Synthesis and Domain-specific Optimization of KressArray-based Reconfigurable Computing Engines
Reiner Hartenstein, Michael Herz, Thomas Hoffmann, Ulrich Nageldinger
University of Kaiserslautern, Dept. of Computer Science, Erwin-Schroedinger-Strasse;
67663 Kaiserslautern, Germany; abakus@informatik.uni-kl.de; http://configware.de

We introduce a concept for reconfigurable hardware platforms along with an according CAD framework. The framework is not restricted to a specific architecture, but is capable of generating mappings for a number of members of the coarse grain reconfigurable KressArray architecture family, and similar platforms. Based on the multi-architecture CAD framework, a design-exploration environment has been set up to find a well suited architecture for a given application domain. We describe the architectural and generic principles of the KressArray family as well as the design flow for the determination of a domain-specific optimized KressArray architecture. Furthermore, we present the principles for the embedding of the KressArray into an accelerator engine, which can be used as a co-processor. Also, an overview about the tools in the CAD framework is given.

Synthesizing Full-Systolic Arrays For Matrix Product On Xilinx's XC4000(E, EX) FPGAs
Abdelkrim Kamel Oudjdja, Sabrina Titri & Mustapha Hamarlain*
CDTA/ Microelectronics and Robotics(*) laboratories
128 Chemin Mohamed Gacem El-Madania 16075, Algiers, Algeria; Email: a_oudjdja@hotmail.com

Matrix product is a compute bound problem that can be efficiently handled by elementary systolic algorithms. From a theoretical point of view, most of the algorithms are very simple and sometimes even trivial. But the task of designing efficient implementation on a fixed-connection network, such as on FPGA where resources are very limited, has been more demanding, however, and sometimes quite tedious. The objective of this paper is twofold: we first describe a full-systolic algorithm for matrix product that has the merit to require no preloading of input data into elementary processors (EPs) and generates output data only from boundary EPs. The resulting architecture can accept an uninterrupted stream of input data and produces an uninterrupted one with a latency of 2N-1 for NxN matrix product. Secondly, we present a methodology for synthesizing a family of very compact arrays on FPGA based essentially upon manual mapping at CLB level coupled with VHDL structural-level.

Virtualization of FPGA via Segmentation
William Fornaciari, Vincenzo Piuri, Luigi Ripamonti
Dipartimento di Elettronica e Informazione, Politecnico di Milano;
Piazza L. da Vinci 32, 20133 Milano, Italy; Email {fornacia, piuri}@elet.polimi.it

To cope with cost of the FPGA systems when very high performance is not the primary goal, is circuit partitioning with part multiplexing on a single FPGA. Dynamic loading, pagination, and segmentation (already used in operating systems) are techniques that allow for creating the Virtual FPGAs in multitasking computing environments. This paper deals with segmentation as an effective solution to load parts of the application, in which several concurrent tasks need to use the shared FPGA device. Circuits, that require resources exceeding the available ones, are physically decomposed by the designers into smaller partitions, sequentially loaded onto the FPGA, in order to recreate the computation as if all of them were loaded into the device at the same time. Experiments have been performed on a personal computer (under the NT4.0 O.S.) equipped with the board by H.O.T. Works based on the Xilinx XC6216 FPGA device. The FPGA support introduced in the O.S. consists of a library through which the applications require services to the virtual FPGA and the segmentation device driver handling the segmentation operations (FPGA configuration loading, state management, I/O management, segmentation fault handling). In the average, we have been able to achieve 75% of hardware execution, i.e. reconfiguration and loading overhead are kept under 25% of the overall computation.