

Editorial

Design Methods for DSP Systems

Markus Rupp,¹ Bernhard Wess,¹ and Shuvra S. Bhattacharyya²

¹*Institute of Communications and Radio Frequency Engineering, Vienna University of Technology, Gusshausstrasse 25/389, 1040 Vienna, Austria*

²*Department of Electrical & Computer Engineering, University of Maryland, College Park, MD 20742, USA*

Received 8 August 2005; Accepted 8 August 2005

Copyright © 2006 Markus Rupp 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.

Industrial implementations of DSP systems today require extreme complexity. Examples are wireless systems satisfying standards like WLAN or 3GPP, video components, or multimedia players. At the same time, often harsh constraints like low-power requirements burden the designer even more. Conventional methods for ASIC design are not sufficient any more to guarantee a fast conversion from initial concept to final product. In industry, the problem has been addressed by the wording design crisis or design gap. While this design gap exists in a complexity gap, that is, a difference between existing, available, and demanded complexity, there is also a productivity gap, that is, the difference between available complexity and how much we are able to efficiently convert into gate-level representations. This special issue intends to present recent solutions to such gaps addressing algorithmic design methods, algorithms for floating-to-fixed-point conversion, automatic DSP coding strategies, architectural exploration methods, hardware/software partitioning, as well as virtual and rapid prototyping.

We received 20 submissions from different fields and areas of expertise from which finally only 12 were accepted for publication. These 12 papers can be categorised into four groups: pure VLSI design methods, prototyping methods, experimental reports on FPGAs, and floating-to-fixed-point conversions.

Most activities in design methods are related to the final product. VLSI design methods intend to deal with high complexity in a rather short time. In this special issue, we present five contributions allowing to design complex VLSI designs in substantially lower time periods.

In “*Macrocell builder: IP-block-based design environment for high-throughput VLSI dedicated digital signal processing systems*”, N.-E. Zergainoh et al. present a design tool, called DSP macrocell builder, that generates SystemC regis-

ter transfer level architectures for VLSI signal processing systems from high-level representations as interconnections of intellectual property (IP) blocks. The development emphasizes extensive parameterization and component reuse to improve productivity and flexibility. Careful generation of control structures is also performed to manage delays and coordinate parallel execution. Effectiveness of the tool is demonstrated on a number of high-throughput signal processing applications.

In “*Multiple-clock cycle architecture for the VLSI design of a system for time-frequency analysis*”, Veselin N. Ivanović et al. present a streamlined architecture for time-frequency signal analysis. The architecture enables real-time analysis of a number of important time-frequency distributions. By providing for multiple-clock-cycle operation and resource sharing across the design in an efficient manner, the architecture achieves these features with relatively low hardware complexity. Results are given based on implementation of the architecture on field-programmable gate arrays, and a thorough comparison is given against a single-cycle implementation architecture.

In “*3D-SoftChip: a novel architecture for next-generation adaptive computing systems*”, C. Kim et al. present an architecture for real-time communication and signal processing through vertical integration of a configurable array processor subsystem and a switch subsystem. The proposed integration is achieved by means of an indium bump interconnection array to provide high interconnection bandwidth at relatively low levels of power dissipation. The paper motivates and develops the design of the proposed system architecture, along with its 2D subsystems and hierarchical interconnection network. Details on hardware/software codesign aspects of the proposed system are also discussed.

In “*Highly flexible multimode digital signal processing systems using adaptable components and controllers*”, V. V.

Kumar and J. Lach present a design methodology for signal processing systems. The targeted class of applications involves those that can be decomposed naturally into multiple application modes, where the different modes operate during nonoverlapping time intervals. The approach developed in the paper emphasizes supporting flexible application of reconfigurability in multimode signal processing architectures, including reconfigurability in datapath components, controllers, and interconnect, as well as both intra- and inter-mode reconfigurability. The approach is demonstrated through synthesis of multimode applications that are composed of various DSP benchmark subsystems.

In “*Rapid VLIW processor customization for signal processing applications using combinational hardware functions*,” R. R. Hoare et al. present a VLIW processor with multiple application-specific hardware functions for computationally intensive signal processing applications. The hardware functions share the register file with the processor to eliminate overhead by data movement. A design methodology including profiling, compiler transformations for combinational logic synthesis, and code restructuring is proposed to map algorithms written in C onto this architecture. Application speedups are reported for several signal processing benchmarks from the MediaBench suite.

A large amount of activities can currently be found in rapid prototyping where it is important to find feasible solutions to a challenging system design in rather short time. A final product may look different than the prototype but the prototype is intended to deliver a first hands-on experience of whether a proposal architectural solution is feasible at all. The prototype thus provides the designers with decisions for a final product while still giving them a chance to further explore parts of the design.

In “*Rapid prototyping for heterogeneous multicomponent systems: an MPEG-4 stream over a UMTS communication link*,” M. Raulet et al. present a rapid prototyping method using the SynDEx CAD tool, a half-automated method, to map algorithms that are typically specified in C onto various real-time platforms. Supported platforms are by Sundance and Pentek using a multitude of conventional DSPs and FPGAs. In order to support various platforms, means to describe hardware and software components as well as their communications links are provided in terms of SynDEx kernels. The communication kernel, for example, supports communication between the various functional units via shared RAMs. The efficiency of the proposed method is shown by a rather challenging example: an MPEG-4 stream is provided over a UMTS link.

