Science of Computer Programming 90 (2014) 161–179

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Employing early model-based safety evaluation to iteratively derive E/E architecture design

V. Rupanov [a,*,1], C. Buckl [c], L. Fiege [b], M. Armbruster [b], A. Knoll [a], G. Spiegelberg [b]

[a] Institut für Informatik, Technische Universität München, Boltzmannstr. 3, 85748 Garching bei München, Germany
[b] Corporate Technology, Siemens AG, Otto-Hahn-Ring 6, 81739 München, Germany
[c] fortiss GmbH, Guerickestr. 25, 80805 München, Germany

HIGHLIGHTS

- A model-based methodology for early evaluation of design decisions is proposed.
- An analysis technique for ISO 26262 safety metrics is presented.
- Essential metamodels to support the methodology have been developed.
- Safety mechanism selection is performed based on model-based analysis results.
- A tool prototype implementing the methodology demonstrated.

ARTICLE INFO

Article history:
Received 16 October 2012
Received in revised form 15 September 2013
Accepted 15 October 2013
Available online 4 November 2013

Keywords:
Automotive systems
Embedded systems
Model-driven engineering
Quantitative safety analysis
ISO 26262

ABSTRACT

ISO 26262 addresses development of safe in-vehicle functions by specifying methods potentially used in the design and development lifecycle. It does not indicate what is sufficient and leaves room for interpretation. Yet the architects of electric/electronic systems need design boundaries to make decisions during architecture evolutionary design without adding a risk of late changes. Correct selection of safety mechanisms from alternatives at early design stages is vital for time-to-market of critical systems. In this paper we present and discuss an iterative architecture design and refinement process that is centered around ISO 26262 requirements and model-based analysis of safety-related metrics. This process simplifies identification of the most sensitive parts of the architecture, selection of the best suitable safety mechanisms to reduce thereby failure rate on the system level and improve the metrics defined by the standard. To support the defined process we present the metamodels that can be integrated with existing DSL (domain-specific language) frameworks to extend them with information supporting further extraction of fault propagation behavior. We provide a framework for architecture model analysis and selection of safety mechanisms. We provide details on the model-based toolset that has been developed to support the proposed analysis and synthesis methods, and demonstrate its application to analysis of a steer-by-wire system model and selection of safety mechanisms for it.

* Corresponding author.
E-mail addresses: vladimir.rupanov@fortiss.org, rupanov@in.tum.de (V. Rupanov), buckl@fortiss.org (C. Buckl), ludger.fiege@siemens.com (L. Fiege), michael.armbruster@siemens.com (M. Armbruster), knoll@in.tum.de (A. Knoll), gernot.spiegelberg@siemens.com (G. Spiegelberg).
[1] Currently with fortiss GmbH.

0167-6423/$ – see front matter
http://dx.doi.org/10.1016/j.scico.2013.10.005

1. Introduction

In today's cars, most of the functionality is implemented using a combination of hardware and software solutions. As more and more safety-critical functions heavily rely on software, safety of software systems becomes a hot topic. The international standard ISO 26262 [1, Part 1] addresses this topic by defining a design process and proposing safety mechanisms. It provides process guidelines for the whole lifecycle, but does not indicate what is sufficient and leaves room for interpretation [2]. An electrical/electronic (E/E) system architect has to account on numerous non-functional design aspects at the same time: real-time properties, safety, security, cost, etc., and needs guidance when making decisions during architecture evolution without adding a risk of late architecture changes. As late architectural changes are very expensive, it is extremely important to support early architectural decisions. An important part of these decisions covered by ISO 26262 is selection and configuration of safety measures.

To reduce the complexity of the E/E architectures, the car manufacturers intend to use generic hardware and software platforms executing mixed-criticality functions, such as AUTOSAR[2] in software domain. This enables software reuse, but introduces limitations to the system-level analysis for early assessment of functional and non-functional properties [3]. In particular, the platform design and configuration are separate from the system as a specific application, yet both need to be analyzed together for successful certification. Despite the Safety Element out of Context (SEooC) concept introduced in ISO 26262, analysis of quantitative properties of such systems remains a major challenge.

In this paper, we propose an approach for safety analysis and design guidance early in the design process. Although safety analysis has been applied in the automotive industry for decades, no common safety lifecycle was applied. ISO 26262 triggers a change in the development lifecycle that requires adaptation and alignment of numerous processes in E/E system development. The standardized lifecycle also acts as an enabling factor for extensive use of model-based tools for engineering support including automation of routine analysis steps, collection of data in a unified format, and reuse of that data in new developments. Another important trend is common acceptance of evolutionary design methodologies, first predicted by [4]. Evolutionary design needs strong support from the modeling side, which combines simplicity of domain specific models with possibility of changes or refinements during late design steps.

We propose to systematically evaluate the system with instantiated safety mechanisms in context of ISO 26262 requirements and assess design alternatives from different viewpoints (safety, performance, cost). The approach starts with a specification of the fault behavior of hardware component models. We relate safety mechanisms to the fault models and provide methods to evaluate achievable design metrics for software applications, running on the hardware platform. Presented design methodology supports evolution of E/E architecture (EEA) under a fixed guarantee that the target Automotive Safety Integrity Level (ASIL) can be reached. This is achieved by stepwise refinement and further combination of software component, hardware and safety mechanism models controlled by evaluation of ISO 26262 architectural metrics based on the combined model. The necessary steps for implementing our approach are discussed including the definition of metamodels, quantitative metrics to be calculated, and design and analysis workflows. Validating instantiation of these items in a tool is accomplished via an application example.

The rest of the paper is organized as follows. We discuss context and the range of design techniques and applications motivating our approach in Section 2. In Section 3 we provide details on our approach. First we gently introduce the intended design flow and describe the interaction of different concerns in a complex application. We provide details on system models and transformations required to perform analysis and rate an architecture. Instantiation of these ideas in a model-based tool is presented in Section 4 along with a validating application. The paper is concluded by a discussion of results and of related work in Section 5.

2. Better engineering today

In this section background information on model-driven development and safety-critical systems is provided, and an overview of state of the art in automotive industry is done. During the last 30 years, electronic systems have become widely used in cars, resulting in numerous advances in driving safety, comfort and controllability of vehicles. The use of computers in vehicle applications raises the question of adequate behavior of automotive systems, especially of those controlling or related to vital functions, such as braking, steering and longitudinal speed control. E/E systems already play a key role in implementation of assistance functions like electronic stability program (ESP) or electronic brakeforce distribution (EBD), and the likely introduction of Drive-by-Wire systems will lead to total reliance of driver safety on E/E systems [5].

Modern development methods, such as model-driven engineering, simplify design of systems through increased level of abstraction. However, the evaluation methods for architectures remain almost the same as with a manual approach. In [6] it has been mentioned that the most important direction, in which architecture modeling needs to be developed, is practicability of applications and automation of the modeling techniques.

[2] AUTOSAR: AUTomotive Open System ARchitecture, more details at: http://www.autosar.org.