



Guest Editor's Introduction

APPLICATION-SPECIFIC PROCESSORS

..... It is my pleasure to introduce this special issue on application-specific architectures. The papers in this issue are based on work presented at the Second International Workshop on Application-Specific Architectures (WASP 2), held last December in San Diego, California. WASP 2 took place in conjunction with the International Symposium on Microarchitecture (Micro-36). Application-specific processors and processing are a very active R&D area, both in academia and in industry. It allows design customization to better meet performance, cost, and energy requirements in many areas. This can be done in a fixed or a configurable way, depending on the design. There are already many application-specific processors in the field, but the industry needs more R&D to make designing such architectures—and tuning them to specific applications—faster and easier. The other organizers and I hope that some of the research presented at WASP 2 will help to achieve these improvements.

The workshop had 49 submissions; WASP 2 accepted nine as regular papers. The articles in this issue are from the highest-ranking papers, as selected by the referees and the special issue editor. These articles present novel approaches and ideas in addressing some of the major problems in this area.

The first article, “Cost-Sensitive Partitioning in an Architecture Synthesis System for Multicenter Processors,” addresses the prob-

lem of designing clustered architectures. These architectures are usually homogenous, but this work allows the creation of heterogeneous clusters. Designers can create such clusters in an application-specific way, defining each cluster’s design and capabilities. An approach is presented for doing so using a compiler to partition operations across clusters, lowering hardware cost without affecting performance.

Another article, “Transforming Binary Code for Low-Power Embedded Processors,” proposes the use of program transformations to reduce the energy consumption of buses to instruction memory and for registers in deep-submicron process technologies. For instance, a new compiler register assignment algorithm achieves a significant reduction in bus transitions for register indexes. Another technique transforms a binary program and provides hardware support to efficiently reduce the energy consumption of individual bus lines for optimized programs. The authors consider the issue of coupling in adjacent bits of a bus, a very important component of power dissipation for nanometer technology.

The next article, “High-Throughput Programmable Cryptocoprocessor,” describes a high-throughput, programmable security engine for use in networking applications. It is programmable with domain-specific instructions to support IPsec applications. The independent cryptocoprocessor runs encryption

Alex Veidenbaum
University of California,
Irvine

standard such as the Advanced Encryption Standard in electronic code book, cipher block chaining-message authentication code (CBC-MAC), counter, and CCM (a new mode that combines the counter and CBC-MAC modes) modes of operation. The authors have synthesized the design in a 0.18-micron CMOS technology and showed it to achieve a throughput of 3.43 Gbps at a 295-MHz clock frequency.

The fourth article, "A Novel Processor Architecture with Exact Tag-Free Pointers," proposes a new architecture to deal with garbage collection in embedded and real-time systems for languages such as Java. The problem for real-time systems is an unpredictable delay introduced by incremental garbage collection. The new processor can unambiguously identify pointers using additional instructions to handle pointers and separate data register and pointer register files. The new instructions include an *allocate object* instruction for creating new objects; separate load and store instructions for pointers; copy and compare pointers instructions; and so forth. The architecture uses additional exceptions for bounds checking and null pointers. Special attribute registers hold the attributes of an object to which the pointer register is referring. An attribute cache stores attributes; this cache is accessible by an additional pipeline stage after the memory stage. The new architecture has a low overhead for garbage collection, low-cost synchronization of collector and application programs, high robustness, and hard real-time capabilities.

The last article, "A Multilevel Computing Architecture for Embedded Multimedia Applications," targets the performance requirements of multimedia applications. It describes and evaluates a parallel SoC architecture to meet these requirements. The architecture targets task-level parallelism, which it exploits by using several processing units, all controlled by a single control processor. This gives rise to a natural programming model that simplifies the programmer's task. Several code transformations help to enhance parallelism. These transformations are based on well-known compiler techniques and are easily incorporated into a compiler for this architecture, thus relieving a programmer from explicit task-level programming. Simulations show the architecture to be very scalable.

I hope that this special issue will serve as a good introduction to the subject for those not very familiar with the growing field of application-specific processors. At the same time, it should be of interest to researchers and developers already working in this exciting and rapidly growing field.

MICRO

Acknowledgments

I acknowledge the hard work of Dr. Alex Orailoglu (UCSD), the workshop chair; the program committee; Peter Petrov (Maryland); and the reviewers who made both the workshop and this special issue possible.

Alex Veidenbaum is an associate professor in the Department of Computer Science at the University of California at Irvine. His research interests include computer architecture, compilers, and embedded systems. Veidenbaum has a PhD in computer science from the University of Illinois at Urbana-Champaign. He is a member of the IEEE Computer Society and the ACM.

Direct questions and comments about this issue to Alex Veidenbaum, Department of Computer Science, University of California at Irvine, Irvine, CA 92697-3425; alexv@ics.uci.edu.

Classified Advertising

SUBMISSION DETAILS: Rates are \$110.00 per column inch. Eight lines per column inch and average five typeset words per line. Send copy at least one month prior to publication date to: Marian Anderson, *IEEE Micro*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; (714) 821-8380; fax (714) 821-4010. Email: manderson@computer.org.

CALTECH. Postdoctoral research positions at Caltech's Center for Advanced Computing Research are open in innovative, high performance computer architecture. Ph.D. in computer science, computer engineering or equivalent required. See (www.cacr.caltech.edu/employment/aca-postdoc.html). Resume to: Susan Powell, spowell@cacr.caltech.edu