A second contribution in this field entitled “*A fully automated environment for verification of virtual prototypes*”, P. Belanovic et al. present a computer-aided design tool for automated derivation and verification support of virtual prototypes. The targeted virtual prototypes include definitions of the hardware/software interfaces in the given system, which enables parallel development and improved validation support across hardware and software. The developed tool operates in the context of algorithmic specifications developed

through the COSSAP commercial design system for signal processing, and also in the context of target platforms based on the StarCore DSP. Retargetability to other algorithm development environments and target platforms is promising due to the general principles and modular architecture of the developed approach.

Many clever ideas to build prototypes based on FPGA were submitted. The three most interesting ones will be presented in this special issue. In “*FPGA-based reconfigurable measurement instruments with functionality defined by user*,” G.-R. Tsai and M.-C. Lin develop an approach using FPGAs to provide a framework for configurable measurement instruments, where the features and functionality of the instruments can be customized flexibly by the user. A hardware kernel for the configurable instrument approach is presented along with associated implementation considerations. Several examples are developed based on the proposed framework to illustrate the utility of the approach.

In “*FPGA implementation of a MUD based on cascade filters for a WCDMA system*”, Q.-T. Ho et al. present an FPGA-based implementation of a multiuser detector for WCDMA transmission systems. They exploit a serial interference structure in form of a cascade filter. Their design methodology strives for support of maximum number of users while reflecting limited FPGA resources and timing constraints. Elaborate resource utilisation studies for VIRTEX II and VIRTEX II Pro FPGAs from XILINX validate their results.

In “*A new pipelined systolic array-based architecture for matrix inversion in FPGAs with Kalman filter case study*,” A. Bigdeli et al. propose an optimized systolic array-based matrix inversion for implementation in FPGAs. The main advantage of their structure is the small logic resource consumption compared to other systolic arrays in the literature. The hardware complexity is reduced from $O(n^2)$ to $O(n)$ for inverting an $n \times n$ matrix. The new pipelined systolic array is used for rapid prototyping of a Kalman filter and compared with other implementations.

Floating-to-fixed-point conversion is an ongoing topic in system design. Although many concepts have been proposed over the years, there is hardly any tool support in commercial EDA products. In “*Floating-to-fixed-point conversion for digital signal processors*,” D. Menard et al. follow a different path than researchers have done before. Rather than minimizing signal-to-quantization noise energy, they minimize code execution time on a DSP for a given accuracy constraint. This method includes taking into account the DSP architectural structure. To evaluate the fixed-point accuracy, an analytical approach is used to reduce the optimisation time compared to existing methods.

In “*Optimum wordlength search using sensitivity information*,” K. Han and B. L. Evans propose a fast algorithm for searching for an optimum wordlength by trading off hardware complexity for arithmetic precision at the system outputs. The optimization is based on the complexity-and-distortion measure that combines hardware complexity information with propagated quantized precision loss. Two case studies demonstrate that the proposed method can find

optimum wordlengths in less time compared to local search strategies.

Markus Rupp
Bernhard Wess
Shuvra S. Bhattacharyya

Markus Rupp received his Dipl.-Ing. degree from 1988 at the University of Saarbruecken, Germany, and his Dr.-Ing. degree in 1993 from the Technische Universität Darmstadt, Germany. He is presently a Full Professor of digital signal processing in mobile communications at the Technical University of Vienna. He is an Associate Editor of IEEE Transactions on Signal Processing, of JASP EURASIP Journal of Applied Signal Processing, and of JES EURASIP Journal on Embedded Systems, and is elected AdCom Member of EURASIP. He authored and co-authored more than 180 papers and patents on adaptive filtering, wireless communications, and rapid prototyping.



Bernhard Wess received the Dipl. degree and the Ph.D. degree in electrical engineering from the University of Technology, Vienna in 1985 and 1993, respectively. He is currently the Head of the Electronic Department at the Vienna Institute of Technology and a lecturer at the University of Technology, Vienna. His current research interests are in the areas of code generation and optimization for digital signal processors and rapid prototyping for digital signal processing systems.



Shuvra S. Bhattacharyya is an Associate Professor in the Department of Electrical and Computer Engineering and the Institute for Advanced Computer Studies (UMIACS) at the University of Maryland, College Park. He is also an Affiliate Associate Professor in the Department of Computer Science. He is coauthor or coeditor of three books and the author or coauthor of more than 90 refereed technical articles. His research interests include VLSI signal processing, embedded software, and hardware/software codesign. He received the B.S. degree from the University of Wisconsin at Madison, and the Ph.D. degree from the University of California at Berkeley. He has held industrial positions as a researcher at the Hitachi America Semiconductor Research Laboratory (San Jose, Calif), and as a Compiler Developer at Kuck & Associates (Champaign, Ill).