A Study on Factors Influencing Power Consumption in Multithreaded and Multicore CPUs

[1]VIJAYALAKSHMI SARAVANAN, [2]SENTHIL KUMAR CHANDRAN, [2]SASIKUMAR PUNNEKKAT AND [3]D P KOTHARI

[1&2] Malardalen University, Sweden and [3]VIT University, Vellore, India

vijayalakshmi.saravanan@mdh.se, cskumar@igcar.gov.in, sasikumar.punnekkat@mdh.se and d.p.kothari@vit.ac.in

Abstract: The ever-growing demand for computational power and high performance has led to a rapid growth in the semiconductor industry. This evolution has seen a continuous increase in CPU performance and the number of transistors on a chip has roughly doubled every two years – proving Moore's law. An inevitable consequence when achieving this is that more functional units, deeper pipelining and larger cache sizes have had to be implemented on the CPU chip. The result is a significant increase in the power consumption. Achieving high performance with low power consumption has been the traditional goal in high-end processors. In order to accomplish high performance, multithreaded and multicore CPUs have become the recent trend in semi-conductor technology. The purpose of this paper is to statistically analyze the various factors that affect power, to study their relationship to quantify their influence on power consumption in multithreaded and multicore CPUs. This paper explores the on-chip power modeling simulation techniques with the existing processors and compares the performance and power trade-off between multicore and multithreaded CPUs. In this paper, we also present review/tutorial of recent advancements in power savings through the implementation of power-limiting micro-architectural features (e.g. out-of-order execution, branch prediction, caching and prefetching) in contemporary multi-core processors, such as Intel Nehalem and AMD's Istanbul processors. The results show that the statistical findings on power consumption are encouraging and useful for low power application and power-aware processor designers.

Keywords: Power consumption, Statistical analysis, Power-limiting factors

1 Introduction

Power consumption has become an industry-wide issue for ever-growing computing systems, and one of the key limiting factors for high performance computing. Power aware processors are in great demand for various high end application domains and many scientific applications that are required to achieve high performance. Wattch [35] power-related simulation tool first presented in 2000. Though this tool presents the architectural model to evaluate the power, area and timing there is urging to scale the circuit accurately and model the accurate power dissipation of all sources. CACTI [37] are the tool to analyze the power, area, and timing for memory based architecture. McPAT [36] is an integrated power, area, and Timing tool which model the dynamic power. McPAT also modeled to produce integrated solution for multicore processor power.

Many design techniques to reduce the power consumption on processor design such as

- ASIC design
- Custom design
- process design and
- Micro-architecture has been adopted to reduce the power consumption of the processors.

ASIC design level generally focuses on Register Transfer Level (RTL) optimization [6], and uses a hardware description language (HDL). Custom design techniques are used to achieve high speed with low power [13], and also used to optimize the individual logic cells, the layout and wiring between cells, and other aspects of design. In process design technology, the chips fabricated with the same process may vary in power and speed due to process variation. Moreover, the micro-architecture design plays a vital role on performance and power consumption [5]. Micro-architectural techniques, such as pipelining and parallelism increases through-put of the processor. Pipelining reduces the instructions per clock cycle (IPC), due to high branch mispredictions and other hazards, and thus can reduce the energy efficiency. Deeply pipelined design is optimal when per

pipeline stage is 6 to 8 FO4 (fanout-of-4) inverter delays [19].

The evolution of parallel machines has led to multicore processors. The continuing growth in parallel machines that the chip manufacturers have focused on is, towards raw performance and CPU speed with less power consumption. All the above mentioned tools and techniques are used for low power chip design and evaluate the on-chip power consumption and also used to model the power, area and timing.

This paper describes how the performance enhancement techniques influence the power consumption in multithread and multicore processors, and studies the statistical behavior of the factors involved in power consumption of general purpose and high-performance computers. A detailed study/review of performance and power related design techniques in recent processors such as Intel Nehalem and AMD Istanbul are also included.

