A SysML and CLEAN Based Methodology for RISC Processor Micro-Architecture Design

Zakaria Lakhdara, LIRE Laboratory, Constantine 2 University, Constantine, Algeria
Salah Merniz, MISC Laboratory, Constantine 2 University, Constantine, Algeria

ABSTRACT

Nowadays, processor micro-architectures are becoming more and more complex. Consequently, designers increasingly need powerful abstraction and structuration mechanisms, as well as design methodologies that automatically and formally derive low-level concrete designs from high-level abstract ones. In this context, this paper proposes a methodology for RISC processor micro-architecture design. The proposed methodology uses mainly SysML to model both ISA and MA levels and the functional language CLEAN to describe them. Functional specifications in CLEAN are automatically generated from the ISA and MA models. These specifications, which are executable and formally verifiable, are used for simulation and verification. The proposed approach is validated by a case study that consists of designing the micro-architecture of MIPS processor. It shows how to easily model and generate CLEAN specifications describing the ISA and MA levels. It also illustrates, with multiple cases, how the generated specifications are used to simulate the MA. The results of the simulation phase prove the efficiency of the proposed modeling and code generation techniques.

Keywords: CLEAN, Formal Verification, Functional Languages, Micro-architecture Design, Modelling, RISC Processor, Simulation, SysML, UML

1. INTRODUCTION

The growing complexity of microprocessor micro-architectures necessitates new design methodologies, with automation and powerful mechanisms of abstraction and structuration. A promising and innovative solution, that emerged the last decade, was the introduction of the high-level modelling language UML (Unified Modelling Language) (UML, 2003) in hardware domain. This was motivated by its successful application in software engineering. Since UML was originally created for software systems, researchers had to adapt it to handle the hardware ones. They have created UML profiles such as: UML-SoC (UML-SoC, 2006), MARTE (MARTE, 2009), TUT (Kukkala et al., 2005), Gaspard (Ben Atitallah et al., 2007), and SysML (SysML, 2007) which extends UML with the appropriate constructs for hardware design. Another necessity was bridging the gap between UML high-level description and register transfer level (RTL). For this

DOI: 10.4018/IJERTCS.2015010105
reason, researchers have created some approaches and tools that transform UML models to the well-practiced Hardware Description Languages (HDL) such as SystemC (SystemC, 2005), VHDL (VHDL, 2009), and Verilog (Verilog, 1995).

The main drawback of these methodologies and approaches is the use of imperative HDLs, as they suffer from the lack of a well-defined semantic definition. Consequently; formally deriving low-level design refinements, that reflect the high-level requirements, becomes a hard task. Therefore, it is necessary first, to validate the generated HDL code before proceeding toward low-level implementations. While some approaches aim at validating the generated HDL code by translating it to formal models, most of the others use simulation-based methods to reach the same purpose. Both approaches suffer from many drawbacks. Formal models are mostly model checking-based. Hence, they still limited by the state explosion problem. On the other hand, simulation-based methods are insufficient to cope with growing complexity, which, according to Moore’s law, doubles almost every two years. At best, simulation methods, which also have the disadvantage of lengthening the time-to-market, can only reduce the number of design faults, but never certify design correctness (Merniz, 2008).

In this perspective, we propose a new methodology for RISC processor micro-architecture design. Our methodology uses SysML, to model the Instruction Set Architecture (ISA) and Micro-architecture (MA) levels; and the functional language CLEAN (CLEAN, 2000) as an HDL to describe them.

Our methodology involves:

- A modelling technique that enables modelling ISA and MA levels of a RISC processor, using three SysML diagrams;
- A code generation technique that transforms ISA and MA SysML models into their corresponding CLEAN specifications, using our code generator CleanSG.

A case study is presented as well, where our methodology is used to design the MIPS micro-architecture. First, we show how the ISA and the MA levels of this processor are modelled using our proposed technique. Thereafter, we show how functional specifications of the ISA and the MA, which are automatically generated, are used for simulation and for formal verification.

The remainder of this paper is organised as follows. Section 2 gives an overview of SysML and CLEAN, respectively. Section 3 discusses some related works. Section 4 presents the proposed methodology. A typical case study is given in section 5. Finally, section 6 concludes the paper.

2. BASIC CONCEPTS

2.1. SysML

SysML is a UML profile for system engineering applications. It reuses some UML 2.0 diagrams, extends others with new constructs (including the notion of block that replaces the notion of class), and adds new ones to model requirements and parametric constraints. The set of diagrams provided by SysML enables modelling the system behaviour and system structure, where the notion of block allows hierarchically developing and modularizing complex designs. Requirements and parametric constraints diagrams enable requirements engineering and performance analysis of such designs. Moreover, SysML supports model and data interchange via XML Meta-data Interchange (XMI).
Related Content

Performance Analysis of 3-Stage Cell Search Process in a WCDMA System
[www.igi-global.com/article/performance-analysis-stage-cell-search/1442?camid=4v1a](www.igi-global.com/article/performance-analysis-stage-cell-search/1442?camid=4v1a)

On Strengthening of Weak Algebraic Manipulation Detection Codes
[www.igi-global.com/article/on-strengthening-of-weak-algebraic-manipulation-detection-codes/168514?camid=4v1a](www.igi-global.com/article/on-strengthening-of-weak-algebraic-manipulation-detection-codes/168514?camid=4v1a)
Towards QoS-Inferred Internet
www.igi-global.com/chapter/towards-qos-inferred-internet/21696?camid=4v1a

Effect of Wireless Channels on the Performance of Ad Hoc Networks
www.igi-global.com/chapter/effect-wireless-channels-performance-hoc/28731?camid=4v1a