Low Power Memory Architectures for Video Applications

Bhanu Kapoor [1]

DSPS R&D Center
Texas Instruments Incorporated
P. O. Box 655474, MS 446
13510, N. Central Expwy., Dallas, TX, 75243

ABSTRACT

We provide data and insight into how the choice of cache parameters affects memory power consumption of video algorithms. We make use of memory traces generated as a result of running typical MPEG-2 motion estimation algorithms to simulate a large number of cache configurations. The cache simulation data is then combined with on-chip and off-chip memory power models to compute memory power consumption. In the area of analysis of video algorithms, this paper focuses on the following issues: We provide a detailed study of how varying cache size, block size, and associativity affects memory power consumption. The configurations of particular interest are the ones that optimize power under certain constraints. We also study the role of process technology in these experiments. In particular, we look at how moving to a more advanced process technology for the on-chip cache affects optimal points of operation with respect to memory power consumption.

1. INTRODUCTION

A growing number of computer systems are incorporating multimedia capabilities for displaying and manipulating video data. At the same time, power consumption has become a critical constraint especially in the design of portable [1] systems. The interest in multimedia combined with the great popularity of portable devices provides the impetus for creating portable video-on-demand system. Even for the desktop units and large computing machines, the cost of removing the generated heat as well the reliability concerns are making power reduction a priority.

Spurred by the high computation and memory bandwidth requirements for popular signal processing applications such as digital wireless communications and multimedia processing, the need for high performance as well as low power processors has never been greater. In the arena of high performance digital signal processor and microprocessor design, a large number of on-chip transistors are being devoted to memory. For example, Digital's Alpha 21064 [2] has approximately 70% of its transistors devoted to the cache. The processing of higher bandwidth signals under a stringent power budget requires a careful analysis of the memory hierarchy in the system design. The decisions related to the memory architecture has a great impact on external memory bandwidth requirements and power consumption of the memory system for these popular signal processing algorithms.

The growing inability of memory systems to keep up with processor requests has significant ramifications for the design of microprocessors in future. Much of the research so far has been focused on reducing memory access latencies [3, 4, 5, 6]. Power consumption implications of CMOS microprocessor design decisions have been studied by Bunda, Athas, and Fussel [7]. They have studied the effects of instruction code densities, cache block buffering, and cache sub-blocks. They make use of bit-wise switching statistics to model power consumption and conclude that block buffering and sub-blocks are

[1] Further author information: Email:kapoor@ti.com, Telephone:972-995-3675, Fax: 972-995-6194

0-8186-8409-7/98 $10.00 © 1998 IEEE

beneficial in reducing cache power consumption and off-chip memory traffic. Su and Despain [8] have studied the power-performance trade-offs in cache design. They have examined block buffering, banking, associativity, and gray code addressing. Their study shows that Gray coding and banking further reduce power consumption. They have also concluded that direct-mapped instruction and set-associative data caches result in better memory access times. Recently, some work on the integration of a microprocessor and DRAM memory on the same die, called Intelligent RAM (IRAM) [9], has been shown to have the potential for dramatic improvements in the energy consumption of the memory system. An IRAM will have far fewer external memory accesses, which consume a great deal of energy to drive high-capacitance off-chip buses.

A high-level analysis of energy optimization through the use of multiple-divided module (MDM) cache architecture was performed by Ko, Balsara, and Nanda [10]. Their model takes miss rates, power consumption of the cache and external memory, and the latencies of cache and external memory. It is concluded that MDMs reduce power consumption by a factor proportional to the number of modules.

Nachtergaele et al [11] describe a power exploration methodology for video applications using the low bit-rate H.263 video decoding algorithm. They show that memory consumes a large fraction of the system power consumption and describe a methodology for reducing power consumption by up to an order of magnitude. Wuytack et al [12] further explore the methodology using a motion estimation algorithm. They also present specialized techniques to optimize power consumption of address generating units. These approaches make use of custom-designed memory units to optimize power consumption for a given application. Liu and Svensson [13] have developed generalized models for memory power consumption in integrated circuits. These models were designed to model any VLSI system but do not consider factors such as miss rates, cache structure, on-chip as well as external bandwidth requirements. Landman [14] presents an overview of the state-of-the-art in high-level power estimation.

