International Journal of Multimedia and Ubiquitous Engineering
Vol.9, No.2 (2014), pp.153-160
http://dx.doi.org/10.14257/ijmue.2014.9.2.15

A Simple Model for Estimating Power Consumption of a Multicore Server System

Minjoong Kim, Yoondeok Ju, Jinseok Chae and Moonju Park*

School of Computer Science and Engineering, Incheon National University
Incheon, Korea

{mjkim,ydju,jschae,mpark}@incheon.ac.kr

Abstract

Balancing the performance and the energy consumption of the servers is one of the important issues in large-scale computing infrastructure such as data centers. Measuring or accurately estimating power consumption of a server is one of the most fundamental and enabling technologies for enhancing energy efficiency of a server because how the server consumes the supplied power is essential for constructing a power management policy. For the purpose, power models for server systems have been extensively studied. However, most of existing works are too complex to be used real-time, because gathering the data for estimating the power consumption causes much overhead. In this paper, we propose a simple power model for a multicore server. Our model is simple enough to gather only four parameters: operating frequency, the number of active cores, the number of cache accesses and the number of the last level cache misses. We show our model is simple but relatively accurate by experiments that show the model has over 90% accuracy.

Keywords: *Multicore, Power model, Server system*

1. Introduction

As the energy costs rise, power management techniques are extensively studied. DVFS (Dynamic Voltage and Frequency Scaling) is a technique to save CPU energy by dynamically adjusting both applied voltage and frequency [1]. Reducing the power consumption of the CPU is important because the CPU is the most power-consuming device when the computing device is actively running. Most of the contemporary CPUs support the DVFS technique, and many OS's like Linux support software control of CPU frequencies. Energy efficiency of a server depends on the policies which determine when to change the frequency level and which frequency level it should change to. To make a decision for energy efficiency, it is important to know how much power is being consumed with current settings, and how it will be changed if some settings are changed. To this ends, an accurate model for power consumption is needed.

Many studied the power model of the server with different approaches. A comparative study on the power models can be found in [2]. More complex analysis on the power consumption of the complete system is presented in [3]. However, in practical use, more

* *Corresponding author*

ISSN: 1975-0080 IJMUE

complex method does not guarantee more accurate results. Furthermore, the existing models require gathering many data on system status, which may cause large overhead on the system. In this paper, we propose a simple power model for multicore servers that can be efficiently used for real-time software power measurement. Although simple, the model provides over 90% accuracy as we show with experiments.

This paper is organized as follows. Section 2 discusses the previous research results and based on those present a power model for multicore servers. In Section 3 we present the experimental results that compare the power measured with actual system and the power predicted by the model. Section 4 concludes our work and suggests future works.

2. A Power Model for Multicore Systems

2.1. Power Consumption Model for Multicore CPUs

Power consumption of CMOS based chips can be classified in dynamic and static power consumption [4]. Dynamic power consumption is dissipated due to switching activity, while static power consumption is due to leakage currents. So the power consumption of the CPU.

Processor speed or computing capacity is almost linearly proportional to the clock speed *f*, and the dynamic power consumption of the CPU is proportional to the multiplication of the clock speed and the square of the supply voltage *V* [5]. So the dynamic power consumption of a CPU can be formulated as:

$$P_{dynamic} = \alpha C V^2 f \qquad (1)$$

where $P_{dynamic}$ is the dynamic power consumption, *C* represents the capacitance, and α is the activity factor of the processor (percentage of gates that switch for each cycle, on average 50%) [6].

When the processor is applied at a low voltage level, the frequency is decreased because of the increased circuit delay. Thus we have $f \propto \frac{(V-V_T)^2}{V}$ where V_T is the threshold voltage which is much smaller than the supply voltage *V* so can be ignored [7]. By letting $f = kV$ where *k* is a constant, Equation (1) becomes to

$$P_{dynamic} = \alpha C f^3 / k^2. \qquad (2)$$

Regarding $\alpha C/k^2$ as a constant β, we have $P_{dynamic} = \beta f^3$.

Static power consumption is given by $P_{static} = I_q V$, where I_q is the leakage current and *V* is the supply voltage [7]. As discussed above, we can let $f = kV$ with a constant *k,* static power consumption is given by

$$P_{static} = \gamma f \qquad (3)$$

assuming there is little change in leakage current.