The paper structure is as follows: section 2 presents the related work in this field, section 3 covers the overview of factors affecting the power consumption, and a comparison of multithreaded vs. multicore CPU is presented in section 4. Finally, a detailed study of Intel Nehalem and AMD's Istanbul processors are presented in section 5, results and discussions in section 6, and few conclusions are drawn in section 7.

2 Related works

There has been tremendous growth in chip fabrication technology for computing systems. The trade-off between performance and power has a vital role in this growth. Processor design techniques such as ASIC, custom design, process design and micro-architecture design are used to achieve high speed with less power consumption. With the scaling advancement in IC Technology, the amount of power the chip dissipates per unit area is increased, due to increase in transistor density. Power consumption is the major factor even for high-end computing. Among these, the most common power affecting factors are ASIC and micro-architectural design level constraints.

In ASIC design and fabrication technology, power consumption depends on the duration the chip is active. Power efficiency can be achieved through logical circuit gating, process technology, process variation, voltage scaling, cell and wire sizing [29]. In this context, considerable amount of research has been carried out on power reduction. Power reduction can be achieved in voltage scaling by using 2 x 4 parallel data paths [3]. Clock gating is the technique which increases power efficiency when the chip is on. Reducing the supply voltage reduces the sub-threshold leakage cur-rent and gate-oxide tunneling leakage [27] [26]. The voltage drop across sleep transistors takes substantial power due to high capacitance [21]. ASIC design techniques adopt CMOS logic for combinational logic circuits. Pass Transistor Logic (PTL) has high speed and low power logic style [23]. In order to achieve low power by using PTL, PMOS, CMOS logic styles, needs more careful cell design and layout. The power gap between EDA (Electronic Design Automation) and CMOS logic is reduced by adopting the custom design flow techniques.

Downsizing the transistors gives a linear reduction in capacitance, and thus a reduction in dynamic and leakage power. To increase the speed on a critical path high speed combinational logic style can be used [6]. However, when the performance of the pipeline is considered, the CMOS circuitry is preferred to have high power consumption per operation. All these logic circuitry have high leakage power than CMOS logic circuits. In order to avoid leakage through NMOS transistors, it is required to keep PMOS transistor alive [30], with a trade-off between power and clock gate sizing, i.e., to reduce the delay, increase the gate sizes on a critical path and capacitance on a chip.

The logic circuit delay can be determined by the various factors such as logic style, layout, and process technology and process variation. There is a smaller reduction in delay while reducing the P/N ratio which provides a substantial reduction in power consumption [15]. Power compiler was able to re-duce the power consumption by 26 % and reduces the chip area by 12 % for no delay penalty at 325MHz of clock frequency. The power consumption and power density are also reduced by using deeper submicron technologies [25]. There are number of sources of process variation that influences the power consumption such as channel width, transistor wire-length and width, proximity effects and wafer defects. Wire length is also another power causing factor in custom design. Optimizing the wire width size reduces the clock net power, thus saves the total power [7]. The

power consumption due to interconnection has significant impact on wire length. By using copper wires, the wire delay and IR drop in wire could be avoided [1]. Narendra et al showed that silicon-on-insulator (SOI) was faster than bulk CMOS. The speed varies while using the SOI of 14 to 28% for 0.18 m CMOS logic gates. The total power was 30% lower at the same delay, but the leakage was 1.2% to 20% larger [14]. Micro-architectural design choices and algorithms also plays a vital role in power consumption, it reduces the power by an order of magnitude [16].

Low power consumption is essential for high performance computing, embedded applications and scientific applications. A substantial amount of research has been carried out on power reduction in multicore. In this paper, we focus on power-limiting factors and analyze their impact through statistical analysis. An exclusive study/review on recent advancements in power saving techniques of currently used commercial processors such as Intel Nehalem and AMD Istanbul are also discussed.

