Chapter 8

PREDICTABILITY IN REAL-TIME SYSTEM DEVELOPMENT

Jinfeng Huang, Jeroen Voeten, Oana Florescu, Piet van der Putten and Henk Corporaal
Faculty of Electrical Engineering Eindhoven University of Technology
5600MB Eindhoven, The Netherlands
J.Huang@tue.nl

Abstract The large gap existing between requirements and realizations has been a pertinacious problem in complex system design. This holds in particular for real-time systems with strict timing constraints and critical-safety requirements. Designers have to rely on a multi-step design process, where design decisions are made at different modelling levels. To ensure the effectiveness of this design process, predictability should be well-supported by design approaches, allowing designers to predict properties of future design outcomes based on existing design results. In this chapter, we first discuss the role of the semantics of design languages and investigated how they can support a predictable design process. Then, the deficiencies, w.r.t. predictability support, of existing design approaches for real-time systems are illustrated by an example. Finally, a predictable design approach for real-time systems is introduced to overcome this problem.

Keywords: Real-time, predictability, semantics, compositionality, composability

Introduction

The aim of real-time system design is to fill the gap between requirements and the realization. However, due to the continuous increase of the functional complexity of real-time systems, and because of stringent timing requirements they have to satisfy, the design gap has increased tremendously. Since traditional code-centric design approaches are obviously not capable of coping with this increasing complexity, designers have to resort to a multi-step design process, where the system is specified and analyzed at different levels of abstractions (see Figure 8.1). This design process usually involves requirement
capture, system modelling, and system synthesis. During requirement capture, the system is specified at the most abstract level, which defines the needs and constraints of the system. During system modelling, designers explore the design space at different abstraction levels, make design decisions through successive design steps and finally propose a proper design solution, which serves as a blueprint to synthesize a realization. During system synthesis, a model is transformed into a realization, which is expected to meet desired properties.

To smoothen the design process and improve productivity, consistency between design outcomes has to be maintained during each design step. In other words, predictability should be well-supported by design approaches, allowing designers to predict properties of future design outcomes based on existing design results.

The remainder of the paper is organized in four sections. In Section 1, We show that semantics of a design language plays an important role for the multi-step design process and has a direct impact on the support for predictability. In Section 2, we will briefly explain the deficiencies of existing approaches in supporting predictability during the design of real-time systems. To solve the problem presented in Section 2, we introduce a predictable design approach for real-time systems in Section 3. Section 4 concludes this chapter.

1. Semantics of design languages

Semantics of design languages has a direct impact on the thinking pattern of developers and the meaning of design outcomes. According to the different abstraction levels of design thoughts, three categories of design languages, requirement, modelling and implementation languages, are involved in the design process.