

A REVIEW ON TRENDS IN MULTICORE PROCESSOR BASED ON CACHE AND POWER DISSIPATION

K.Aruli¹, B.Nivetha², G.N.Jayabhavani³, Dr. M.Saravanan⁴

^{1 2}Student, M.E-Applied Electronics, IFET College Of Engineering, Tamilnadu

³ Assistant Professor, Electronics And Communication Engineering, IFET College Of Engineering, Tamilnadu

⁴ Associate Professor, Electronics And Communication Engineering, IFET College Of Engineering, Tamilnadu

Abstract - A multi-core processor is an integrated circuit to which two or more processors attached for enhanced performance, reduced power consumption, and more efficient simultaneous processing of multiple tasks. Multi-core processing is a growing industrial trend as single core processors rapidly reach the physical limits of complexity and speed. It exploits increased feature-size and density. It increases functional units per chip and limit energy consumption per operation. This paper provides the review of various papers in multicore architecture, discussing various parameters like number of cores, cache coherence and power dissipation.

Key Words: multicore, single core, cache, energy etc...

1. INTRODUCTION

A multi-core processor is a processing system composed of two or more independent cores. A many-core processor is one in which the number of cores is large enough that traditional multi-processor techniques are no longer efficient. A multi-core processor implements multiprocessing in a single physical package. Cores in a multi-core device may be coupled together tightly or loosely. For example, cores may or may not share caches, and they may implement message passing or shared memory inter-core communication methods. Common network topologies to interconnect cores include: bus, ring, 2-dimensional mesh, and crossbar.

A multi-core processor has many advantages over a single core processor in performing multitasking operation. It has its own cache, even though it has some positive and negative views when we go for increasing number of cores. The performance can be improved in the multi-core processor based on the software algorithms used and their implementation. Amdahl's law describes that few gains will be limited by software which runs in parallel manner. Few Problems were identified while implementing multiple cores on a single chip. With increased number of multicore processors, Power and

temperature management are two main concerns, and also cache coherence problem, using the entire potential of multicore processor is another issue. There are two types in multicore processor.

- Symmetric multicore system.
- Asymmetric multicore system.

All cores are identical in symmetric multi-core systems and they are not identical in asymmetric multi-core systems. Just as with single-processor systems, cores in multi-core systems may implement architectures such as superscalar, vector processing, or multithreading. Multicore has better performance. For example, in multicore processor, two complex operations can be performed at a time like CD burning with parallel graphical works. Multi-core processing is a growing industry trend as single core processors rapidly reach the physical limits of possible complexity and speed.

2. CACHE COHERENCE

2.1 Improving Multi-Core Performance Using Mixed-Cell Cache Architecture.

In this paper Samira M. Khan et.al., discussed about a mixed cell architecture that improves multi-core performance by allowing the use of both robust and non-robust cells. This system mechanisms store modified data only in robust lines by modifying the cache replacement policy and handling writes to non-robust lines. In a multi-core processor, this best mechanism improves performance by 17%, and reduces dynamic power in the L1 data cache by 50% over prior mixed-cell proposals.

The key idea behind this paper mixed- cell cache is to protect modified lines by storing them in robust cells, while using the remainder of the cache for clean lines. They have used a simple error detection and correction mechanisms to detect errors in clean lines [1]. They allocate, write misses to robust lines, and read misses to the clean lines. On a subsequent write to a clean line, they investigated the three alternatives to ensure modified data is not lost.

2.2 Performance of Cache Memory Subsystems For Multicore Architectures

In this paper N. Ramasubramanian et.al., discussed about a performance of cache memory is evaluated through the parameters such as cache access time, miss rate and miss penalty. The influence of cache parameters over execution time is also discussed in this paper.

The primary objective of this paper is the evaluation of the impact, caches have on different instructions set architectures [2]. This is achieved by taking one particular benchmark from SPLASH2 and finding the access time on ALPHA ISA using M5sim and comparing the results with the results obtaining using CACTI an X86 ISA.

