Efficient Support of Hardware Debugging through FPGA Physical Design Partitioning

John Lach, William H. Mangione-Smith, Miodrag Potkonjak
56-125B Engineering IV University of California Los Angeles, CA 90095
Phone: (310) 794-1630 Fax: (310) 825-7928
jlach@icsl.ucla.edu, billms@icsl.ucla.edu, miodrag@cs.ucla.edu

Emulation based on FPGA technology, for functional verification, is becoming a widespread practice for digital integrated circuit designers. One drawback to this debugging and testing technique is the lengthy time spent in the back-end computer-aided design (CAD) tools for each design iteration. Even for small and localized debugging changes large portions of the design are typically re-placed-and-routed. We have developed a more fine-grained approach that allows the CAD tools to re-place-and-route only the portions of the design affected by the debugging changes. This goal is achieved by partitioning the design at the physical level into independent components. Design changes are localized to the affected components, allowing more limited re-placement-and-routing. The result is a shorter time between emulation and debugging iterations, and thus a shorter time-to-market for the design. Experiments on two large designs and six smaller MCNC benchmarks quantify the reduced back-end CAD tool time.

Exploiting Early Partial Reconfiguration of Run-Time Reconfigurable FPGAs in Embedded Systems Design

Byoungil Jeong, Sungjoo Yoo, Kiyoung Choi
School of Electrical Engineering
Seoul National University
{bijeong,ysj,kchoi}@poppy.snu.ac.kr

With run-time reconfigurable FPGAs, we can perform partial reconfiguration, which allows reconfiguration of a part of an FPGA while the other part is executing some functional computation. The partial reconfiguration of a function can be performed earlier than the time when the function is really needed. Such early partial reconfiguration can hide the reconfiguration time overhead more effectively. In this paper we incorporate the technique of early partial reconfiguration of FPGA into hardware- software partitioning for FPGA-based embedded systems design. We model the problem as an integer linear programming. In the model, we consider overlapping functional computation and partial reconfiguration. Experimental results show that the proposed method achieves 38% performance gain and 64% hardware cost reduction on the average over the conventional static reconfiguration method. It also achieves 13% performance gain and 9% hardware cost reduction on the average over the lazy partial reconfiguration method.

Extra-Dimensional Island-Style FPGAs

Herman Schmit
Carnegie Mellon University
herman@galant.ece.cmu.edu

This paper proposes modifications to standard island-style FPGAs that provide interconnect capable of scaling at the same rate as typical netlists, unlike traditionally tiled FPGAs. The proposal uses a logical third and fourth dimensions to create increasing wire density for increasing logic capacity. The additional dimensions are mapped to standard two-dimensional silicon. This innovation will increase the longevity of a given cell architecture, and reduce the cost of hardware, CAD tool and Intellectual Property (IP) redesign. In addition, extra-dimensional FPGA architectures provide a conceptual unification of standard FPGAs and time-multiplexed FPGAs, openning up new possibilities for prototyping of extremely large FPGAs and for IP transfer.