3 Overview of Factors Affecting the Power Consumption

Total power consumption in a circuit includes dynamic, short-circuit, and leakage power [5]. All these three significantly impact more on total power consumption in multicore processors.

$$P_{total} = P_{dynamic} + P_{short\text{-}circuit} + P_{leakage}$$

$$P_{dynamic} = P_t . C_l . V_{dd}^2 . f_{clk}$$

$$P_{Leak} = K . 10^{Vth/S} . V_{dd}$$

Dynamic power is due to charging and discharging of the transistors associated with the capacitor. A significant reduction in dynamic power can be achieved by reducing unnecessary switching activity by appropriately selecting clock gate. Dynamic power in a modern processors due to the following components: Gate (34%), Interconnect (51%) and Diffusion (15%).The general approaches to low power design in dynamic component is to lower V_{dd} , reduce C_l and lower P_t .Leakage power is the power consumed when the logic circuit is idle and it contributes more to average power especially when a chip has been idle for long. If the gate thickness increases, gate leakage current increases exponentially. The general approaches to low power design in leakage component such as greater Vth, reduce the transistor size (K) and reduce the supply voltage V_{dd}. Short circuit power is consumed when both the pull-up and pull-down devices in a CMOS circuit are partially on for a small, but finite amount of time. It typically contributes less than 10% to total dynamic power [5].

The major sources of on-chip power consumption of circuit level components are clock tree, registers, control and data path logic circuitry and memory. Some of the other minor factors that contribute to power consumption are logic style, logic design, cell and wire sizing, floor planning, placement and technology mapping. However, the on-chip power consumption factors that contributes to multicore processors fabricated using a deep-submicron technology when a circuit is active is:

- Transistor density
- Voltage
- Clock frequency
- Die size
- Cache size
- Process technology and Process variation
- Number of cores
- Number of threads and
- Micro-architecture

In our analysis we have taken the above factors and performed a statistical analysis to identify the major influencing factors on power consumption and low power management techniques are implemented in recent processors.

4 Comparison of Power Consumption of Multithreaded versus Multicore CPUs

4.1 Multithreaded versus Multicore CPUs

Performance has been the traditional goal of processor technology. During the past few decades, more focus was given to achieve performance enhancement with low power. Performance enhancement can be achieved through multithreaded and multicore processors by

executing multiple threads concurrently, to exploit thread level parallelism. Multithreaded processors exploit thread level parallelism by executing multiple threads at a time on cores which provides benefits for greater resource utilization, better instruction throughput, tolerates long latency events wherein multicore processors implements multiple processing on a single chip. These CPUs are known as chip multiprocessing (CMP) cores.

As a tremendous growth in transistor technology that increases manufacturing economies and frequency, power consumption also increases proportionately. In order to achieve high level of parallelism, combined design techniques are used in GPU processors like ATI RV770, and NVIDIA GT200. To bridge the gap between high performance and low power and to facilitate high end processors, larger functional units, deeper pipelines, and larger caches are used on a chip which in turn increases the number of transistors on a chip. By Moore's law, more transistors on a chip increases CPU speed and power consumption. C.J Anderson et al., studied that up to 70% of the total on-chip dynamic power and major chip power is from the clock. Though many statistical modeling tools and techniques available to evaluate the performance, power, area and timing, our main focus on this paper is to identify the key factors involved in power consumption in multithreaded and multicore processors and perform a statistical regression analysis.

4.2 Regression Analysis

Multiple regression is a statistical technique to investigate the impact of various simultaneous influences upon a single dependent variable. The form of multiple regression equation is

$$Y = b_1X_1 + b_2X_2 + ::: + b_nX_n + C$$

Where b_i's are regression coefficients that represent the amount the dependent variable Y changes when the corresponding independent variable X_i changes 1 unit. C is a constant where the regression line intercepts the Y-axis. The multiple correlations that represent the percent of variance in the dependent variable Y explained collectively by all of the independent variables are represented as R^2.

In order to perform the statistical analysis, we have taken the published data from their official website [10] [12] [8] [9] for multithreaded and multicore CPUs are given in Tables 1 and 2 respectively. Wherever a range of value is provided (as indicated with # in the tables), the upper bound value is considered for regression analysis [33] [34].

