A Comparative Power-Performance Analysis of Microarchitecture Effects on Heterogeneous CPU-GPU

Vijayalakshmi Saravanan and Ramalingam Sridhar
University at Buffalo
Email: {vsaravan, rsridhar}@buffalo.edu

ABSTRACT

High Performance Computing (HPC) users are exploring heterogeneous computing through the integration of CPU-GPUs to maximize the computational throughput. The performance of the processors depends on many micro-architectural parameters such as issue-width, functional units and pipeline depth. The increasing number of processor cores and depth of instruction pipelines continues to add complexity to task/thread parallelism and the ratio of power/performance. To adequately address these issues and potential benefits and pitfalls that may arise from this heterogeneous processors, it is important to have a deep understanding of application-level and microarchitecture-level demands from CPU-GPU cores. Multi-core CPUs have already been studied thoroughly owing to their larger history in the field. This paper aims to evaluate the scalability of CPU-GPU per stream cores and micro-architectural behavior for parallel applications executing on each core type in heterogeneous CPU-GPU processor simulation. To obtain this, we conduct a set of detailed benchmarks for many-core systems from PARSEC [2] on the ESESC simulator [5].

1. INTRODUCTION

Following the recent substantial development of multi-core CPUs in the field of high performance computing, the arrival of heterogeneous computing through the integration of CPUs and GPUs aims at achieving higher performance gains. Owing to this advancement in technology and with technology scaling, one needs to study the characteristics of hardware configuration and strive to enhance resource utilization. But the recent introduction of sophisticated NVidia GPUs and AMD APUs increases the usage of heterogeneous computing systems in research and the engineering domain. Hence, it has become inevitable to evaluate the performance of these mixed computing environments. At the same time CPU-GPU systems should be designed with power efficiency in mind. Furthermore, the power budget allocated to the CPU-GPU processors leads to performance degradation. By evaluating the impact of CPU-GPU microarchitecture configuration in different application domains, the performance and power trade-off can be identified. The micro-architecture effects on CPU and GPU memory system behavior is studied [1]. The workload characterization of parallel applications for cloud/server [2][3], GPU applications [4], mobile and embedded application [7]. All of these studies characterized the applications on a single architectural platform. In [8], the authors studied the micro-architecture dependent and independent parameters. Another class of related work seeks to compare the performance of different applications using different architectures or different techniques [6]. The work focuses on the applications and performance limitations. Our work focuses on beyond the memory system and workload characterization that can cause the power and performance trade-offs. This paper also aims to evaluate the number of cores per CPU and GPU unit, in order to address the power-performance trade-offs.

2. MOTIVATION

- Desktop, laptops, mobile and high-performance computing systems incorporating GPU's in order to increase the throughput. Today, most of these computing systems employ CPUs and GPUs on separate chips. Within the last couple of years, heterogeneous computing systems evolved to provide fast computing.
- Modern processors widely employ large on-chip resources such as functional units, caches and pipeline to improve performance which in turn increase the complexity. The performance of the processor depends on many factors such as issue-width and pipeline depth. Out of these factors, one of the major contributors that affect the architectural performance is the issue-width, number of core or stream cores and pipeline depth.
- The relationship between these specific micro-architectural parameters of heterogeneous CPU-GPU computing are seldom analyzed in published studies. This paper proposes to study the trade-offs between power- performance and thermal constraints of CPU-GPU processors.
- The processor performance is based on two important metrics: architecture and application. The difficulty in exploiting more instruction level parallelism (ILP) and the support of multi-threading in GPU processors has led to architectural innovation. Hence, this motivated us to do research on micro-architectural impacts and its power-performance efficiency on CPU-GPU.

3. METHODOLOGY

To adjust various microarchitecture system parameters and to collect detailed statistics, we simulate and compare the heterogeneous CPU-GPU as described in below section. This section also describes the various benchmarks and simulation environment used for this comparison.

Architecture & Simulated System configuration: As this paper centered on micro-architectural behavior, we used the Enhanced SuperEScalar (ESESC) simulator [5], the PARSEC benchmark suite shown in Table 1 and the Samsung Exynos4 ARM processor architecture, a best-in-class mobile processor for performance and efficiency. The following is a summary of Exynos4's features: Instruction Window: 128 ROB, 80 LSQ Functional Units: AUNIT, BUNIT, CUNIT, L&S UNIT (Load & Store), Fetch Width: 4, Branch Predictor Type: Hybrid; Decode Width: 8, BTB: 512, L1 D-Cache: 32KB, 2-way L1 I-Cache: 32 KB, 2-way Private L2 Cache Memory latency: 150 cycle.

