Synthesis of Partitioned Shared Memory Architectures for Energy-Efficient Multi-Processor SoC

Kimish Patel[‡] Enrico Macii [‡] Massimo Poncino [*]

[‡] Politecnico di Torino
Torino, ITALY 10129

[*] Università di Verona
Verona, ITALY 37134

Abstract

Accesses to the shared memory in multi-processor systems-on-chip represent a significant performance bottleneck. Multi-port memories are a common solution to this problem, because they allow to parallelize accesses. However, they are not an energy-efficient solution.

We propose an energy-efficient shared-memory architecture that can be used as a substitute for multi-port memories, which is based on an application-driven partitioning of the shared address space into a multi-bank architecture.

Experiments on a set of parallel benchmarks show energy savings of about 56% with respect to a dual-port memory artchitecture, at a very limited performance penalty.

1. Introduction

Modern SoC platforms normally host several processors that communicate and exchange data through shared memories. Accesses to the shared memories are significantly slower than accesses to local ones, since they require some form of arbitration, thus becoming a major bottleneck for the overall system bandwidth. Memory bandwidth can be increased by making use of different types of on-chip embedded memories, which provide shorter latencies and wider interfaces [1, 2, 3]. Multi-port memories are probably the most widely used solution for improving bandwidth. The adoption of multi-port memories, however, comes at the price of a significant increase in area, wiring resources, and energy.

This work presents a novel memory architectural template that combines the bandwidth advantages of the multi-port approach, with the advantages, in terms of energy and access time, of partitioned, single-port memories.

Based on analytical expression for performance and energy consumption that allow to explore the energy-performance tradeoff, we present experimental results showing that the proposed architecture provides eenergy savings as high as 69% with respect to a dual-port memory configuration, with comparable optimization of the memory bandwidth.

2. Multi-Port and Partitioned Architecture

We target multi-processor SoCs architectures, where each core has a cache and a private memory, containing private data and code, accessed through a local bus, and it is also connected through a common global bus to a shared memory.

We propose a shared memory architecture that combines the bandwidth advantages of multi-port memories with the energy and performance advantages, of partitioned, single-port memories [4]. Figure 1 conceptually shows the dual-port and the partitioned single-port memory architectures (A_i and D_i refer to addresses and data from processor i, respectively), for a two-processor configuration. In the dual-port scheme (Figure 1-(a)), the two read/write ports allows to bind each processor to one port, thus conceptually avoiding the need of a shared bus. In the partitioned architecture (Figure 1-(b)), addresses and data must be properly driven to connect the processor to the the required memory block. Under this scheme, the memory address space is mapped over single-port banks that can be simultaneously accessed by the different processors. Each bank covers a subset of the address space, with no replication of memory words. Since the memory blocks are single-ported and contain non-overlapping subsets of addresses, simultaneous accesses from the processor can be parallelized only if they *fit into different memory blocks*. Otherwise, the potentially

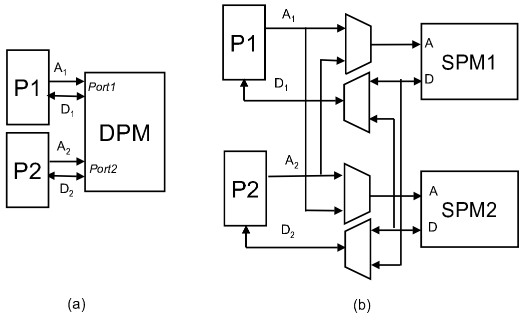

Figure 1. Dual-Port (a) and Partitioned Single-Port (b) Architectures.

1530-1591/04 $20.00 (c) 2004 IEEE

parallel access must take place into two consecutive cycles. This implies that the partitioned scheme will always have some performance penalty with respect to the multi-port architecture.

Energy efficiency is enforced by two facts: First, the single-port blocks have an energy access cost which is by far smaller than that of monolithic (either single or dual-port) memories; second, address mapping is application driven, thus it accounts for the cell access frequency to determine the size of the memory blocks which is most suitable for memory minimization.

3. Memory Architecture Synthesis

The following analysis refers to the case of two processors, although the ideas could be extended to the more general case of multiple processors.

