A Model-Driven Approach for Verification-based Development of Embedded Systems

Vincenzo Stoico

vincenzo.stoico@graduate.univaq.it,
University of L'Aquila,
L'Aquila, Italy

Abstract

The rising adoption of Embedded Systems leads the designers in dealing with stringent functional and non-functional requirements. An abstract representation (i.e., a model) of the system can be crucial to cope with the complexity of modern Embedded Systems specifications. Indeed, a system model allows the designers to concentrate on relevant properties and discard unnecessary details.

This work aims to the creation of a Model-Driven Electronic System-Level (ESL) development process that mixes formal e non-formal modeling activities to describe and verify system properties. The methodology begins from a high-level behavioral representation and refines it through several stages in a top-down fashion. Each step serves for the verification of a subgroup of system properties.

The proposed development process is made by the following steps: Co-Specification, Co-Verification, Co-Analysis, and Design Space Exploration (DSE). The Co-Specification goal is the conception of a high-level behavioral specification. At this stage, a few indispensable concerns permit us to build models: at system-level, verifiable, implementable, and representing concurrent flows. At the current state, our work introduces an extension of the UML Communication Diagram to express data and control flows. Indeed, the Extended Communication Diagram (ECD) includes operators to model several kinds of communication.

Co-Verification involves a first analysis of the model created during Co-Specification. The versatility of the ECD enables the mapping with existing Model of Computations (MoCs) to verify different properties. Currently, the ECD is mapped to Finite State Machines (FSMs) to verify Linear-Time Properties (LTPs) using the UPPAAL Model Checker. Thus, Co-Verification mainly checks properties related to system configuration and communication.

The Co-Analysis goal is to estimate system Non-Functional Properties (NFPs) and to check the fulfillment of non-functional requirements. For this goal, the model is equipped with performance domain-specific concerns and executable semantics. The Modeling and Analysis of Real-Time and Embedded systems (MARTE) profile helps to cover the semantics missing in the previous steps. The MARTE model is transformed into a Layered Queuing Network (LQN) performance model. Then, the estimations obtained from the LQN execution are annotated back in the MARTE model.

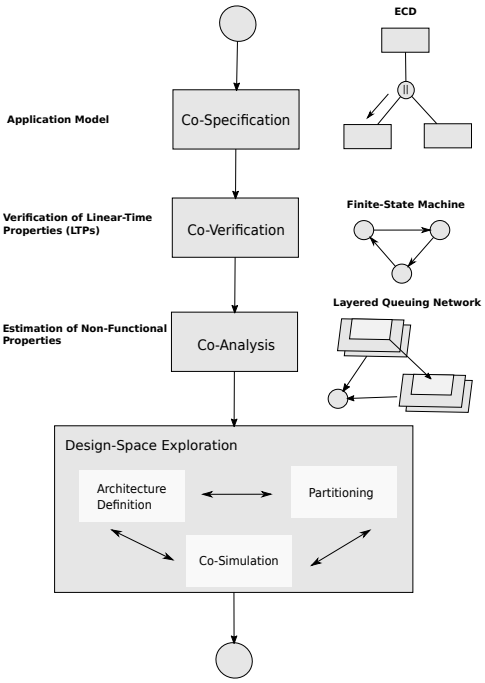

Fig. 1. Model-Driven Electronic System-Level Design Flow

DSE starts from Co-Analysis estimations, generates a possible hardware-software partitioning, and checks partitioning validity through simulation (i.e., Co-Simulation). The hardware components are taken from a repository of commercially available off-the-shelf (COTS) components. This process is iterative until the obtained solution satisfies system requirements. As a result, it is obtained a model embodying system application, the hardware platform, and their mapping.

Fig.1 summarizes the main steps of the MD-ESL flow. The work studies how to enrich Model-Driven Engineering of Embedded Systems with Formal Methods. Moreover, the research studies the effective integration of a Model-Driven front-end with existing Embedded Systems development environments. Finally, new insights are proposed at each step. Co-Specification employs an abstract notation mappable to several MoCs. This feature eases the analysis of different properties during Co-Verification. Instead, Co-Analysis shows how system models can be transformed into performance models to estimate NFPs. Finally, the DSE outputs a suitable partitioning driven by properties estimation and verification.