Chapter 10

UML-EXECUTABLE FUNCTIONAL MODELS OF ELECTRONIC SYSTEMS IN THE ViPERS VIRTUAL PROTOTYPING METHODOLOGY

P.F. Lister, V. Trignano, M.C. Bassett and P.L. Watten
Centre of VLSI and Computer Graphics,
University of Sussex,
Brighton, BN1 9QH, UK
Tel: +44 1273 678050
Fax: +44 1273 678030
P.F.Lister@sussex.ac.uk

Abstract

This paper presents the use of UML-Executable Functional Models (UML-EFM) in the context of the ViPERS virtual prototyping methodology [Lister et al., 2004a, Lister et al., 2004b] for System-on-Chip design. The concepts, the implementation and the experiments presented in this paper were developed at the University of Sussex (UoS) in the Centre of VLSI and Computer Graphics as part of an EU project [ViPERS]. The ViPERS methodology and its employment of the executable functional models have been developed to face the contemporary challenges of System-On-Chips by integrating key design methodologies with the graphical and interactive features of virtual prototyping. The fast evolution in silicon technology and its consequences on the market of hand held electronic products, is making the adoption of new design methodologies mandatory, with modern techniques for the design, development and manufacturing of consumer electronics. Executable functional models provide a means to simulate the target device in different phases of the design flow and analyse its requirements (behaviours, interfaces, etc), architecture (HW/SW partitioning) and finally its digital implementation. A key contribution includes the combination of an interactive 2D photorealistic model with its functional executable model implemented as a UML state machine; the experiment is applied to an RF home-based remote control used to control a cooking stack.
1. Introduction

A close look at the market of consumer electronics reveals that nowadays a significant slice of it is occupied by hand-held devices. The rapid advance in silicon technology is enabling a substantial increase in the number of transistors per chip [ITRS, 2003]. This growth in complexity is parallel to other phenomena, for example the shortened time to market, and the high competition among manufacturing companies. Designers and engineers are therefore facing the dilemma of having to produce highly technological and complex systems in a limited time. To reduce the gap between complexity and time to market new design methodologies are being proposed. The ViPERS methodology links key trends in SoC design with modern interactive and graphical features of virtual prototyping. At the heart of this methodology is the desire to test virtual prototypes of electronic products at different stages of the design, development and manufacturing processes.

The first step in the ViPERS methodology is the analysis phase and the consequent derivation of an UML-executable functional specification. It is clear from the research in the field of requirements and specification development [RUP] that the specification work is unlikely to be confined to the period before implementation begins. Determining accurate product requirements and specifications is a vital stage in the development of a commercially viable device and executable functional models can help extend the value and meaning of the requirements and specification phase to further ensure the validity of this work prior to implementation [Kimura and Verlag, 2002]. Hence there is a need to rapidly feed changes in the requirements into the implementation tool chain in an evolutionary way. It is common for a design house to be given a written specification for a prototype device. Often the specification is not complete enough for the first resultant prototype to be satisfactory to the client, resulting in some design iterations. If the design house were to build a virtual prototype or even several alternative schemes, the client can clarify the functional specification before any hardware or software is built. The virtual prototype is a form of communication and reference in addition to the functional specification and any other requirements of the design [Preece et al., 2002]. A key aspect being highlighted is to ensure that effort spent in the early product definition phase should be reused as much as possible in the later implementation and test of the device. Several methods of requirements gathering have been explored including traditional written reports and UML based tooling. UML [OMG, 2003] provides the means to document detailed requirements which can lead, with the aid of software tools such as Rational Rose RealTime and