Flexibility and Circuit Overheads in Reconfigurable SIMD/MIMD Systems

S. Arrabi[1], D. Moore[2], L. Wang[1], K. Skadron[1], B.H. Calhoun[1], J. Lach[1], B.H. Meyer[3]
[1]University of Virginia, [2]University of Michigan, [3]McGill University
arrabi@virginia.edu

Dynamically reconfigurable SIMD/MIMD architectures made from simple cores have emerged to exploit diverse forms of parallelism in applications [1,2]. In this work, we investigate the circuit-level overhead and flexibility tradeoffs of such architectures through the design of a custom reconfigurable SIMD/MIMD system.

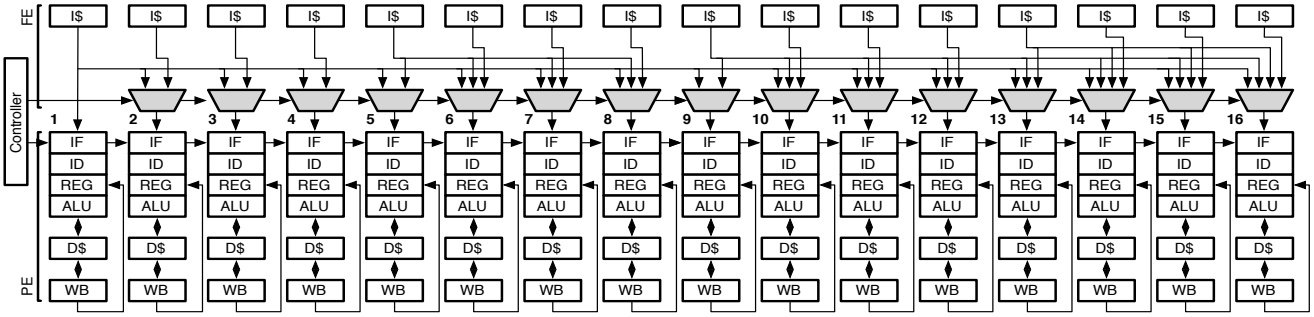
We partitioned a simple in-order processor, OpenRISC [3], into an FE (Front End) and a PE (Processing Element). Using a 16-core array of those partitioned blocks and a simple reconfigurable interconnect consisting mainly of multiplexers, we created a reconfigurable, variable-width SIMD and MIMD system. Using synthesis tools, circuit simulators, cache energy simulators (CACTI6.5 [4]), and an architecture simulator (gem5 [5]), we determined the best FE and PE partitioning as well as investigated the optimal degree of flexibility as a function of energy consumption (final design below).

This work focused on two main aspects of designing such systems – partitioning and flexibility. To compare partitioning strategies, we analyzed the energy consumption of possible FE/PE partitions (FE is I$ only, FE is I$ & IF, FE is I$ & IF & ID, or FE is I$ & IF & ID with duplicate ID in PE) by determining (1) the per-cycle energy consumption of different processor components, (2) the energy consumption communicating across wires of different length, (3) the number of signals driven from FEs to PEs (and back again) under different partitioning approaches, and (4) the utilization of each component across benchmarks. (1) and (2) were determined through circuit simulation in a commercial 90nm process, the synthesizable RTL of the processor provided (3), and gem5 was used to determine (4). The resulting analysis determined that sharing just the I$ between the different cores (i.e., FE is I$ only) provides the lowest total energy in SIMD mode despite the fact that IF and ID are duplicated in each PE. This counter-intuitive result is caused by the energy consumption difference between the IF and ID components (~8% of the total) and driving the extra signals between those stages and the EX units (~25%), demonstrating the importance of interconnect energy in reconfigurable SIMD/MIMD architectures.

To explore the overhead-flexibility tradeoff of different levels of system reconfigurability (represented in the table below), we compared the multiplexer-based reconfigurable interconnect systems in the optimal configuration for a particular application to a fixed system implementation. Assuming a perfect control system that switches to the optimal configuration for each benchmark, we found that the systems supporting 2, 3, 4, and 5 configurations achieve, on average over all benchmarks, runtime equal to 82%, 65%, 59%, and 59%, respectively, relative to serial execution, and total energy consumption (including the savings from sharing the I$ and the overhead of interconnect) of 99%, 81%, 67%, 70%, respectively, relative to 16-way MIMD. Adding flexibility therefore provides benefits to a point, but then the circuit overhead begins to outweigh the flexibility benefits. In our specific benchmark suite, supporting 4 configurations achieved similar performance as 5 configurations but with lower energy.

# Configurations	2	3	4	5
Configurations Supported	16-way SIMD; 16-way MIMD	16-way, 2 8-way SIMD; 16-way MIMD	16-way, 2 8-way, 4 4-way SIMD; 16-way MIMD	16-way, 2 8-way, 4 4-way, 8 2-way SIMD; 16-way MIMD
Multiplexers	15 2to1	8 2to1, 7 3to1	4 2to1, 8 3to1, 3 4to1	4 2to1, 6 3to1, 4 4to1, 1 5to1
Extra Interconnect	1 16x, 15 1x	1 16x, 1 8x, 14 1x	1 16x, 1 8x, 2 4x, 12 1x	1 16x, 18x, 2 4x, 4 2x, 8 1x

[1]K. Sankaralingam, *et al.*, "Exploiting ILP, TLP, and DLP with the polymorphous trips architecture," *IEEE Micro*, vol. 23, Dec. 2003.
[2] C. Lyuh, *et al*, "A novel reconfigurable processor using dynamically partitioned SIMD for multimedia applications," *ETRI Journal*, 2009.
[3] http://opencores.org/or1k/Main_Page
[4] N. Muralimanohar, *et al*., "Cacti 6.5: A tool to model large caches".
[5] N. Binkert, *et al.*, "The GEM5 simulator," *ACM COMP AR*, 2011.



Block diagram for the SIMD/MIMD reconfigurable system after the analysis of partitioning and overhead-flexibility tradeoffs

This work was supported in part by NSF grant no. CCF-0903471.