Generalization of the Systolic Array

Gradually wearing off. Currently the dominance of the basic computing paradigm is gradually wearing off with growing use of Reconfigurable Computing (RC) - bringing profound changes to the practice of both, scientific computing and ubiquitous embedded systems, as well as new promise of disruptive new horizons for affordable very high performance computing. Due to RC the desk-top personal supercomputer is near [3]. To obtain the payoff from RC we need a new understanding of computing and supercomputing. To bridge the translational gap, the software / configware chasm, we need to think outside the box.

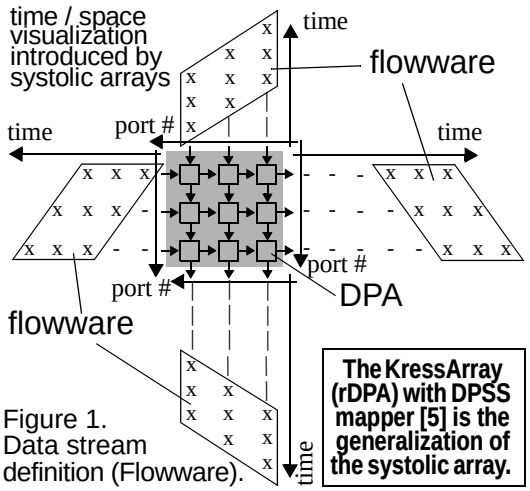

Figure 1. Data stream definition (Flowware).

Going toward the dual paradigm mind set is the current strong trend: the duality of the instruction-stream-based CPU paradigm, and its counterpart, the data-stream-based anti machine paradigm based on data counters instead of a program counter. The von Neumann paradigm using a program counter [1] [2] will not be explained here Since about the early 80ies one of the two the roots of the anti machine paradigm has been hidden inside the systolic array community [3] [4], mainly a group of mathematicians, who have nicely defined the concept of „data streams“, coming along with a beautiful visualization by time/space diagrams (see fig 1). Systolic arrays have been popularized in 1981 by H. T. Kung [4]. But for mathematicians a systolic array has only been a data path, a pipe network, but it has not been a computational machine, because the sequencer has been missing. But by the mathematicians, development of a methodology for generating these data streams at run time has been considered being somebody else's job. Refusing a transdisciplinary perspective this has been a typical result of the „formal“ reductionism having been fashionable at that time.

Mathematicians' paradigm trap. The classical systolic array suffered from a severe restriction. It could be used only for applications with strictly regular data dependencies. For a hammer many things in the world look like a nail. About 25 years ago for the mathematicians working on systolic arrays all applications looked like algebraic problems. From this point of view their algebraic synthesis algorithms generating a systolic architecture from a mathematical formula have been based only on linear projections, which yield only uniform arrays with linear pipes: usable only for applications regular data dependencies.

Generalization of the systolic array. Later Rainer Kress holding an EE degree discarded the mathematician's synthesis algorithms and used simulated annealing instead, for his DPSS (Data Path Synthesis System) [5]. This means the generalization of the systolic array: the KressArray [6], or super-systolic array (we may also call it Kress-Kung-Array), also supporting any wild forms of pipe networks, including any non-linear pipes like spiral, zigzag, branching and merging, and many other forms. Now reconfigurability makes sense. Fig. 2 shows how the sequencing part is added to the systolic array data paths, so that we get a complete computational machine. Instead of a program counter we have multiple data counters supporting parallelism of data streams. The GAG (generic address generator, [7] [9] [10]) is a data sequencer connected to a RAM block for data storage. GAG and RAM block are parts of the ASM (Auto-sequencing Memory), a generalization of the DMA (Direct Memory Access). The following paragraphs explain the machine principles and software / configware co-compilation techniques for the duality of machine paradigms.

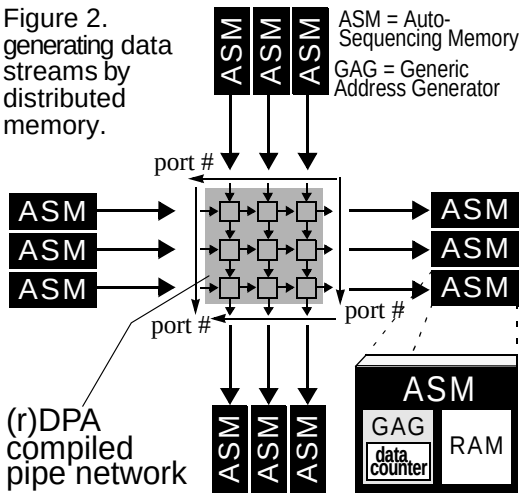

Figure 2. generating data streams by distributed memory.

Literature

[1] A. Burks, H. Goldstein, J. von Neumann: Preliminary discussion of the logical design of an electronic computing instrument; US Army Ordnance Departmentt Report, 1946.

[2] H. Goldstein, J. von Neumann, A. Burks: Report on the mathematical and logical aspects of an electronic computing instrument; Princeton IAS, 1947.

[3] N. Petkov: Systolic Parallel Processing; North-Holland; 1992

[4] H. T. Kung: Why Systolic Architectures? IEEE Computer 15(1): 37-46 (1982)

[5] R. Kress et al.: A Datapath Synthesis System (DPSS) for the Reconfigurable Datapath Architecture; Proc. ASP-DAC 1995

[6] http://kressarray.de

[7] M. Herz et al. (invited paper): Memory Organization for Data-Stream-based Reconfigurable Computing; 9th IEEE International Conference on Electronics, Circuits and Systems (ICECS), 15.-18. September 2002, Dubrovnik, Croatia

[8] http://flowware.net

[9] M. Herz, et al.: A Novel Sequencer Hardware for Application Specific Computing; ASAP'97

[10] M. Herz: High Performance Memory Communication Architectures for Coarse-grained Reconfigurable Computing Systems; Dissertation, Univ. Kaiserslautern, 2001, URL: [11]

[11] http://xputers.informatik.uni-kl.de/papers/publications/HerzDiss.html

Reiner Hartenstein
http://hartenstein.de