In this paper, we carry out a study of memory consumption of typical video algorithms. Due to high computation and bandwidth requirements of algorithms such as motion estimation, memory power consumption is as critical a factor as high performance in putting together a DSP or microprocessor based video processing system. In our study, we use a hierarchical motion estimation algorithm typically found in the implementations of MPEG-2 video compression [15] standard. We make use of memory traces generated as a result of running a typical MPEG-2 motion estimation algorithm to simulate a large number of cache configurations. The cache simulation data is then combined with on-chip and off-chip memory power models to compute memory power consumption. We provide a detailed study of how varying cache size, block size, and associativity affects memory power consumption. The configurations of particular interest are the ones that optimize power under certain constraints. We also study the role of process technology in these experiments. In particular, we look at how moving to a more advanced process technology for the on-chip cache affects optimal points of operation with respect to memory power consumption.

The motion estimation algorithm used in the experiments described in this paper is a hierarchical [15, 16] algorithm which uses four levels of sub-sampled images to come up with a motion vector for a given block. The second level image has half of the width and half of the height of the original image. In a similar fashion, second level is filtered and decimated to create the next two levels. A "full" search is performed on the last level of image to find the best match for each search block at this level. The algorithm is designed to work with the standard frame sizes used in MPEG-2 video codecs.

Section 2 describes the simulation methodology and the modeling aspects for our study. In Section 3, we discuss the experimental results and provide some guidelines for power-efficient memory system design. This is followed by conclusions and some suggestions for future work in Section 4.

2. SIMULATION METHODOLOGY

The study of cache behavior of video algorithms is a daunting task in itself. First of all, just encoding a few frames of MPEG-2 video sequences generate a huge amount of memory trace. For example, in order to analyze the

hierarchical motion estimation algorithm, we simulated approximately 247 million memory address references (more than 2.0 Gbytes of trace data) for each set of cache parameters. While approximate miss rates can be found using much smaller traces, our study found that Gbytes of trace data is necessary to accurately determine on-chip and off-chip memory bandwidth requirements which are essential for computing memory power consumption.

The traces containing a sequence of memory references were generated using the Quick Profiler and Tracer (QPT) [17, 19] program, an exact and efficient program profiling and tracing system. The cache simulation using the trace generated by QPT was carried out using a trace-driven cache simulator called DineroIII [18, 19] which supports sub-block placement. We have generated the trace data for the hierarchical motion estimation algorithm compiled on a Sun Ultrasparc 1 workstation.

The off-chip memory bandwidth calculations are carried out as follows:

$R_{bw} = N_r * (f_r / F)$ Mbytes/second
$W_{bw} = N_w * (f_r / F)$ Mbytes/second
Traffic = $(R_{bw} + W_{bw})$ Mbytes/second

where N_r is the number of words fetched from the external memory in Megabytes, N_w is the number of words copied back to the external memory in Megabytes, and their sum is the total traffic in Megabytes per second. The on-chip memory bandwidth calculations use the total number of demand fetches to cache, including the instruction and data fetches. The read and write bandwidths for the data cache and the bandwidth requirement on the instruction cache is then combined with the SRAM read and write power consumption data to compute the on-chip memory power consumption. The bandwidth requirements along with the read and write widths of the SRAM determine the necessary clock rate of operation, which is then used for computing memory power consumption.

The video processing system being considered here assumes a 64 Mbit Rambus DRAM based design providing up to 600 Mbytes/s of external bandwidth. A typical Rambus memory system, as shown in Figure 1, has three main elements: the Rambus channel, the Rambus DRAMs, and a Rambus interface on the memory controller. In addition to the cache and Rambus DRAM power models, active signals on the channel and their corresponding pin capacitance values are taken into account when computing memory power consumption.

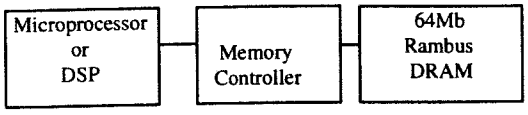

Figure 1: System block diagram.

