not required to get a type certificate. This is expected to change however. The OEMs begin to recognize this, and try to shape the standards early by actively participating in working groups and committees. The ISO 26262, that is currently drafted and expected to be released in 2011, will be an application field specific safety standard for the automotive industry und use the IEC 61508 as mother norm. The current working draft of the ISO 26262 recognizes certification as measure to provide "increased evidence for a tool qualification".

## 4 Certification of ASCET's code generator

This section explains in brief the tool to be certified, ASCET, before it introduces the term "fit for purpose", and continues with a description of the certification process.

### 4.1 Introduction to ASCET

As shown in Figure 1 , the user of ASCET can define model components in different notations (block diagram, ESDL code, state machines, boolean tables, etc.), thereby giving him the flexibility to describe a function in the most suitable form. To support variant management and a small footprint of the generated code on the target, ASCET strictly separates function, data, and implementation.

When the code generation process is started, the model is transformed into an intermediate representation before it is expanded to C code. Typically, the user will first simulate the modelled controller function on a PC. He will start simulating in the physical domain, before later on rerunning the simulations under consideration of behavioral changes due to quantization effects imposed by the chosen data types. These paths through the tool are depicted in the left and middle branch of Figure 1, respectively. The right branch shows the tool chain to the embedded target. ECCO is an optimizer that adapts the C code to be generated for a particular target.

### 4.2 Fit for purpose

The IEC 61508 was drafted to regulate the development of safety relevant systems consisting from software, hardware, and optionally human users. ASCET as a software application is not safety relevant even though it generates safety relevant code. It is hence not reasonable to apply the full set of requirements. This is expressed by "SIL x -- fit for purpose". It expresses that a tool is suitable to support development projects up to safety integrity level "x".

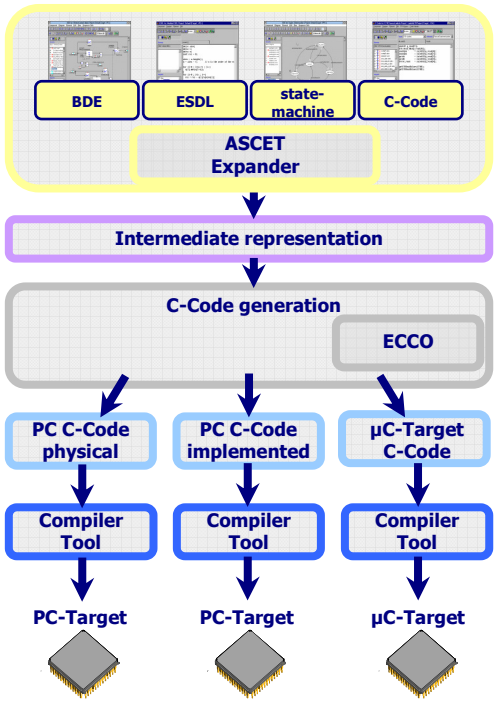


Figure 1: Functional overview of ASCET.

### 4.3 The Certification

A certification project can be broken down into the following sequence of work packages:

- Project Definition: The scope of the certification is defined together with the customer. Obviously, the transformation function model to c-code needs to be scrutinized. Most customers however have integrated ASCET into tool chains and customized it. Great care needs hence to be taken to include a meaningful set of product components as well as customer specific addition into the certification project.

- Preparation of Certification: The requirements that need to be fulfilled by ASCET and its development processes are extracted from the norm. This is depicted in figure 3. ASCET as an application is not safety relevant; it however generates C code that runs on a safety relevant

system. The IEC 61508 on the other hand addresses the development of a safety system. The independent certifying authority ETAS used, the TÜV Nord, took that in to account by weeding out meaningless requirements and including others specifically tailored to a code generator (=special function requirements). The TÜV Nord also set up an assessment plan detailing the criteria for a successful certification, and handed it over to ETAS.

- Preparation of Documentation: Process, development, and product documentation needs to be collected. Before they are handed over to the certifying authority they need to be brought into a suitable form to substantiate the fulfilment of the IEC 61508 requirements. This work package requires by far the most effort.

- Inspection & Audits: The documentation is then inspected by the certifying authority. Audits and interviews with developers, product/project managers, and testers are conducted.

- Evaluation: The certifying authority decides in favour or against a successful certification based on the findings gathered during the inspections & audits, and in accordance to the assessment plan

- Project Finish: The project finishes with the issue of the certificate. The certificate is valid for 2 years before it needs to be renewed.

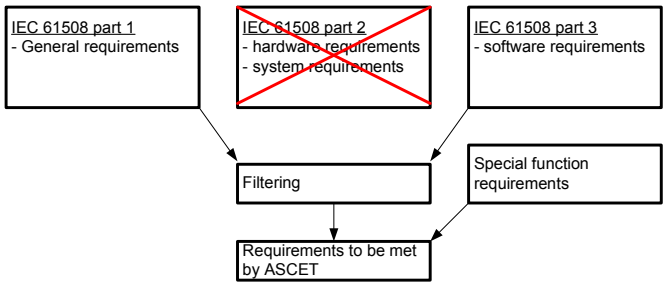

Figure 3: Compilation of requirements.

## 5 Outlook

The adoption of the IEC 61508 is currently spreading out with an increasing pace, especially in Asia. The OEMs include more and more safety relevant functions and implement those using ECUs. The necessity for systematic and risk based approaches such as that offered by IEC 61508 is obvious.
As there is currently no industry specific safety standard available for the automotive industry, the IEC 61508 is directly applied. The ISO 26262 that will rely on the IEC 61508 as mother standard is currently being drafted. This implies that IEC 61508 will not only gain influence as stand-alone standard but also as mother for application or industry specific norm.

The IEC 61508 has many things in common with quality or process oriented standards as CMM or ISO 9001. Compliance to the IEC 61508 should hence not be perceived as additional process effort but as tool to improve quality and efficiency of the development process and the product. Certified tools as ASCET supports this while reducing the effort at the same time.