

Verification and Validation According to IEC 61508: A Workflow to Facilitate the Development of High-Integrity Applications

Mirko Conrad, Jonathan Friedman, and Guido Sandmann

The MathWorks

Copyright © 2009 The MathWorks, Inc. Published by SAE International with permission.

ABSTRACT

Model-Based Design with production code generation has been extensively utilized throughout the automotive software engineering community because of its ability to address complexity, productivity, and quality challenges. With new applications such as lane departure warning or electromechanical steering, engineers have begun to consider Model-Based Design to develop embedded software for applications that need to comply with safety standards such as IEC 61508.

For in-vehicle applications, IEC 61508 is often considered state-of-the-art or generally accepted rules of technology (GART) for development of high-integrity software [6,11]. In order to demonstrate standards compliance, the objectives and recommendations outlined in IEC 61508-3 [8] must be mapped onto processes and tools for Model-Based Design.

This paper discusses a verification and validation workflow for developing in-vehicle software components which need to comply with IEC 61508-3 using Model-Based Design. It discusses tool support by using a Simulink based tool chain for Model-Based Design as an example.

INTRODUCTION

The increasing application of **embedded software** in commercial vehicles and passenger cars has resulted in a staggering complexity that has proven difficult to negotiate with conventional design approaches and processes.

More and more projects need to comply with external safety standards, because modern software-based electronic control units (ECUs) control or interact with critical systems such as brakes and steering. Examples include the electromechanical APA steering system [10] for the Volkswagen Tiguan [7].

Development approaches and processes play a decisive role in addressing the complexity, productivity, and quality challenges. Because of its ability to address these challenges, Model-Based Design has been extensively used throughout the automotive software engineering community. More recently, automotive OEMs and suppliers have begun to consider and adopt Model-Based Design for the development of embedded software for high-integrity applications that need to meet the IEC 61508 standard.

IEC 61508 is an international, industry-independent safety standard titled "Functional safety of electrical/electronic/programmable electronic safety-related systems". The seven parts of the standard (referred to as IEC 61508-1 to IEC 61508-7) were published between 1998 and 2000. IEC 61508-3 [8] "Software Requirements" is concerned with software development, verification, and validation.

Model-based design tools such as **Simulink®** and **Stateflow®** can be used throughout multiple phases of the software life cycle. Model-Based Design enables early simulation and testing of the functional behavior as well as automatic code generation for production systems.

Production code generation within Model-Based Design has replaced manual coding in various vehicle domains and generated code is increasingly being deployed in high-integrity applications.

The successful use of Model-Based Design for high-integrity applications requires extra consideration and rigor because additional requirements imposed by IEC 61508 have to be met. The standard does not cover advanced software development, verification, and validation technologies, such as Model-Based Design and code generation. If high-integrity software is developed, using such technologies, objectives and recommendations of the standard must be mapped to the processes and tools used.

To ensure quality and functional safety, models and generated code should be subjected to a combination of quality assurance measures. Further, compliance with the IEC 61508 safety standard needs to be demonstrated.

This paper discusses how Model-Based Design and associated verification and validation techniques can be leveraged to develop automotive applications that comply with IEC 61508. It covers the mapping of selected IEC 61508-3 objectives onto Model-Based Design.

A WORKFLOW FOR VERIFICATION AND VALIDATION OF MODELS AND GENERATED CODE

IEC 61508-3, like other standards, calls for **application-specific verification and validation** regardless of the tool chain and the development process used. When using Model-Based Design with production code generation, application-specific verification and validation need to answer the following questions:

- Does the model correctly implement its (textual) requirements?
- Does the object code to be deployed in the ECU correctly implement the model's behavior?

To facilitate a seamless use of Model-Based Design in the context of IEC 61508, a **reference workflow** that describes the verification and validation activities necessary for IEC 61508-3 compliance has been developed and audited by the TÜV SÜD certification authority.

Following the suggested division of the verification and validation task for applications developed using Model-Based Design and production code generation, the workflow would be divided into the following steps [1]:

1. Demonstrate that the model is correct and meets all requirements.

2. Show that the generated code is equivalent to the model.

The first step is **design verification**, which combines suitable verification and validation techniques at the model level. The second step is **code verification**.

DESIGN VERIFICATION — VERIFICATION AND VALIDATION AT THE MODEL LEVEL

The goal of design verification is to gain confidence in the model that is being used for production code generation. Design verification takes place at the model level, before the code is generated.

Following the spirit of IEC 61508-3, the design verification part comprises a combination of reviews, static analyses, and comprehensive functional testing activities at the model level [5]. These activities together help provide confidence that the design satisfies the associated requirements, does not contain unintended functionality and complies with defined modeling guidelines.

Requirements for design verification can be derived from IEC 61508-3 clauses 7.4.6 (code implementation), 7.4.7 (software module testing), and 7.4.8 (software integration testing).

Reviews and Static Analyses at the Model Level

Model subsystems considered as modules (model components) should be reviewed. If feasible, manual model reviews should be supported by automated static analyses of the model.

Modeling guidelines should be used and adherence with the guidelines should be assessed. Modeling constructs that are not suited or not recommended for production code generation should not be used.

Tool Support - Model reviews can be facilitated by using reports or interactive renditions (Web views) generated with Simulink Report Generator. Adherence to a modeling guideline can be partially enforced by using predefined or customized checks in Model Advisor [3], including MathWorks Automotive Advisory Board (MAAB) and IEC 61508 checks.

Module and Integration Testing at the Model Level

Model components should be functionally tested using systematically derived test vectors. The objective of module testing is to demonstrate that each model component performs its intended function and does not perform any unintended functions. The testing techniques mentioned in IEC 61508-3 table B.2 (dynamic analysis and testing) can be used to derive test vectors for the model.