Guest Editorial

This special issue contains nine of the best papers from the 2002 IEEE International Conference on Application-specific Systems, Architectures, and Processors. The conference was held in San Jose, California on July 17–19, 2002. The papers for this special issue were selected based on the reviews of the conference papers and a further round of reviews of the full papers. These papers have been extended to contain results not reported in the original conference papers.

We have divided the papers into three groups. The first three papers deal with methodologies and tools for application-specific systems. The next three papers discuss high-performance, application-specific architectures. The last three papers present results on computer arithmetic for application-specific processors.

The first paper by A. Turjan, B. Kienhuis, and E. Deprettere presents a new technique to determine at compile time whether a FIFO buffer is sufficient for every producer/consumer pair of a Kahn Process Network. When additional memory is required, their technique provides an address generation and a lower bound on the amount of memory needed to perform the reordering. The technique facilitates the mapping of signal processing applications to parallel architectures. It has been incorporated into the Compaan compiler framework, which transforms DSP applications written in Matlab into Kahn Process Networks.

The second paper by H. Blume, H.T. Feldkaemper, and T.G. Noll introduces a model-based approach for design space exploration of heterogeneous system-on-chip architectures. Their approach uses cost functions to estimate implementation specific parameters, such as power dissipation, area, and throughput. Implementation parameters for a variety of important signal processing operations on FPGAs, ASICs, and programmable DSPs are derived and serve as the basis for the design space exploration models.

The third paper by A. Darté and G. Huard examines the complexity of array contraction, which is used to transform array variables to scalar variables within a loop. Array contraction saves memory and increases locality. The authors prove that when there are at least two fusion-preventing arcs, fusing loops so that as many arrays as possible can be contracted is strongly NP-complete. They also prove that shifting loops to contract as many arrays as possible is strongly NP-complete. Finally, they show how the problems of loop fusion and loop shifting can be solved exactly using integer linear programming, when the number of variables and constraints is sufficiently small.

The fourth paper by R. Chamberlain, M. Franklin, P. Krishnamurthy, and A. Mahajan presents the architecture of a multicomputer interconnection network that uses optical technology. The interconnect uses Vertical Cavity Surface Emitting Lasers (VCSELs) and flexible fiber image guides to implement a physical ring topology that is logically configured as a multiring. A discrete-event simulation model is used to examine interconnect performance issues including latency, throughput, fairness, and the impact of reconfigurability for a set of signal processing applications.

The fifth paper by J. Draper, J.T. Barrett, J. Sondeen, S. Mediratta, C.W. Kang, I. Kim, and G. Daglikoca presents the 32-bit scalar and 256-bit Wide Word processing units of the Data-Intensive Architecture (DIVA) system. The 32-bit scalar processing unit performs integer processing and includes several special-purpose instructions, while the 256-bit Wide Word processing unit performs fine-grain parallel operations on 8-, 16-, or 32-bit operands. The DIVA system employs Processor-In-Memory (PIM) chips as smart-memory coprocessors to achieve significant speedups for bandwidth-limited applications, such as multimedia applications and sparse-matrix computations.

The sixth paper by R.B. Lee and A.M. Fiskiran introduces the PLX instruction set architecture for multimedia processing. PLX has several unique features including a fully subword-parallel architecture, datapath scalability from 32-bit to 128-bit words, and a new form of predication for reducing conditional branches. PLX is designed for use in constrained environments, which require low-cost, low-power, and high-performance multimedia processing.

The performance of PLX processors is evaluated on important multimedia kernels and compared to the performance of other instruction set architectures.

The seventh paper by J.-A. Pineiro, M.D. Ercegovic, and J.D. Bruguera presents a high-radix digit-recurrence algorithm for computing the logarithm of a number. The algorithm uses a table look-up in the first iteration and selection by rounding for subsequent iterations. A sequential architecture that implements the algorithm is proposed and evaluated for a variety of operand sizes and radices. Certain implementations of the proposed algorithm are more than four times as fast as a redundant radix-2 implementation.

The eighth paper by N. Burgess introduces a variety of approaches for evaluating logarithmic-depth parallel prefix adders. These approaches include a delay model based on a slight modification to the Logical Effort methodology, a area model based on wire pitch, and a new parameter for determining cell count and whether a prefix tree exhibits idempotency. The delay and area models are used to evaluate the Flagged Prefix Adder, which supports absolute difference computations.

The ninth paper by P. Kornerup analyzes various 4-to-2 adders for multi-operand addition. These types of adders are often used to accumulate partial products in high-speed multipliers. The paper shows that several recently proposed techniques for 4-to-2 adders are theoretically equivalent, and require only minor differences in their implementation. It also shows that these techniques are equivalent to using standard 4-to-2 carry-save adders, except for a few signal inversions.

We are very grateful to all the authors who submitted papers to this special issue and to the reviewers who provided feedback on the papers.

Michael J. Schulte

Shuvra S. Bhattacharyya

Robert Schreiber

Guest Editors