

WARP PROCESSOR FOR DYNAMIC HARDWARE/SOFTWARE PARTITIONING

BACKGROUND: Hardware/software partitioning is the process of dividing an application into software running on a microprocessor and hardware co-processors. While partitioning can yield significant improvements in speed and energy consumption, the design process is quite difficult. Hardware/software partitioning design requires a tool flow with the integration of profilers and the use of special compilers and synthesis tools, adding a level of complexity well beyond that of ordinary software design. Optimization of a partition in response to actual usage is also difficult, since static simulations cannot be done in real-time. Because of these problems, hardware/software partitioning is currently impractical for most microprocessor applications.

DESCRIPTION: University of California researchers have invented a warp processor, a microprocessor that allows the dynamic and transparent partitioning of an executing software's critical kernels to on-chip configurable logic. This dynamic approach allows the techniques associated with dynamic software optimization to be applied to hardware/software partitioning. In particular, the profiler, compiler, and

synthesis tool functions are entirely on-chip, so that warp processor partitioning does not require extra designer effort or disruption to standard tool flow.

APPLICATIONS: Almost any kind of microprocessor-based application could enjoy the benefits of this warp processor technology. Thus, the potential impact of this invention could be very far reaching, touching everything from mainframe computers to relatively simple consumer electronic items like TV set-top boxes.

ADVANTAGES: The dynamic partitioning of the UC warp processors offers considerable performance speedups (figure 1) and reduced energy consumption (figure 2) over their corresponding embedded benchmarks, comparable to that offered by traditional hardware/software partitioning. However, dynamic partitioning has the further advantage of being completely transparent, allowing a designer to gain the benefits of partitioning while writing a regular software application using standard software tools and tool flow. Dynamic partitioning can also adapt to an application's actual usage in real-time, eliminating the need for optimization via cumbersome static simulations.

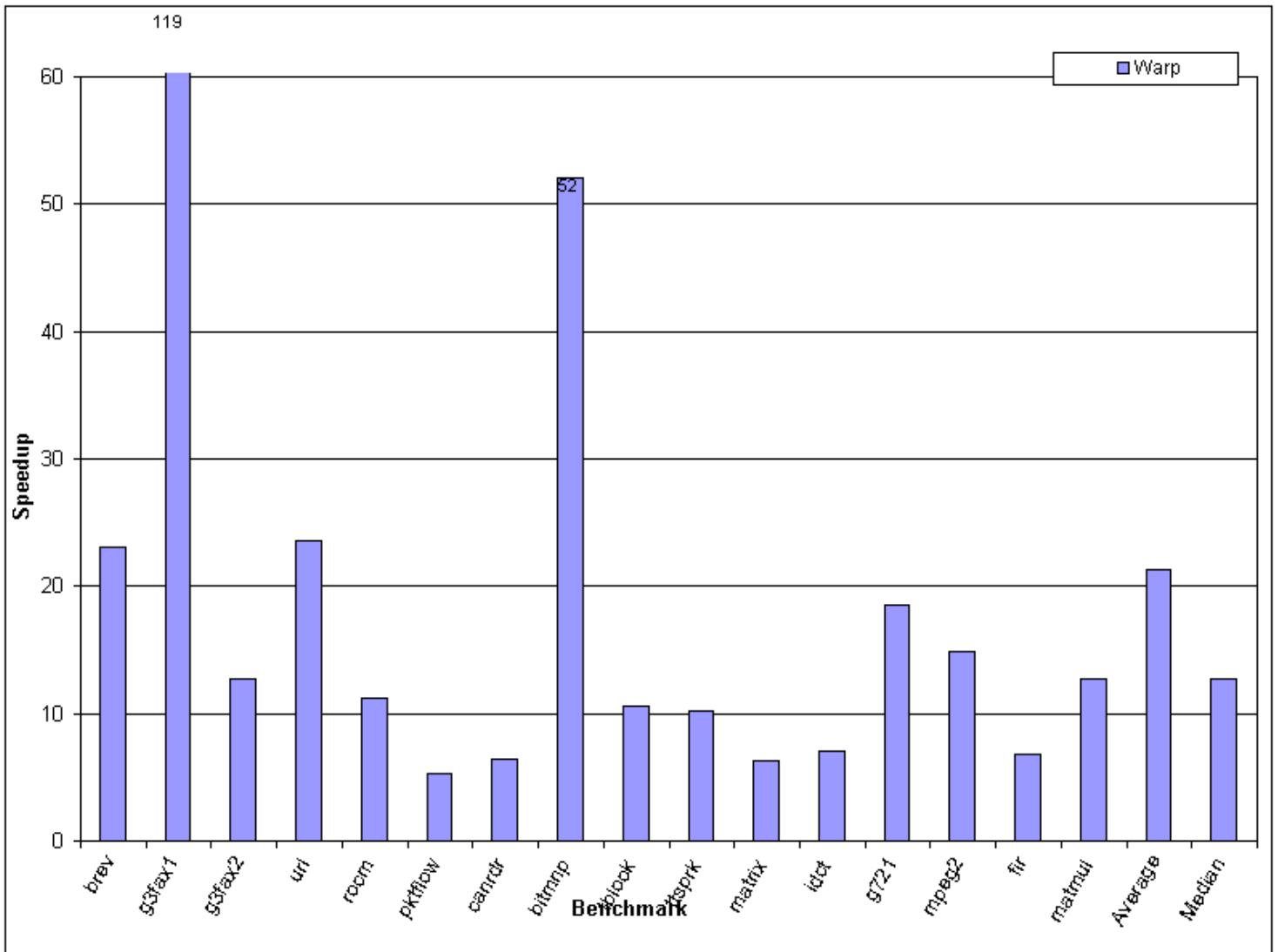


FIGURE 1—Performance speedups made possible by the UC warp processor for various embedded benchmarks.