For the on-chip power modeling, we make use of spice-simulated data for caches designed in 0.25- and 0.18-micron technologies, operating at up to 750 MHz. This clock-rate is only sufficient to support smaller frame sizes. For CIF and larger frame sizes, the required clock rate to support the on-chip bandwidth requirements is higher than 750 MHz for the SRAMs used in our sudy. The on-chip power model is then extended to account for bandwidth requirements of an application, instruction and data cache miss rates, and bandwidth utilization factor for a given application. For off-chip power model, we make use of the data sheet numbers for the 64Mb Rambus DRAM [20] with the specified pin capacitance values for the memory controller and the DRAM. A model based on these numbers is then extended to account for measured external bandwidth requirement for an application under a given cache configuration.

3. SIMULATION RESULTS

In order to calculate memory power consumption of the motion estimation, the access rates of each of the memory components, on-chip as well as off-chip, must be determined. The primary cache memory organizational characteristics that determine the external memory bandwidth (traffic) requirement are a cache's size, C, its associativity, A, and the block size B. Thus traffic is a function of these parameters:

Traffic = f (C, A, B)

In addition, the choice of cache replacement policy such as the least recently used, FIFO, and random as well as the choice unified cache versus a cache divided into instruction-only and data-only caches can also have an impact on the traffic.

Memory power consumption as a function of cache size attains a minimum value for a cache size depending on block size and associativity. The external memory bandwidth requirement decreases with the increasing cache size. The larger caches have lower miss-rates which results in less data movement between the cache and the external memory and this results in lower external power consumption. However, the power consumption of on-chip cache increases with the increasing size of the cache. The caches used in these experiments are direct-mapped and have separate instruction and data caches. The replacement policy for all the results presented in this paper is the least recently used (LRU). The memory power consumption is normalized with respect to the power consumption corresponding to the configuration using 4 Kbytes each of direct-mapped instruction and data cache with a block size of 8 Bytes. The data points for four block sizes, ranging from 8 Bytes to 32 Bytes, are shown in Figure 2. Each plot has a minimum which shifts to the right as the block size is increased. The on-chip power consumption numbers are for the cache designed 0.25-micron technology.

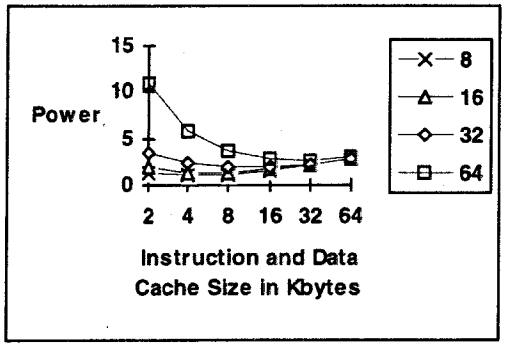

Figure 2: Memory power consumption versus cache size for various block sizes.

The memory power consumption typically increases with the increasing block size and the rate of increase depends on the cache size. As the block size is increased, each miss brings in larger amount of data from the external memory to the cache. However, this also reduces the probability of further misses. The plots for four cache sizes, ranging from 2 Kbytes to 32 Kbytes each of instruction and data cache, are shown in Figure 3. As cache size is increased, block size plays less important a role as shown by flatter plots for the larger caches.

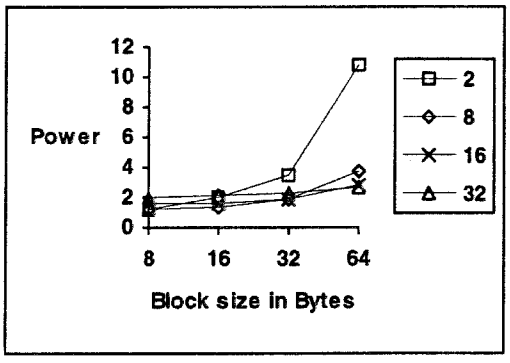

Figure 3: Memory power consumption versus block size for various cache sizes.