Other components with large power consumption in multicore SoC are cache memories. Most of today's processors have on-chip cache memories, which often occupy large area in the chip. Therefore, the power consumption due to accesses to the cache memory should be accounted. The power consumption of the cache memories will be proportional to the number accesses to the cache memories. We do not distinguish the level of the cache memories, assuming they will consume approximately the same power because they are on the same die. The power consumption of the cache memories are given by:

$$P_{cache} = \varepsilon N \quad (4)$$

where N is the number of cache accesses.

In [8], it is pointed out that the uncore subsystem is not scalable in most SoC architecture. In this case, we can assume that static power consumption is almost not changed with the number of cores. Therefore, the number of active cores affects the dynamic power consumption, while the state of the multicore processor itself affects the static power consumption. With this assumption, the CPU power consumption is modeled as follows:

$$P_{CPU} = P_{dynamic} \times c + P_{static} + P_{cache} = \beta f^3 c + \gamma f + \varepsilon N \quad (5)$$

where c is the number of active cores.

2.2. Power Consumption of Other Components

Many research results reported that most components other than CPU and memory consume almost constant power steadily regardless of the system activities [3, 9]. For example, the difference between power consumption when a network card is idle and active is less than 1 watt [10], and the standard deviation of disk power consumption is very small [3]. Therefore, we simply assume that the power consumption of components in a server other than the CPU and memory is constant at P_{other}.

Memory component, DRAM, is a significant part of the total energy consumption [3, 9]. Like CPUs, DRAM is also a CMOS device. Thus operating DRAM requires energy dissipation as described in Subsection 2.1. However, as the DRAM currently equipped to the contemporary server systems does not provide dynamic frequency scaling, we can assume that the power consumption of a DRAM in idle state is almost not changed [11, 12]. So we can assume the total power consumption of a DRAM is proportional to the number of DRAM operations. The number of DRAM operations can be estimated by measuring the number of misses in the last level cache. So the DRAM power consumption is given by:

$$P_{DRAM} = \omega L \quad (6)$$

where ω is a constant and L is the number of misses in the last level cache.

2.3 The System Power Model

Integrating the power models described above, we developed a power consumption model for the entire server system as follows:

$$\begin{aligned} P_{system} &= P_{CPU} + P_{DRAM} + P_{other} \\ &= \beta f^3 c + \gamma f + \varepsilon N + \omega L + P_{other} \end{aligned} \quad (7)$$

P_{other} is considered as a constant in our model.

Table 1. Average power consumption of the server (Watts)

Program	Number of Cores / Frequency (GHz)	1	2	3	4	5	6	7	8
Simple	1.596	66.29	69.81	72.90	76.47	76.65	76.73	76.82	76.89
	1.729	66.90	70.90	74.75	78.60	78.77	78.93	79.01	79.02
	1.862	67.52	71.92	76.47	80.96	81.22	81.31	81.44	81.57
	1.995	68.23	73.23	78.89	83.88	83.91	84.10	84.22	84.35
	2.128	69.02	74.74	80.84	86.88	87.24	87.47	87.57	87.80
	2.261	69.92	76.73	83.22	90.21	90.58	90.89	91.02	91.30
	2.394	71.43	79.18	86.65	95.01	95.42	95.81	95.94	96.26
RAMspeed	1.596	64.99	68.79	72.78	76.56	76.63	76.86	77.42	77.68
	1.729	65.54	69.82	74.22	78.71	78.59	78.98	78.48	79.77
	1.862	66.12	71.09	75.88	80.10	79.90	80.50	80.86	81.50
	1.995	66.88	72.80	77.34	81.51	81.85	81.53	83.23	83.57
	2.128	67.61	74.44	79.43	83.24	83.71	84.60	84.51	86.07
	2.261	68.31	75.76	80.66	85.48	86.65	86.37	87.48	87.90
	2.394	69.61	77.76	83.48	88.48	89.61	90.64	90.23	91.23

3. Obtaining Parameters of the Power Model

We have measured the actual power consumption of a server to obtain the parameters of the power model described in Section 2. Intel Server System SR1690WB with an Intel Xeon E5620 processor was used as the test machine. It is equipped with a 4GB DDR3 RAM and a 500GB HDD. The Intel Xeon E5620 CPU used here has 4 cores. The CPU supports 7-level frequency: 2.394GHz, 2.261GHz, 1.995GHz, 1.862GHz, 1.729GHz, and 1.596GHz. The power consumption of the server is measured using the power meter HPM-300A, which provides data every about 0.25 second. The power measured is the power consumed by the entire server system including CPU, fans, HDD, main board, and power supply unit, etc. The error in power measure is less than ±0.4%. While the room temperature was maintained at 15˚C.