The partitioning of the shared memory is based on the fact that the partitioned scheme is advantageous *if simultaneous accesses to memory fit into different memory blocks*. The optimal partitioning will thus be achieved by splitting the address space at a point for which this chance is maximized.

The procedure works as follows. First, we profile the application in order to determine the percentage λ of execution cycles that potentially access the shared memory simultaneously (called *parallel cycles*). “Potentially” means that simultaneous access is possible only if the accesses do not overlap on the same block. λ expresses this potentiality since it depends on what addresses fall in what block.

Based on that, we explore all possible bi-partitions of the address space, and compute the corresponding value of λ. Called B the *boundary address* between partitions, the optimal (maximum performance) value of λ is simply obtained by simply computing and choosing its maximum value. The corresponding value B_{max} defines the partitiong boundary.

Figure 2 shows the variation of λ as a function of B, for a specific application. We notice that the curve is not monotonic, showing the sensitivity of λ to the access pattern.

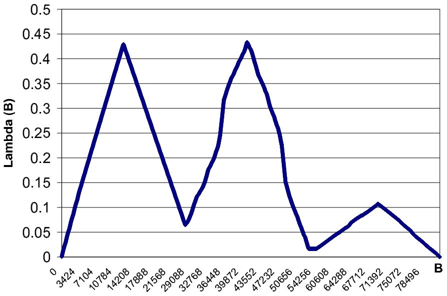

Figure 2. Behavior of $\lambda(B)$ vs. B.

4. Experimental Results

We have implemented our partitioned memory scheme in ABSS [5], an architectural simulator for multiprocessor systems. Performance and energy models for the shared memory relative to a 0.18μ technology by ST Microelectronics. Stanford’s SPLASH suite [6] have been used as parallel benchmarks.

Figure 3 compares the energy consumption of the partitioned memory and of the dual-port memory schemes, normalized with respect to the values of a monolithic single-port memory. Values do not include the cost of the decoding logic. The dual-port architecture incurs in an average energy penalty of 46% (maximum 50%), whereas the partitioned scheme reduces energy by 36%, on average (maximum 54%). The energy savings come to the price of a small performance penalty (2.48% on average).

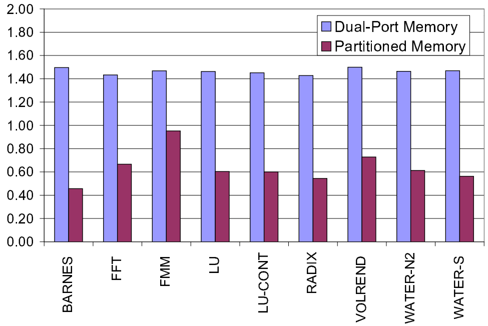

Figure 3. Energy Consumption Results.

The decoder, synthesized on the same technology as the one used for memories, and under the application of the memory access trace of one of the benchmarks, dissipates 0.35 μJ (about 1.7% of total memory energy consumption), and its delay is $310ps$, well within the slack resulting from accessing smaller memory blocks.

References

[1] F. Catthoor, et al. *Custom Memory Management Methodology Exploration for Memory Optimization for Embedded Multimedia System Design*, Kluwer, 1998.

[2] P. Panda, N. Dutt, *Memory Issues in Embedded Systems-on-Chip Optimization and Exploration*, Kluwer, 1999.

[3] A. Macii, L. Benini, M. Poncino, *Memory Design Techniques for Low-Energy Embedded Systems*, Kluwer Academic Publishers, 2002.

[4] L. Macchiarulo, A. Macii, L. Benini, M. Poncino, “Layout-Driven Memory Synthesis for Embedded Systems-on-Chip,” *IEEE Transactions on Very Large Scale Integration (VLSI)*, Vol. 10, No. 2, pp. 96-105, April 200

[5] D. Sunada, D. Glasco, M. Flynn, *ABSS v2.0: A SPARC Simulator*, Technical Report CSL-TR-98-755, CSL, Stanford University, April 1998.

[6] J. P. Singh, W.-D. Weber, A. Gupta, “SPLASH: Stanford Parallel Applications for Shared-Memory”, *Computer Architecture News*, Vol. 20, No. 1, pages 5-44, March 1992.