The M5 is an emulation tool that is capable of performing event driven simulation. It enables users to simulate a multi-core environment with error modularity close to the hardware. Objects are components such as CPU, memory, caches which can be accessed through interfaces. Events involve clock ticks that serve as interrupts for performing specific functionality. There are two basic modes of operation, namely: full system mode and system call emulation mode. The major difference between the two modes of operation is that, the later executes system calls on a host machine, whereas the former emulates a virtual hardware whose execution is closer to the real system.

2.3 Improving Cache Performance by Exploiting Read-Write Disparity

In this paper Alaa R. Alameldeen et.al., discussed about cache management techniques that increase the probability of cache hits for critical read re-quests, potentially at the cost of causing less critical write re-quests to miss.

Cache read misses stall the processor if there are no independent instructions to execute. In contrast, most cache write misses are off the critical path of execution, since writes can be buffered in the cache or the storage buffer. The key contribution of this paper is the new idea of distinguishing between lines that are reused by reading versus those that are reused only by writing to focus cache management policies on the more critical read lines. They proposed a Read-Write Partitioning (RWP) policy that minimizes read misses by dynamically partitioning the cache into clean and dirty partitions, where partitions grow in size if they are more likely to receive future read requests.

They showed that exploiting the differences in read-write criticality provides better performance over prior cache management mechanisms. For a single-core system, RWP provides 5% average speed up across the entire SPEC CPU2006 suite, and 14% average speed up for cache-sensitive benchmarks, over the baseline LRU

replacement policy. They also show that RWP can perform within 3% of a new yet complex instruction-address-based technique, Read Reference Predictor (RRP) that bypasses cache lines which are unlikely to receive any read requests, while requiring only 54% of RRP's state overhead. On a 4 core system, our RWP mechanism improves system throughput by 6% over the baseline and out performs three other state-of-the-art mechanisms evaluate.

3. POWER AND ENERGY DISSIPATION

3.1 Power And Energy Profiling of Scientific Applications on Distributed Systems.

In this paper X. Feng et.al discussed about a Direct measurement approaches target direct, automatic profiling of power consumption at the component level of a server [4]. The authors proposed a methodology to separate component power after conversion from AC to DC current in the power supply of a typical server. They automated data profiling, measurement and analysis by creating a tool suite called Power Pack. They mapped the power/energy consumption to application segments and exploited parallel performance inefficiencies characterization of non-interactive distributed applications.[3]This method is straightforward for estimating power consumption and provides power profiles for components for which power cannot be estimated by simulation-based and hardware counter-based approaches. However, this method must add extra circuits to measure DC current and must set up the mapping between current variance and component utilization.

In addition, this method resorts to professional software to collect power data from data acquisition systems. Therefore, it is not convenient to feed power data to other runtime systems to control power consumption.

3.2 A Multi-Core Approach to Addressing the Energy-Complexity Problem in Microprocessors

In this paper, Rakesh Kumar, Keith Farkas et.al., discussed about the mechanism of reducing processor power dissipation in multicore architectures. As processors continue to increase in performance and speed, processor power consumption and heat dissipation are the challenging issues in in the designing high performance systems. Multi-core architecture where all cores execute the same instruction set, but have different capabilities and performance levels, whereas Prior chip level multiprocessors (CMP) using multiple copies of the same core. In this proposed system, different applications will be having different resource requirements during their execution [6]. Some applications may have a large amount of instruction-level parallelism (ILP), the same application in some sort of core used with little ILP will

consume more power. Hence, it can be possible to run an application with less complexity instead of high complexity and can achieve the similar performance. In this discussion heterogeneous multi-core architecture concentration of voltage and frequency reduces the parameters of the entire core. Thus power reductions are equivalent on both the portions of the core. Gating-based approaches do not address the power consume, rather than increasing multithreading. Heterogeneous multi-core architecture is based on the hypothesis of choosing best core; performance of the core depends on the applications. One application may get a better solution from wide issue and dynamic scheduling, other benefits from neither new execution type by executing only one application runs at a time on only one core. This design point could assume maximum of one thread running at one time, we implement cache coherence by flushed the unwanted cache from one core before execution is moved to another core. The multi-core processor is assumed to be designed with 0.10 micron technology. At 0.10 microns, the cache will have an area about half of the die-size of the Pentium 4.

