Special Session on "Exposed Data Path Architectures: Recent Advances and Applications"

The concept of "exposed datapath" is used in conjunction with processor architectures where the programmer can control more fine grained details of the datapath than in "traditional" processor architectures. For example, Transport Triggered Architectures (TTA) expose the data transports between the register files and function units for the direct control of the programmer. Some architectures, while otherwise having a general purpose register based instruction-set, allow the programmer to explicitly route previously computed results from a function unit to the next one to implement register bypassing in software. The additional datapath control extends the instruction scheduling freedom, simplifies the control hardware logic, eases the customization of the datapath interconnection network, and increases the instruction-level parallelism scalability, with the obvious drawbacks of wider instruction words and added compilation complexity.

This special session presents recent work related to exposed datapath processors. A total of seven included articles address such topics as energy-efficiency, multi-core and data level parallelism from an exposed datapath point of view.

Parallel processing was a popular topic in the included papers. For example, the Loki architecture presents an 8-core tile that is intended as a building block for many-core processors. Loki exposes the on-chip interconnection network to the instruction set of the processor with interesting implications. The session also presents work that proposes a 128 processing element SIMD datapath with explicit bypassing, thus improves support for data level parallelism using exposed datapath techniques. The proposed solution boasts significant savings on energy consumption of registers and memories when compared to a traditional RISC processor.

Program compilation targeting exposed datapath architectures was another popular topic in the session. With the growing popularity of stream processing, one paper presents a source-code annotation-based programming solution for a dataflow-style exposed datapath processor. The dataflow-flavored processor can be adapted to a specific application by polymorphic register files. In addition, code generation issues related to supporting OpenCL on the aforementioned exposed datapath SIMD architecture was discussed in one paper.

The number of good quality papers submitted to the session was a positive surprise. It indicates that exposed datapath architectures are receiving increased attention from the academic community. It makes perfect sense because the idea of exposing more details of the architecture to the compiler goes well with the current hot topic of low power parallel computing, thus it is easy to predict that we will see increased activity in this topic in the coming years.

Jani Boutellier, University of Oulu, Finland
Pekka Jääskeläinen, Tampere University of Technology, Finland