The system memory power consumption as a function of associativity typically decreases with the increasing value of associativity. The plots for four cache sizes, ranging from 2Kbytes to 32 Kbytes, are shown in Figure 4. For smaller caches, there is a big reduction in power consumption as we go from a direct-mapped cache to a two-way set associative cache. This is mainly due to the improved hit rates as a result of increasing associativity, which reduces the number of off-chip accesses to the external memory.

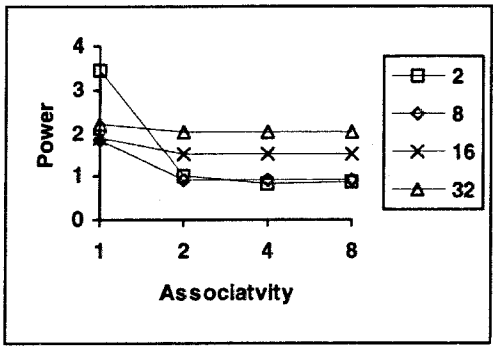

Figure 4: Memory power consumption versus associativity for various cache sizes.

The memory power consumption increases with the increasing bit-rate of the algorithm. For the motion estimation algorithm used in our experiments, there is an approximately linear dependence between power consumption and the input video bit-rate as we increase the bit-rate from 1.3 Mb/sec to 83 Mb/sec by changing the

sizes of the video frames. Among the cache replacement policies such as the least recently used (LRU), first-in first-out (FIFO), and random replacement, LRU policy performs the best with respect to the memory power consumption.

The memory power consumption reduces, as expected, as we move to a more advanced technology with smaller device sizes. The plots for the 0.25- and 0.18-micron technologies are shown in Figure 5. Each plot has a point of minimum power consumption and this point shifts to the right as we move to a more advanced technology. There is an optimal power utilization point at a larger cache size in more advanced technologies. The plots shown in Figure 5 are for direct-mapped instruction and data cache with a block size of 32 Bytes. Similar behavior is seen for other combinations of block size and associativity.

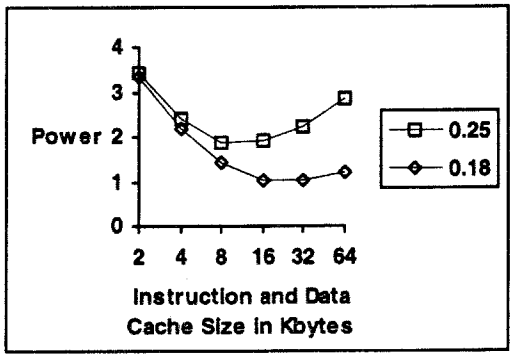

Figure 5: Memory power consumption versus cache size for two technologies.

In this case, the least power-consuming configuration is a cache which is 4X larger in the more advanced technology. Due to the increasing external memory bandwidth requirements for applications such as the video processing, 3D graphics, animation, and virtual reality, larger caches will provide more optimal configurations for overall memory consumption of future systems. This is a good news from system power consumption point of view as it is supporting the natural evolution of processor design with larger on-chip memories.

4. CONCLUSIONS

We have provided a study of how varying cache size, block size, and associativity affects memory power consumption. As it is clear from the experimental results, there is a point of diminishing return with respect to all the cache parameters in achieving low power memory architectures. Multi-level memory hierarchies may present a way out of this problem and are being used these days. We will like to extend this study to multi-level caches. In addition, a detailed study of various options available with video algorithms is important. In particular, studying the effects of quantization and other parameters that effect the bit-rate of the algorithms should provide useful data from the standpoint of bandwidth requirement. As it is evident from our experiments, power consumption can be reduced significantly by appropriately configuring the memory architecture of the system. In addition, such a study can lead to reconfigurable memory architecture which adapts the parameters to a given application.