3.3 Technology Trends in Computer Architecture and their Impact on Power Subsystems

In this paper Kevin Kettler et.al discussed about the high-level trends in computer architecture and the related drivers [5]. A survey of the microprocessor, graphics, memory, and I/O subsystems in emerging computing technology that will drive increased demands for power. As the industry improves from one process dimension to a smaller dimension, the results would be provided in several ways. Thereby, users can go to the same device with smaller die sizes, higher frequency, lower power, and higher yields. In some cases, the silicon developers in designing of improvement in die size allow additional features into a device. This results in better functionality, even with the smaller process geometry. The continuous process of integrating new features and technology results in the current generation of computers with better performance.

The memory subsystem is an integral part of CPU performance. When the price of memory becomes low as, per silicon technology improved it helps to support overall system performance. The increasing number of advanced applications and their complexity of data leads increase the amount of memory needed by the system. The increasing features of DDR2 and DDR3 memory technologies produce power and thermal challenges. Solutions to reduce these thermal problems such as dedicated fans for system memory in architecture design. Heat spreaders can also be employed to help reduce the heat.

Each successive generation of system enables new solutions of software and hardware that increases the advantages in existing and creates new usage for

computers. The power system and its resulting usage by components is an integral part of the infrastructure. The challenge for the industry is to implement power systems that can more effectively involved in the increasing demands of future computing platforms.

4. CONCLUSIONS

A multi-core architecture has two or more independent cores. The performance can be improved by implementing software algorithms. With these advance technologies it faces few challenges such as power dissipation, memory consistency and cache coherence problem etc. With this review, we concentrated on two parameters such as cache coherence and heat dissipation implementation of robust cells and non-robust cells, a multicore architecture mechanism is improved by 17% and reduced dynamic is achieved in cache about 50% by Simple error detection and correction mechanism, which makes the system performing better. The Multicore architecture mechanism is improved, with the implementation of robust cells and non-robust cells by simple error detection and correction mechanism, which makes the system performing better.

Heat dissipation is another challenging issue in multicore processor when we need to improve the performance and execution time. In order to maintain the performance less complexity instruction can be used which runs in ILP. When designing the multicore processor, rather than increasing the number of core, Gate-based approach can be implemented in multithreading, the performance could be better.

REFERENCES

- [1] Samira M. Khan et.al, "Improving multi-core performance using Mixed-cell cache architecture." High Performance Computer Architecture (HPCA2013), 2013 IEEE 19th International Symposium.
- [2] N. Ramasubramanian et.al "Performance of cache memory subsystems for multicore architectures", International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.1, No.5, October 2011.
- [3] H. Chang et.al., "Power Pack: Energy Profiling and Analysis of High-Performance Systems and Applications". IEEE Transactions on Parallel and Distributed Systems (TPDS), 21:658-671, 2010.
- [4] X. Feng et.al., "Power and Energy Profiling of Scientific Applications on Distributed Systems" in Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS), 2005.

[5] Kevin Kettler, "Technology Trends in Computer Architecture and Graphics Chip Set their Impact on Power Subsystems", *0-7803-8975-1/0511620.00* Q2005 IEEE.

[6] Rakesh Kumar Keith et.al. "A Multi-Core Approach to Addressing the Energy-Complexity Problem in Microprocessors". Proceedings of the Workshop on Complexity-Effective Design (WCED), June 2003.

[7] Jay Hoeflinger, Prasad Alavilli, Thomas Jackson, and Bob Kuhn, "Producing scalable performance with Open MP: Experiments with two CFD applications," *International Journal of Parallel computing*. 27(2001), 391-413.

[8] en.wikipedia.org/wiki/Multi-core_processor.