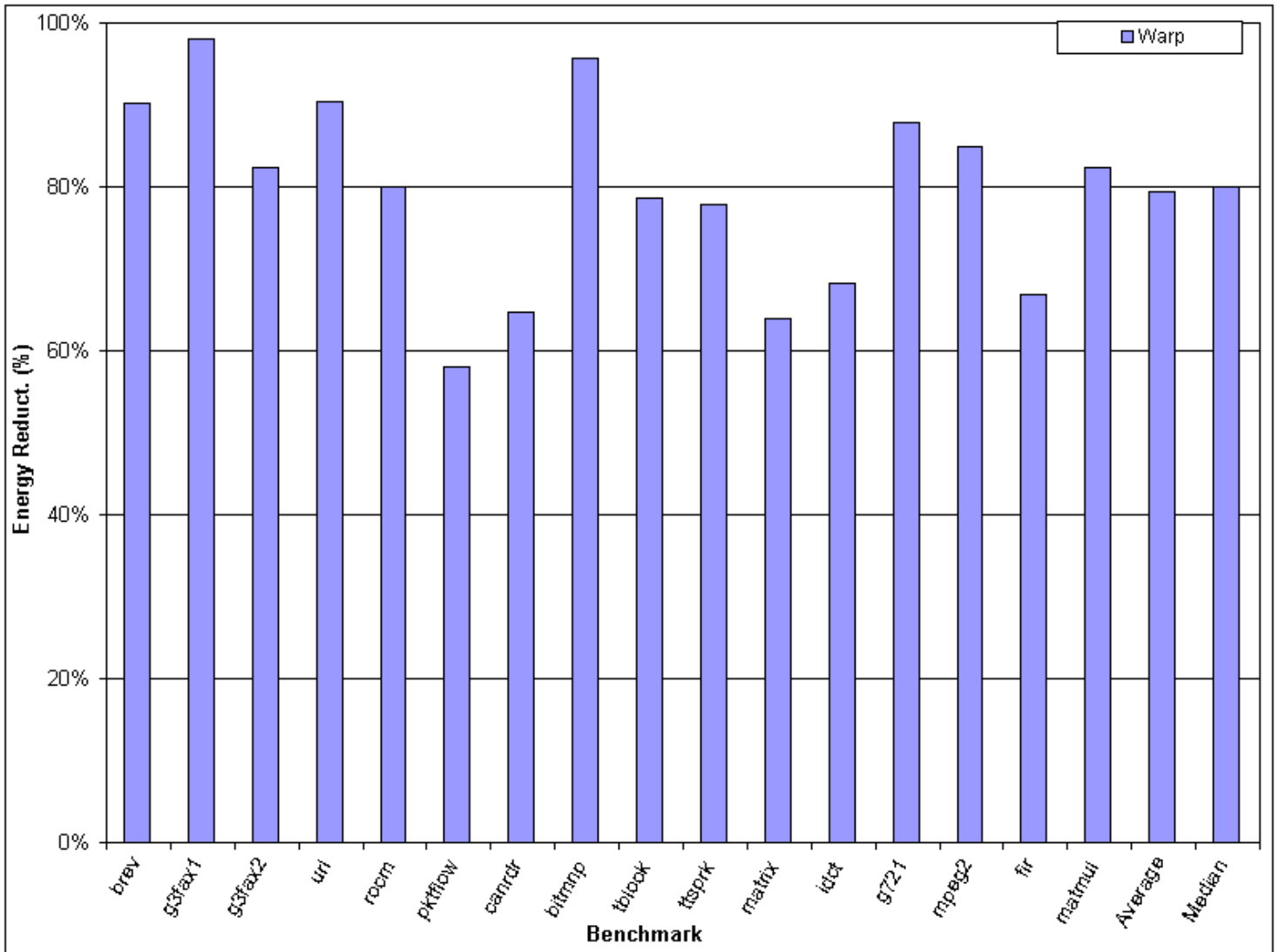


FIGURE 2—Energy reductions made possible by the UC warp processor for various embedded benchmarks.

PUBLICATIONS:

- [Dynamic Hardware/Software Partitioning: A First Approach, *IEEE/ACM Design Automation Conference \(DAC\) 2003*, pp. 250–255, June 2003.](#)
- [A Configurable Logic Architecture for Dynamic Hardware/Software Partitioning , *Design Automation and Test in Europe Conference \(DATE\) 2004*, February 2004.](#)
- [Dynamic FPGA Routing for Just-in-Time Compilation , *IEEE/ACM Design Automation Conference \(DAC\) 2004*, pp. 954–959, June 2004.](#)
- [Techniques for Synthesizing Binaries to an Advanced Register/Memory Structure, *ACM/SIGDA Symp. on Field Programmable Gate Arrays \(FPGA\)*, February 2005.](#)