5. ACKNOWLEDGMENTS

We are grateful to Jim Larus of University of Wisconsin-Madison for his help in installation of the QPT software on Solaris Version 5.5.1 and in providing answers to several questions regarding the usage of the software. Thanks to Mark D. Hill of University of Wisconsin-Madison for his help on the usage of DineroIII Cache Simulator. Thanks to Patrick Bosshart, Paul Fuqua, and Yiwon Wang of Texas Instruments for their help during this project. Finally, thanks to Susan Hric of TI for her help in the installation of the required GNU software tools.

6. REFERENCES

[1] Anantha Chandrakasan, "Low Power Digital CMOS Design", PhD Thesis, University of California, Berkeley, 1994.

[2] D. Doberpuhl et al, "A 200 MHz 64b Dual-issue CMOS Microprocessor", *IEEE Journal of Solid-State Circuits*, vol. 27, no. 11, Nov., 1992.

[3] Mark D. Hill and Alan Jay Smith, "Experimental Evaluation of On-chip Microprocessor Cache Memories", *Proc. of Eleventh International Symposium on Computer Architecture*, June 1984, Ann Arbor, MI, pp. 158-174.

[4] Alan J. Smith, "Cache Memories", *Computing Surveys*, Vol. 14, No. 3, pp. 473-550, September 1982.

[5] John Hennessy and David Patterson, Computer Architecture: A Quantitative Approach. Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1996.

[6] Steven A. Przybylski, "Cache and Memory Hierarchy Design: A Performance Directed Approach", Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1990.

[7] J. Bunda, W. Athas, and D. Fussel, "Evaluating Power Implications of CMOS Microprocessor Design Decisions", *Proceedings of the 1994 International Workshop on Low Power Design*, April 1994, pp. 147-152.

[8] C. Su and A. Despain, "Cache Design Trade-offs for Power and Peformance Optimization: A Case Study", *Proceedings of the 1995 International Symposium on Low Power Design*, April 1995, pp. 63-68.

[9] Richard Fromm, Styllianos Perissakis, Neal Cardwell, Christoforos Kozyrakis, Bruce McGaughy, David Patterson, Tom Anderson, and Katherine Yelick, "The Energy Efficiency of IRAM Architectures", *Proc. of 24th Annual International Symposium on Computer Architecture*, June 1997.

[10] U. Ko, P. Balsara, and A. Nanda, "Energy Optimization of Multi-level Processor Cache Architecture", *Proceedings of the 1995 International Symposium on Low Power Design*, April 1995, pp. 45-49.

[11] L. Nachtergaele, F. Catthoor, B. Kapoor, D. Moolenaar, and S. Janssens, "Low-power Storage Exploration for H.263 Video Decoder System", *1996 IEEE Workshop on VLSI Signal Processing*, pp. 115-126, Nov., 1996.

[12] S. Wuytack, F. Catthoor, L. Nachtergaele, H. De Man, "Power Exploration for Data Dominated Video Applications", *1996 International Symposium on Low Power Electronics and Design*, pp. 359-364, Aug, 1996.

[13] D. Liu and C. Svensson, "Power Consumption Estimation in CMOS VLSI Chips", *IEEE Journal of Solid-State Circuits*, pp. 663-670, Jun, 1994.

[14] P. Landman, "High-level Power Estimation", *1996 International Symposium on Low Power Electronics and Design*, pp. 29-36, Aug, 1996.

[15] V. Bhaskaran and K. Konstantinides, "Image and Video Compression Standards," pp. 105-128, Kluwer Academic Publishers, 1995.

[16] Bede Liu and Andre Zaccarin, "New Fast Algorithms for the estimation of Block Motion Vectors", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 3, No. 2, 1993.

[17] James R. Larus, "Quick Profiler and Tracer(QPT)", *http://www.cs.wisc.edu/~larus/warts.html*

[18] Mark. D. Hill, "DineroIII Cache Simulator", *http://www.cs.wisc.edu/~larus/warts.html*

[19] Mark D. Hill, James R. Larus, Alvin R. Lebeck, Madhusudhan Talluri, and David A. Wood, "Wisconsin Architectural Research Tool Set", Computer Architecture News, 21(4):8-10, August 1993.
http://www.cs.wisc.edu/ ~larus/warts.html

[20] Rambus Website: Products, http://www.rambus.com/html/products.html