To obtain the parameters, of the power model, we measure the power consumption of the server system when CPU-intensive jobs are executed on various core-frequency configurations. Since there are 4 cores and 7 levels of frequencies, totally 28 configurations were tested. We have tested two programs: one is a simple program that calculates cumulative sum of a register-stored value and the other is RAMspeed SMP 3.5.0, a benchmark program that measures memory performance. Table 1 shows the measured power in this experiment.

With these results, we calculated the parameters in Equation (7) using regression analysis. Table 2 summarizes the obtained parameters. Because the number of cache accesses and misses are too large, N and L are replaced by $N/10^9$ and $L/10^6$.

Table 2. Parameters calculated for CPU power model

Parameter	Value
β	0.114403
γ	5.323903
ε	1.278825
ω	0.194341
P_{other}	52.34901

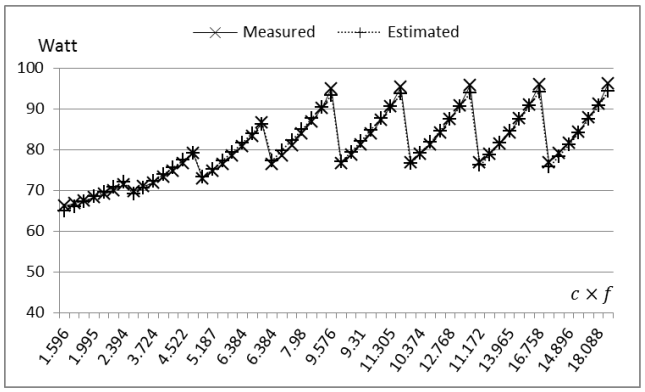

Figure 1. Measured vs. Estimate power consumption (Simple addition)

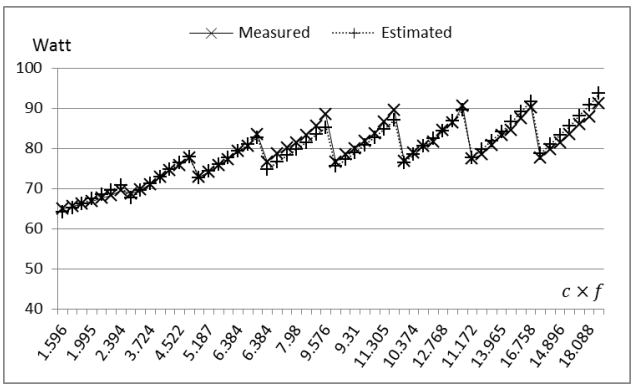

Figure 2. Measured vs. Estimated power consumption (RAMspeed)

Figure 1 and Figure 2 compare the actual power measured with the server and the power estimated using our power model with the parameters in Table 2. As we can see from Figure 1 and Figure 2, our model is accurate when there is small number of DRAM accesses. The average error between the measured data and the model data is about 1.06%, and the maximum error is about 3.64%.

4. Application of the power model

To test the validity of our model, we performed more general benchmark test that utilizes various computing devices such as network card, disk, and DRAM. We choose an HTTP server to be measured software, since it is the most widely used server software. Apache HTTP server benchmarking tool (ab) was used to make requests on the Web server.

As we assumed in our model, we expect the power consumption of network card, disk, and other miscellaneous components can be predicted using the data we obtained in the experiment in Section 3. The measured power consumption is summarized in Table 3.

Table 3. Average power consumption of the system running an HTTP server (Watts)

Number of Cores Frequency (GHz)	1	2	3	4	5	6	7	8
1.596	66.60	70.81	74.93	78.95	79.59	81.03	81.33	81.93
1.729	67.27	72.21	76.92	81.65	82.57	84.28	84.92	85.59
1.862	68.18	73.89	79.31	84.82	85.95	86.89	88.70	89.17
1.995	68.69	75.46	81.96	88.33	89.30	90.56	92.74	93.62
2.128	69.69	77.44	85.02	92.53	94.16	95.12	97.49	98.68
2.261	70.65	79.76	88.06	96.81	98.31	99.75	102.5	103.7
2.394	72.21	82.72	92.12	102.7	104.8	106.3	109.4	111.4