Table 1 Benchmarks and Program size

Program	Application Domain	Workload Nature	Program size
1.Blackscholes	Financial Analysis	Data-parallel	65,536 options
2.Swaptions	Financial Analysis	Data-parallel	16 swap, 20,000 simulations
3.Facesim	Animation	Data-parallel	1 frame, 372,126
4.Fluidanimate	Animation	Data-parallel	10 frame, 300K particles
5.X264	Media processing	Pipeline	128 frames, 640x360 pixels
6.Ferret	Similarity search	Pipeline	256 queries, 34,973 images
7.Canneal	Engineering	Unstructured	2500000 elements
8.Bodytrack	Computer vision	Data-parallel	4 frames, 4,000 particles

Power- Performance Comparison Results

In this section, we present our simulation results and analysis of architectural parameter change. We consider three micro-architecture design points that differ in issue-width, number of cores ranging from quad-core to 16 and their issue-width ranging from 4 to 16 and various pipe-lined implementation. As we scale the processor performance, we

adjust the few parameters inside the architecture configurations such as fetch width, functional units and instruction window.

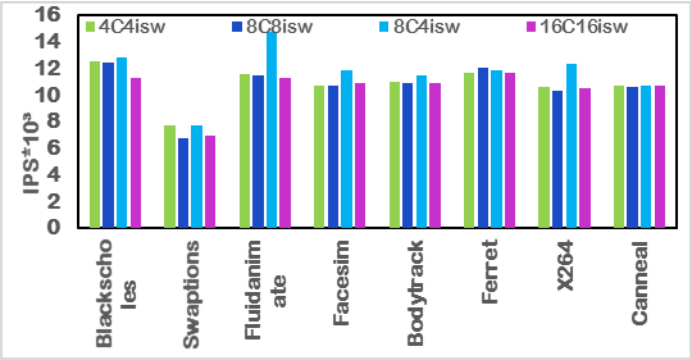

Figure 1. Issue-width vs. IPS

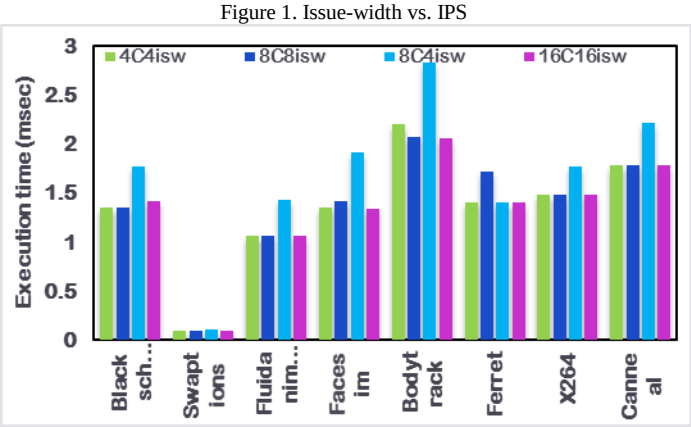

Figure 2. Issue-width vs. Execution time

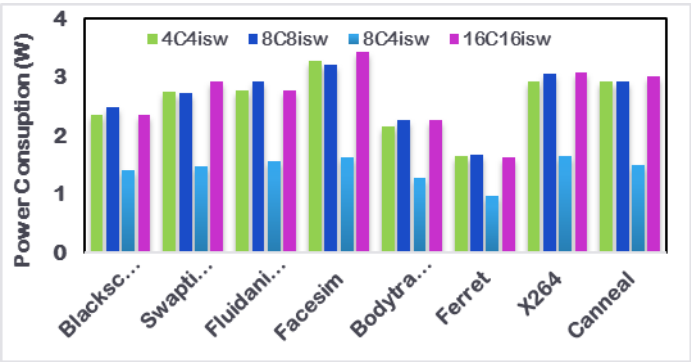

Figure 3. Issue-width vs. Power consumption

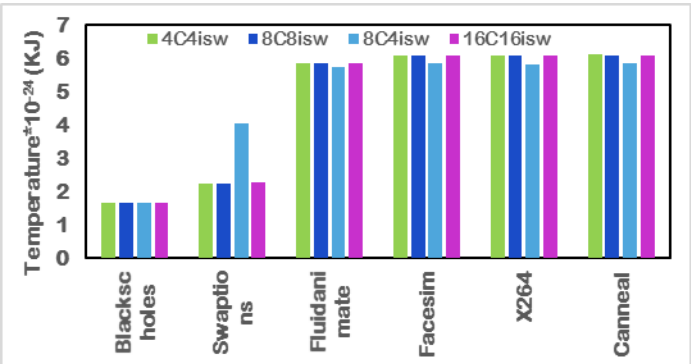

Figure 4. Issue-width vs. Temperature

The experiment is based on the heterogeneous workloads as categorized in the Table 1. The power-performance for the different workloads are plotted in Figure 1- Figure 4. The PARSEC applications such as program number 1-4 and 8 are compute intensive workloads capable of parallel processing. Program number 1 and 2 are number crunchers and hence the need is to have only higher computational processing whereas program number 3,4 and 8 require both higher computational and graphical processing.

a. Ferret and X264 are six-stage pipeline models with varying functional aspects. Ferret is computational intensive whereas X264 is both computational and graphical intensive workload.
b. If we consider pipeline modeled workloads such as ferret and X264 has the best throughput IPS and execution time for eight-four core with eight-four issue-width (8C4isw) respectively. The reason is that both of these benchmarks pipeline model has complexity of hardware switching which leads to number of instruction executed is greater for the 16 core issue-width.

In summary, our study focuses on evaluating the micro-architectural parameters in a heterogeneous CPU-GPU setup rather than single architecture platform. The results show that for all the compute intensive workloads capable of parallel processing, eight-core with four issue-width is the optimal design which is high performing as well as energy and thermal efficient; except for Ferret which recorded a meagre 1.75% drop in IPS performance as compared to its best run with eight cores with eight issue-width. However, eight-core eight issue-width has the best execution time for X264 & four core with four issue-width for ferret when compared to others. Overall, our simulation results show that eight-core with four issue-width has performed better than other combinations of these micro-architectural parameters on IPS, energy and thermal efficiency.

