



Computer Architecture Challenges

Rechnerarchitektur – Herausforderungen der nächsten Jahre

Theo Ungerer, University of Augsburg

Computer architecture developments have made an overwhelming progress within the last five decades driven by technology advances that are well approximated by Moore's law of doubling the density of transistors on a chip every 18–24 months. These advances allowed to develop high-performance PCs and workstations based on microprocessors to the point of current multi-core based PCs, servers, and embedded devices.

Processor architecture hit two different "walls" during this progress. First came the "memory wall" caused by the gap in the frequency increase of DRAM and processor chips and by microarchitecture techniques that allowed to utilize instruction level parallelism beyond the ability to load enough instructions and data from external memory. The memory wall led to sophisticated memory hierarchies with primary-level code and data caches, a large on-chip secondary cache, and sometimes even third-level caches.

Still the increase in the cycle rate continued until a few years ago. Intel predicted the 5 GHz microprocessor in 2004 when introducing its long pipeline in the Prescott microarchitecture of the Pentium IV. The SIA prognosis of 1998 suggested a 6 GHz cycle rate for 2008 and a 10 GHz cycle rate for the year 2011.

Unfortunately, the so-called "power wall" made these predictions obsolete. Higher cycle rates result in a higher thermal dissipation loss with potential overheating of the processor chip. That substantially slows down the increase of cycle rates, however, the doubling of the number of transistors on the chip still continues. Utilization of instruction level parallelism by more speculation and larger secondary level caches allow for less and less performance gain of sequential programs. The resort was found in multi-cores – integrating several CPUs on a single processor chip leading to a paradigm shift in processor architecture developments. The continuation of Moore's law is currently projected onto the number of cores on a processor chip. Starting from currently two to four cores of general-purpose microprocessors, the projection lets us expect to fit as many as 256 of such cores on a chip in ten years from now. Unfortunately, most of the programs are still sequential and a major challenge in software development arises to parallelize application programs. It remains an open question how much parallelism can be derived from high-volume relevant application programs such that the high costs of future processor chips will be redeemed. The question how far the parallelization of

application programs will push the number of cores of future processor chips can be denoted as "parallelism wall", which threatens to limit the number of cores of future processor chips.

Still design and programming of multi-cores and many-cores – the latter with potentially hundreds of heterogeneous cores on a chip – remain the largest challenge in processor architecture and software development. This issue contains several papers that cover future multi-core and many-core architectures.

The paper of Koen DeBosschere of Ghent University, the coordinator of the European Network-of-Excellence on "High Performance and Embedded Architectures and Compilers" (HiPEAC-2) on "Upcoming Computing Challenges" describes the current evolutions in technology and architecture and lists the major research challenges in computer architecture that need to be mastered in the coming years to further continue the exponential performance growth predicted by Moore's law.

Rainer Buchty and Wolfgang Karl of University of Karlsruhe focus on "Design Aspects of Self-organizing Heterogeneous Multi-core Architectures". The authors propose the "Digital On-demand Computing Organism for Real-time Systems DoDOrg", which is a research co-



operation of five research groups of the University of Karlsruhe that states an organic self-organizing system architecture approach with a tiled architecture consisting of reconfigurable hardware units. The concepts of system introspection and of self-aware memory are described in detail.

Jürgen Teich of University of Erlangen-Nuremberg proposes “*Invasive Algorithms and Architectures*” that allow a single-processor program to explore neighbor cores and copy itself to these cores in a phase of “inversion”, and then to execute in parallel on the available number of cores. After the parallel execution the program may perform a “retreat” and resume again sequential execution on a single processor core. An invasive algorithm is able to spread itself for parallel execution based on the availability of processing resources. Such self-organizing behavior is proposed for massively parallel many-cores beyond the year 2020.

The contribution of Stefan Döbrich and Christian Hochberger of the Technical University of Dresden is called “*Towards Dy-*

namic Software/Hardware Transformation in AMIDAR Processors”. They analyze that reconfigurable architectures cannot be dynamically adapted to the requirements of the running applications and fill the gap by proposing the AMIDAR concept (*Adaptive Microinstruction Driven Architecture*). Instructions are broken down to a set of tokens which are distributed to a set of (specialized) functional units (FU). The authors propose the synthesis of specialized FUs at runtime and show the automatic integration of a synthesized FU into the running processor.

Lastly, Sascha Uhrig of University of Augsburg and of jamuth Systems GmbH & Co. KG introduces by “*A Flexible Java-on-Chip Solution*” into the jamuth Java framework which provides a multithreaded Java processor as IP core for programmable logic devices (FPGAs), a runtime system, and the necessary offline tools. The topic of Java in embedded real-time systems was addressed by the Komodo project that was funded by the German Science Foundation (DFG) in the years 2001 till 2004 at the Uni-

versities of Augsburg and Karlsruhe. The results obtained by the Komodo project flow into the jamuth IP processor core. The paper demonstrates the design efforts necessary to transform an academic prototype into a commercial product.

Prof. Dr. Theo Ungerer



Prof. Dr. Theo Ungerer is Chair of Systems and Networking at the University of Augsburg, Germany. Since 2003 he is also scientific director of the Computing Center of the University of Augsburg. His research interests are in the areas of embedded processor architectures, embedded real-time systems, ubiquitous systems, and organic computing.

Address: Department of Computer Science, University of Augsburg, 86159 Augsburg, E-Mail: ungerer@informatik.uni-augsburg.de