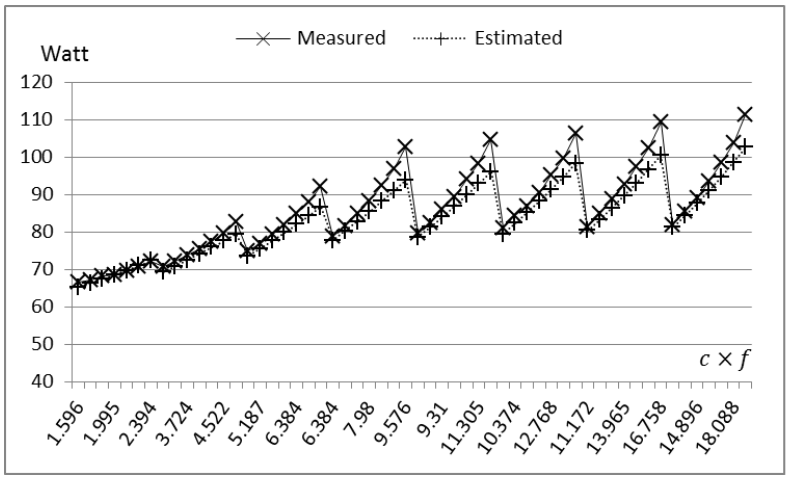

Figure 3. Comparison of the actual measurements and the model calculation

Using our power consumption model, we calculate the power consumption of the server and compare the results with the actual measurement data. Figure 3 compares the measured data and our model prediction. However, the average error is 3.10% and the maximum error is 8.54%, so the accuracy of our model prediction is above 90% in the worst case.

5. Conclusion and Future Work

In this paper, we proposed a simple power model for multicore server systems only with 4 parameters: the current frequency, the number of active cores, the number of cache accesses, and the number of the last level cache misses. Unlike existing power modeling techniques most of which are too complex to be used for real-time application, our power model is simple enough to be used for practical use, while achieving high accuracy. Experimental results show that our model shows over 90% accuracy, 96.9% on the average. However, our model needs to be validated with more application services, because the experiments were performed using artifact workloads. We plan to study our approach further with a data center test bed which has many different kinds of server systems, and it will be our future work.

Acknowledgements

This work was supported by the Incheon National University Research Grant in 2013.

References

[1] M. Weiser, B. Welch, A. J. Demer and S. Shenker, "Scheduling for Reduced CPU Energy", 1st USENIX Conf. on Operating Systems Design and Implementation, (**1994**), pp. 13-23.
[2] S. Rivoire, P. Ranganathan and C. Kozyrakis, "A Comparison of High-Level Full-System Power Models", Power Aware Computing and Systems (HotPower), **(2008)**, pp. 3.
[3] W. L. Bircher and L. K. John, "Complete System Power Estimation Using Processor Performance Events", IEEE Transactions on Computers, vol. 61, no. 4, **(2012)**, pp. 563-577.
[4] P. Langen and B. Juurlink, "Leakage-Aware Multiprocessor Scheduling for Low Power", 20th International Parallel and Distributed Processing Symposium, **(2006)**, pp. 80.
[5] A. Chandrakasan, S. Sheng and R. Brodersen, "Low-Power CMOS Digital Design", IEEE J. of Solid-State Circuit, vol. 27, no. 4, **(1992)**, pp. 473-484.
[6] T. Mudge, "Power: A First Class Design Constraint", IEEE Computer, vol. 34, no. 4, **(2001)**, pp. 52-57.
[7] I. Hong, D. Kirovsky, G. Qu, M. Potkonjak and M. Srivastava, "Power Optimization of Variable Voltage Core-based Systems", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 18, no. 12, **(1999)**, pp. 1702-1714.
[8] V. Gupta, P. Brett, D. Koufaty, D. Reddy and S. Hahn, "The Forgotten 'Uncore': On the Energy-Efficiency of Heterogeneous Cores", 2012 USENIX Annual Technical Conference, Short Paper, **(2012)**.
[9] D. Economou, S. Rivoire, C. Kozyrakis and P. Raganathan, "Full-System Power Analysis and Modeling for Server Environments", 2nd Workshop on Modeling, Benchmarking, and Simulation, **(2006)**, pp.70-77.
[10] R. Sohan, A. Rice, A. W. Moore and K. Mansley, "Characterizing 10 Gbps Network Interface Energy Consumption", IEEE 35th Conference on Local Computer Networks, **(2010)**, pp. 268-271.
[11] T. Vogelsang, "Understanding the Energy Consumption of Dynamic Random Access Memories", 43rd Annual IEEE/ACM International Symposium on Microarchitecture, **(2010)**, pp. 363-374.
[12] H. David, C. Fallin, E. Gorbatov, U. R. Hanebutte and O. Mutlu, "Memory Power Management via Dynamic Voltage/Frequency Scaling", 8th ACM International Conference on Autonomic Computing, **(2011)**, pp. 31-40.