4. CONCLUSIONS

We presented a CPU-GPU heterogeneous power-performance analysis for various workloads and analyzed its micro-architectural behavior of energy efficient mobile and desktop CPU-GPU processors. We have studied the four different systems having different cores and issue-width. The results show that high issue-widths/cores of 16 provide very little to no increase in performance in terms of IPS and the power dissipated at the same time is very high. Hence, the ideal issue-width would be eight-core with four issue-width wherein power-performance optima is observed.

5. ACKNOWLEDGEMENTS

This work was supported in part by NSF grant CNS-1422304 and “Schlumberger Foundation Faculty for the Future” (FFTF) award.

REFERENCES

[1] Hestness, J., Keckler, S. W., & Wood, D. A. (2014, October). A comparative analysis of microarchitecture effects on CPU and GPU memory system behavior. In Workload Characterization (IISWC), 2014 IEEE International Symposium on (pp. 150-160). IEEE.

[2] Bienia, C., Kumar, S., Singh, J. P., & Li, K. (2008, October). The PARSEC benchmark suite: characterization and architectural implications. In Proceedings of the 17th international conference on Parallel architectures and compilation techniques (pp. 72-81). ACM.

[3] Burtscher, M., Nasre, R., & Pingali, K. (2012, November). A quantitative study of irregular programs on GPUs. In Workload Characterization (IISWC), 2012 IEEE International Symposium on (pp. 141-151). IEEE.

[4] Gregg, C., & Hazelwood, K. (2011, April). Where is the data? Why you cannot debate CPU vs. GPU performance without the answer. In Performance Analysis of Systems and Software (ISPASS), 2011 IEEE International Symposium on (pp. 134-144). IEEE.

[5] Ardestani, E. K., & Renau, J. (2013, February). ESESC: A fast multicore simulator using time-based sampling. In High Performance Computer Architecture (HPCA2013), 2013 IEEE 19th International Symposium on (pp. 448-459). IEEE.

[6] Diamond, J., Burtscher, M., McCalpin, J. D., Kim, B. D., Keckler, S. W., & Browne, J. C. (2011, April). Evaluation and optimization of multicore performance bottlenecks in supercomputing applications. In Performance Analysis of Systems and Software (ISPASS), 2011 IEEE International Symposium on (pp. 32-43). IEEE.

[7] Kim, H., Maxwell, R., Patel, A., & Lee, B. K. (2012). Performance Evaluation of Multi-Threaded System vs. Chip-Multi-Processor System. PhD thesis, University of Texas at San Antonio.

[8] Matthew Sinclair, Henry Duwe, Karthikeyan Sankaralingam. Porting CMP Benchmarks to GPUs. Technical Report TR-1693, Department of Computer Sciences, The University of Wisconsin-Madison, 2011.