Science of Computer Programming 118 (2016) 40–59

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Assertion-based monitoring in practice – Checking correctness of an automotive sensor interface

Thang Nguyen[a], Dejan Ničković[b,*]

[a] Infineon Technologies AG, Villach, Austria
[b] AIT Austrian Institute of Technology, Vienna, Austria

ARTICLE INFO

Article history:
Received 19 January 2015
Received in revised form 3 November 2015
Accepted 3 November 2015
Available online 10 November 2015

Keywords:
Assertion-based monitoring
Analog and mixed-signal verification
Case study
Temporal logic
Automotive airbag system

ABSTRACT

This paper provides an evaluation of the assertion-based monitoring technology for mixed-signal systems applied to a real-world case study from the automotive domain. We first motivate the case study by presenting the state-of-the-practice verification and validation work-flow typically used in the automotive industry. We identify the shortcomings of this work-flow, and propose a more rigorous and automated methodology based on monitoring correctness of simulated mixed signal designs with respect to Signal Temporal Logic (STL) assertions, which formalize the requirements from the design specification. We apply this assertion-based monitoring framework to check the correctness of a Distributed System Interface (DSI3) mixed-signal protocol implementation in a modern airbag system-on-chip application. We present all the relevant steps in our proposed work-flow and evaluate the results. We discuss potential benefits of the framework and identify its current shortcomings. Finally, we propose a number of future research directions based on the case study outcome.

1. Introduction

A modern car is a cyber-physical system-of-systems (CPSoS) that puts together a number of embedded elements that are often developed independently. The systems in a car are heterogeneous, combining digital controllers with analog sensors and actuators. They interact with their physical environment and are interconnected through the vehicle physics, as well as communication protocols. This results in complex interactions generating emergent behaviors that are not predictable in advance. Many components in a car, such as the airbag deployment and differential braking, are *safety critical*. Hence, correct system integration in the automotive domain is crucial to achieve high standards with respect to safety and security. For instance, the automotive standard ISO 26262 [1] obliges suppliers to provide sufficient evidence about their components to the regulatory bodies.

Due to the heterogeneity and the complexity of components and sub-systems in modern cars, verification and validation (V&V) poses a major challenge in the automotive domain and represents the main bottleneck in the design process. Verification by simulation and manual testing are the dominant methods used in the V&V practice of the automotive industry. However, these techniques have the weakness of being ad-hoc, inefficient and prone to human errors.

The research community has investigated a number of approaches that address V&V issues for mixed-signal systems. Formal verification of systems combining continuous and discrete dynamics has been mainly studied by the *hybrid systems* [2,3]

* Corresponding author.
E-mail addresses: Thang.Nguyen@infineon.com (T. Nguyen), dejan.nickovic@ait.ac.at (D. Ničković).

http://dx.doi.org/10.1016/j.scico.2015.11.002

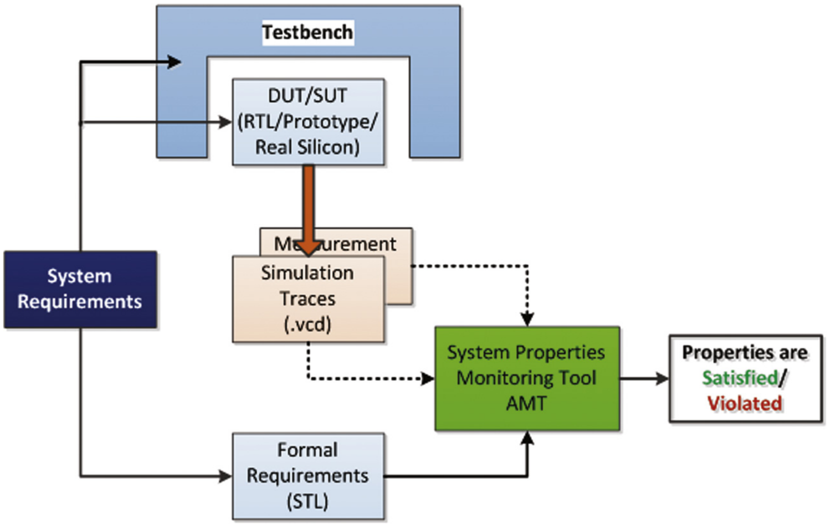
Testbench
DUT/SUT (RTL/Prototype/ Real Silicon)
Measurement
Simulation Traces (.vcd)
System Requirements
System Properties Monitoring Tool AMT
Properties are Satisfied/ Violated
Formal Requirements (STL)

