Why Europe needs Reconfigurable Computing

(DRAFT) Reiner Hartenstein, TU Kaiserslautern, http://hartenstein.de

Increasing the architectural complexity and increasing the clock frequency of single-core microprocessors (uniprocessors) has come to an end (e. g. the intel Pentium 4 successor has been stopped). Instead multi-core microprocessor chips are emerging from the same vendors. However, this kind of classical parallelism does not scale well - a very expensive lesson we have already learnt from supercomputing community with massively increasing the number of processors when going cheap COTS (commodity off the shelf). With the the growing degree of parallelism, the programmer productivity goes down drastically (The Law of More). It is an illusion to believe, that scalability gets really better, when all these processors will be resident on a single chip. This will not get better, as long as the monopoly of the von Neumann mind set is not relieved, where the classical fundamental paradigm is based on concurrent sequential processes and message passing through shared memory, both being massively overhead-prone and memory-cycle-hungry (this effect is known as *the memory wall*).

The von Neumann paradigm is loosing its dominance. To-day, by far most compute power comes from non-von-Neumann accelerators attached to the CPU: a mixture of (a) hardwired accelerators, and (b) reconfigurable accelerators, like, for instance, FPGAs (field-programmable gate arrays). The CPU (central processing unit) looses this central role by becoming an auxiliary processing unit mainly to run legacy code. Much more MIPS than running software are running configware, although most engineers do not yet know how to spell „configware“.

In contrast to hardwired accelerators, FPGA-based accelerator implementations do not need very expensive specific silicon (avoiding mask cost etc.). For this reason, FPGAs are also used for rapid prototyping (we call this „emulation“), orders of magnitude faster than simulation - having a very important impact on design automation methodologies. This flexibility is also used for remote patches at the customer’s site, for dynamic reconfiguration at run time (e. g. in multi-standard multi media nomadic devices and in many other areas), for fault tolerance and remote repair (e. g. in space crafts). For providing spare parts, setting up an emulation logistic is important for products with a life times much longer than the life time of the microchip and its fab line (e. g. in automotive, aerospace, industrial, military etc.). Such an emulation methodology is also a means to reduce the number of controller types needed (e. g. in a car).

FPGAs have become mainstream years ago in embedded systems, and more recently also in scientific computing in every application: Reconfigurable Computing. This is a new computing paradigm based on configware instead of software. Configware is not instruction-stream-based and needs compilation methods fundamentally different from compiling software. Compared to software solutions speed-up factors up to 4 orders of magnitude have been obtained, due to a different form of parallelism, which is drastically less overhead-prone than classical parallelism from concurrency by communicating (von Neumann) sequential processes. Also major supercomputer vendors went to Reconfigurable Computing, where a highly welcome side effect of migration to FPGAs is the massive reduction of the equipment cost and of the electricity bill amount.

A less welcome side effect of this paradigm shift are educational deficits needing a training on the job, since typical CS or CE curricula ignore Reconfigurable Computing - still driving the dead road of the von-Neumann-only mind set [1]. For this reason, the EU should also support strategies to innovate curricula, which should be interleaved with efforts in developing better application development tools also taking into account language acceptance problems of different application domains. Because of the current tool disaster DARPA has organized end of April 2006 a special meeting to develop new strategies in tool development.

A computational density by 4 orders of magnitude higher than with a FPGAs is obtained by using a rDPA (reconfigurable DataPath Array) with rDPUs instead of CPUs. A DPU has no program counter and its operation is transport-triggered by the arrival of operand data. This Kress-Kung machine paradigm (the counter part of von Neumann) is based on free form large pipe networks of rDPUs (without memory wall & compilation is easy), but not on concurrent sequential processes. For software people this is much more easy to understand than FPGAs. Europe is leading in this area.

Europe plays a leading role in FPGA, Reconfigurable Computing and applications. In this area the European FPL annual conference series is the oldest and by far the largest in the world (2006 in Madrid: more than 400 submissions). In these areas Europe has a rich variety of firms and research institutes being highly qualified. This strength of Europe should be supported by the EU-funds. The EU should not lose its strength in FPGA-supported and Reconfigurable-Computing-supported embedded systems design to keep diversity and to help innovation, targeting a variety of markets and applications.

[1] R. Hartenstein: Why we need Reconfigurable Computing Education; http://hartenstein.de/RCedu.html