Guest Editorial: Special Issue on Embedded Multicore Applications and Optimization

Yung-Chia Lin¹ · Jenq-Kuen Lee² · Francois Bodin³

Published online: 10 January 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

Embedded systems with multi-core designs are becoming increasingly important for signal processing, machine learning, and multimedia applications from both industry and academia in recent years. While embedded multi-core systems will look to play an important role ahead for application designs, many challenging problems remain to be resolved. Applications, programming models, compilers, API designs, architecture designs, and software tools all need to contribute to the advance of embedded multi-core computing for signal processing, pixel processing, machine learning, and multimedia applications. In this special issue, we aim to bring together researchers in the related areas to present the latest developments and technical solutions concerning various aspects of embedded multi-core computing for signal processing and applications.

This special issue consists of four papers related to embedded multi-core computing. In the first paper, “A Halide-based Synergistic Computing Framework for Heterogeneous Systems [1]”, by Liao et al., is focused on the computing framework that extends Halide to improve program execution performance. Optimizations are developed to improve performance by generating adequate the parallel code and eliminating extra memory copies. The

authors characterize and discuss the performance of two image processing programs and their framework on the heterogeneous platforms.

Second, the paper “GPUBlocks: GUI Programming Tool for CUDA and OpenCL [2]”, by Hwang et al., presents a GUI tool called GPUBlocks that can facilitate parallel programming on multicore computer systems. Programmers simply need to drag-n-drop blocks, fill the fields of the blocks, and connect them according to array or matrix computations that are specified by algorithms. GPUBlocks can then translate block-based code to CUDA or OpenCL programs. A couple of optimization constructs have also been offered for rapid program optimization.

For the third paper, “Rapid Hybrid Simulation for Exploring the Design Space of Signal Processors with Dynamic and Scalable Timing Models [3]”, authored by Yeh et al., presents a rapid hybrid emulation/simulation framework. This framework allows the user to execute full-blown system with application software and plug in emulators, simulators, and timing models for various components in the prototype system. In this system, with the just-in-time model selection mechanism it switches timing models dynamically.

The last paper in this special issue, entitled “Support OpenCL 2.0 Compiler on LLVM for PTX Simulators [4]” and authored by Yang et al., revises the low level virtual machine (LLVM) compiler to extend it to OpenCL 2.0 features to allow using the community GPGPU-Sim simulator.

The above four papers in this special issue cover language and simulation layers for embedded multi-core computing.. These papers provide frontier information related to heterogeneous computing, embedded compilers, and embedded multi-core programming tools.

Acknowledgements The work is supported in part by research grants from Taiwan MOST and Mediatek.

✉ Jenq-Kuen Lee
jklee@cs.nthu.edu.tw

Yung-Chia Lin
yungchia.lin@mediatek.com

Francois Bodin
francois.bodin@irisa.fr

¹ MediaTek USA Inc. (Woburn), Woburn, MA 01801, USA

² National Tsing-Hua University, Hsinchu 300, Taiwan

³ Institut de Recherche en Informatique et Systèmes Aléatoires, Campus de Beaulieu, 35042 Rennes, Rennes Cedex, France

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Liao, S-W, Kuang, S-Y, Kao, C-L and Tu, C-H. (2019). A Halide-based Synergistic Computing Framework for Heterogeneous Systems, *Journal of Signal Processing Systems*, this issue.
2. Hwang, Y-S, Lin, H-H, Pai, S-H, and Tu, C-H. (2019). GPUBlocks: GUI Programming Tool for CUDA and OpenCL, *Journal of Signal Processing Systems*, this issue.
3. Yeh, C-W, Tu, C-H and Hung, S-H. (2019). Rapid Hybrid Simulation for Exploring the Design Space of Signal Processors with Dynamic and Scalable Timing Models, *Journal of Signal Processing Systems*, this issue.
4. Yang, C-C, Wang, S-C, Hsu, M-Y, Chang, Y-M, Hwang, Y-S and Lee, J-K. (2019). Support OpenCL 2.0 Compiler on LLVM for PTX Simulators, *Journal of Signal Processing Systems*, this issue.

Yung-Chia Lin, Ph.D., received his Ph.D. in computer science and B.S. degree in physics from National Tsing-Hua University. He is working at MediaTek, Inc. for more than 11 years, and specializes in optimizing compiler development and DSP processor architecture design. His research interests include LLVM as well as other general compiler technologies, parallel programming models, modern processor architectures, and all emerging technologies in embedded systems.

Jenq Kuen Lee received the B.S. degree in computer science from National Taiwan University in 1984. He received a Ph.D. in computer science from Indiana University in 1992, where he also received a M.S. (1991) in computer science. He was a key member of the team who developed the first version of the pC++ language and SIGMA system while at Indiana University. He was also a recipient of the most original paper award in ICPP '97 with the paper entitled "Data Distribution

Analysis and Optimization for Pointer-Based Distributed Programs". In 2005, he received Taiwan MOEA funding to lead a research team to develop compilers for PAC VLIW DSP processors with distributed register files by collaborating with ITRI STC. He is also a recipient of Google Research Award (Mountain View), 2009. He was also a director for Taiwan MOE ESW (embedded system software) consortium 2008–2013. In 2010, he received a Taiwan MOEA economic contribution award (Deep Plow Award) for his contribution in embedded compiler research. From Oct. 2015 to Oct. 2018, he participated in the new version of OpenCL proposals with Khronos OpenCL Next DSP Feature Set. His current research interests include optimizing compilers, AI framework compilers, embedded multicore compilers and systems, and computer architectures.

Prof François Bodin (male) is presently Professor at the University of Rennes I. He was the founder and CTO of the former company Caps-Enterprise specialized in programming tools for high performance computing (HCP). He is also at the origin of another company, Tocea that focuses on control of source code quality and refactoring. His research contributions include new approaches for exploiting high performance processors in scientific computing and in embedded

applications. He held various positions including Chairman of the Irisa computer science laboratory in Rennes. His activity currently focuses on the convergence between HPC and data technologies. He is the holder of the chair "Mobility in a sustainable city" awarded by the Rennes I Foundation.