Fig. 1. Assertion-based monitoring flow with STL assertion language and AMT tool.

community. It consists in computing over-approximations of reachable sets of states of the circuit, modeled as a hybrid automaton (differential equations with mode switching). Despite the important progress achieved in this research field in recent years [4], such techniques [5–9] still do not scale up to the size and complexity of transistor-level circuit models. In addition to hybrid system verification, there are other orthogonal analytical approaches to study similar systems. For instance, static analysis and abstract interpretation were used to develop a framework for inferring continuous time properties of systems consisting of synchronous components that interact by quasi-synchronous composition [10].

Assertion-based monitoring (also known as runtime verification in the software community) is a promising technology for verification of analog and mixed-signal (AMS) designs, i.e. designs that consist of interacting digital and analog components. It successfully exports some well-established ingredients from digital verification to the AMS domain, while retaining the relative simplicity and scalability of the simulation-based verification. In essence, assertion-based monitoring frameworks consist of an assertion language used to formalize the requirements that describe the correct interaction between analog and digital components, including timing constraints due to the communication delays. The formal assertions are then automatically translated into *monitors*, programs that read simulation traces of the design-under-test and check for the assertion satisfaction/violation.

Signal Temporal Logic (STL) [11,12] is an assertion language extending Linear Temporal logic (LTL) [13]. LTL enables declarative, formal and compact specification of reactive system requirements. Its original use was for evaluating sequences of states and events in digital systems. A typical property stated in temporal logic is `always (req -> eventually ack)`. This property says that it is always the case that a request `req` eventually triggers an acknowledgment `ack`. STL extends LTL to specification of properties involving both digital and real-valued variables defined over dense time. Offline monitoring of STL was implemented in the tool AMT [14]. The monitoring flow based on using STL for formalizing assertions and monitoring them with AMT is depicted in Fig. 1.

In this work, we apply the assertion-based monitoring framework from Fig. 1 to check the correctness of a sophisticated automotive sensor interface integration in a modern system-on-chip (SoC) airbag system, developed by Infineon Austria AG. The correct integration of the SoC with its sensor interface is specified in the Distributed System Interface (DSI3) protocol standard [15]. We present the work-flow of the case study in which we use STL to formalize DSI3 requirements and AMT tool to monitor the simulation traces. We evaluate the case study results and discuss the lessons that we learned regarding the applicability of this approach in industry.

This paper is an extension of the conference article [16] and contains several new contributions. The case study is extended with three more scenarios. The original scenario, in which the airbag SoC is configured with one sensor attached to its interface, is now complemented with the scenario in which four sensors are connected to the system. In addition, we also added two scenarios in which we inject faults to the connected sensor models that alter the expected communication between the airbag SoC and the sensors. We also present a complete formalization of the discovery mode requirements considered in [16]. Finally, we provide a more complete discussion about the related work and lessons learned based on the new observations that resulted from extending the case study.

2. State-of-the-practice in the V&V of automotive applications

In this section, we present the state-of-the-practice verification work-flow by the Power Train and Safety Department at Infineon Technology Austria AG, illustrated in Fig. 2. The work-flow describes the collaboration between the Tier-1 (system developer and integrator) and Tier-2 (hardware and software component developers) teams. The work-flow starts with the