Editorial

ESL Design Methodology

Deming Chen,¹ Kiyoun Choi,² Philippe Coussy,³ Yuan Xie,⁴ and Zhiru Zhang⁵

¹ Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

² School of Electrical Engineering, Seoul National University, Seoul 151-742, Republic of Korea

³ Department of Sciences and Techniques, Lab-STICC, Université de Bretagne-Sud, Lorient 56321, Cedex, France

⁴ Department of Computer Science and Engineering, Pennsylvania State University at University Park, University Park, PA 16802-1294, USA

⁵ High-level Synthesis Department, Xilinx Inc., San Jose, CA 95124, USA

Correspondence should be addressed to Deming Chen, dchen@illinois.edu

Received 15 May 2012; Accepted 15 May 2012

Copyright © 2012 Deming Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

ESL (*electronic system level*) design is an emerging design methodology that allows designers to work at higher levels of abstraction than typically supported by register transfer level (RTL) descriptions. Its growth has been driven by the continuing complexity of IC design, which has made RTL implementation less efficient.

ESL methodologies hold the promise of dramatically improving design productivity by accepting designs written in high-level languages such as C, System C, C++, and MATLAB, and so forth, and implementing the function straight into hardware. Designers can also leverage ESL to optimize performance and power by converting compute intensive functions into customized cores in System-on-Chip (SoC) designs or FPGAs. It can also support early embedded-software development, architectural modeling, and functional verification.

ESL has been predicted to grow in both user base and revenue steadily in the coming decade. Meanwhile, the design challenges in ESL remain. Some important research challenges include effective hardware/software partitioning and co-design, high-quality high-level synthesis, seamless system IP integration, accurate and fast performance/power modeling, and efficient debugging and verification, and so forth.

With the invitation of *Journal of Electrical and Computer Engineering* of the Hindawi Publishing Corporation, we started the effort of putting together a special issue on ESL design methodology. After call for papers, we received submissions from around the globe, and after a careful review and selection procedure, eight papers are accepted into this

special issue. These papers cover a wide range of important topics for ESL with rich content and compelling experimental results. We introduce the summaries of these papers next. They are categorized into four sections: high-level synthesis, modeling, processor synthesis and hardware/software co-design, and design for error resilience.

2. High-Level Synthesis

In the paper “*Parametric yield-driven resource binding in high-level synthesis with multi-Vth/Vdd library and device sizing*” Y. Chen et al. demonstrated that the increasing impact of process variability on circuit performance and power requires the employment of statistical approaches in analyses and optimizations at all levels of design abstractions. This paper presents a variation-aware high-level synthesis method that integrates resource sharing with Vth/Vdd selection and device sizing to effectively reduce the power consumption under given timing yield constraint. Experimental results demonstrate significant power yield improvement over conventional worst-case deterministic techniques.

D. Menard et al. present in the paper “*High-level synthesis under fixed-point accuracy constraint*” a new method to integrate high level synthesis (HLS) and word-length optimisation (WLO). The proposed WLO approach is based on analytical fixed-point analysis to reduce the implementation cost of signal processing applications. Authors demonstrate that area savings can be obtained by iteratively performing WLO and HLS, in the case of a latency constrained application, by taking advantage of the interactions between these two processes.

In “*Highlevel synthesis: productivity, performance, and software constraints*” by Y. Liang et al., a study of HLS targeting FPGA in terms of performance, usability, and productivity was presented. For the study, the authors use an HLS tool called AutoPilot and a set of popular-embedded benchmark kernels. To evaluate the suitability of HLS on real-world applications, they also perform a case study using various stereo matching algorithms used in computer vision research. Through the study, they provide insights on current limitations of mapping general purpose software to hardware using HLS and some future directions for HLS tool development. They also provide several guidelines for hardware friendly software design.

3. Modeling

In “*Task-level data model for hardware synthesis based on concurrent collections*” by J. Cong et al., a task-level data model (TLDM), which can be used for task-level optimization in hardware synthesis for data processing applications, was proposed. The model is based on the Concurrent Collection model that can provide flexibility in task rescheduling. Polyhedral models are embedded in TLDM for concise expression of task instances, array accesses, and dependencies. The authors demonstrate examples to show the benefits of the proposed TLDM specification in modeling task level concurrency for hardware synthesis in heterogeneous platforms.

A. Barreteau et al. in the paper “*A state-based modeling approach for efficient performance evaluation of embedded system architectures at transaction level*” address the important topic of performance evaluation for SoC based on transaction-level modeling (TLM). The authors propose a generic execution model and a specific computation method to support hardware/software architecture evaluation. The benefits of the proposed approach are highlighted through two case studies.

4. Processor Synthesis and Hardware/Software Codesign

The paper “*Automated generation of custom processor core from c code*” by J. Trajkovic et al. presents a novel solution to constructing a processor core from a given application C code. The proposed methodology starts with an initial data path design by matching code properties to hardware elements and iteratively refines it under given design constraints. The experimental results show that the technique scales very well with the size of the C code, and demonstrate the efficiency of the technique on wide range of applications, from standard academic benchmarks to industrial size examples like the MP3 decoder.

The paper “*Hardware and software synthesis of heterogeneous systems from dataflow programs*” by G. Roquier et al. stresses that sequential programming model does not naturally expose potential parallelism to target heterogeneous platforms. Therefore, this work presents a design method that automatically generates hardware and software components and their interfaces, from a unique high-level description of the application, based on the dataflow

paradigm. The work targets heterogeneous architectures composed by reconfigurable hardware units and multicore processors. Experimental results using several video coding algorithms show the effectiveness of the approach both in terms of portability and scalability.

5. Design for Error Resilience

The paper “*Selectively fortifying reconfigurable computing device to achieve higher error resilience*” by M. Lin. et al. introduces the concept of Selectively Fortified Computing (SFC) for mission-critical applications with limited inherent error resilience. The SFC methodology differs from the conventional approaches that use static temporal and/or spatial redundancy and complicated error prediction or estimation techniques. It selectively allocates hardware redundancy for the key components of a design in order to maximize its overall error resilience. The experimental results from a 720P H.264/AVC encoder prototype implemented with a Virtex 5 device demonstrated the effectiveness of SFC operating under a wide range of error rates.

6. Concluding Remarks

ESL design area is a fast evolving field. We hope this special issue would provide a snapshot of the current research activities in ESL design area and offer useful references for researchers who are interested in this exciting field. Finally, we would like to thank all the 29 reviewers for this special issue wholeheartedly, whose effort has made this special issue successful.

Deming Chen
Kiyoun Choi
Philippe Coussy
Yuan Xie
Zhuru Zhang