The regression equation based on the collected data for multithreaded CPU is:

$$Y = (174.5\ X_1) + (0.87\ X_2)\ (132.5\ X_3)\ (0.13\ X_4) + (21.3\ X_5)\ (0.47\ X_6)\ (0.75\ X_7) + (1.2\ X_8)\ 556.2$$

The coefficient of determination, $R^2 = 0.94$ indicates that there is a good relation between the factors (independent variables) and power (dependent variable).

The regression equation for multicore CPU is:

$$Y = (72.93\ X_1)\ (0.24\ X_2)\ (50.1\ X_3) + (0.11\ X_4) + (9.78\ X_5)\ (1.54\ X_6)\ (1.48\ X_7)\ (0.039\ X_8)\ 114.12$$

The coefficient of determination, $R^2 = 0.86$ indicates that there is a good relation between the factors (independent variables) and power (dependent variable). The results of both multithreaded and multicore CPUs indicate that Clock frequency (X_1), Voltage (X_3) and No. of cores (X_5) are significant contributors among the 8 factors considered shown in Tables 1 and 2.

4.3 F-Test:

In order to find the degree of most influencing power causing factor, we performed the regression model and to quantify it through F-test analysis. F-test could be done with two different variables. In our analysis, we have taken power with all the mentioned power causing factors and analyzed the results are shown in Tables 3 and 4. In general F-test has the following parameters. Mean - The mean of each of the samples, Variance - The variance of each of the samples, Observations - The number of values in each of the samples, df - The Degrees of Freedom for each of the samples. F - The F Statistic. An F statistic close to 1 provides evidence that the sample variances are equal. The higher the F statistic the less likely that the null hypothesis is true.

Table 1 Characteristics of Multicore CPUs

Processor(s)	Clock Freq(GHz)	Process (nm)	Voltage (V)	Transistors (no)	Cores (no)	Cache size (MB)	Threads (no)	Die Size (mm2)	Power (watts)
AMD Phenom™ II X4	3.4	45	1.425	450	4	0.5	4	285	140
Dual-Core AMD Opteron™	3.2	90	1.4	758	2	1	2	285	125
IBM Cell	3.2	45	1.8	241	8	0.5	2	221	110
Intel® Core™ i3-540 Processor	3.06	45	1.485	382	2	4	4	81	73
Intel® Core™ i5-750 Processor	2.66	45	1.315	774	4	8	4	296	95
Intel® Core™2 Duo Processor E8600	3.33	45	1.3625	410	2	6	2	107	65
Intel® Core™2 Extreme Processor QX9650	3	45	1.3625	820	4	12	4	214	130
Intel® Core™2 Extreme Processor QX9770	3.2	45	1.3625	820	4	12	4	214	136
Intel® Core™2 Extreme Processor QX9775	3.2	45	1.212	820	4	12	4	214	150
Intel® Core™2 Quad Processor Q8200	2.33	45	1.3625	456	4	4	4	164	95
Intel® Core™2 Quad Processor Q9505S	2.83	45	1.3625	456	4	6	4	164	65
Intel® Pentium® D Processor 935	3.2	65	1.3375	376	2	4	2	162	95
Intel® Pentium® Processor E6600	3.06	45	1.3625	228	2	2	2	82	65
Intel® Pentium® Processor Extreme Editio	3.73	65	1.3375	376	2	4	2	162	130
Intel® Pentium® Processor G6950	2.8	32	1.4	382	2	3	2	81	73
Intel® Xeon® Processor X3470	2.93	45	1.4	774	4	8	8	503	95
Intel® Xeon® Processor X5492	3.4	45	1.212	820	4	12	4	214	150

Table 2 Characteristics of Multithreaded CPUs

Processor(s)	Clock Freq.(GHz)	Process (nm)	Voltage (V)	Transistors (no)	Cores(no)	Cache size (MB)	Threads(no)	Die size(mm)2	Power (watts)
Intel® Itanium® Processor	1.6	130	1.25	592	1	9	1	432	122
Intel® Itanium® Processor 9310	1.6	90	1.25	2000	2	10	4	596	130
Intel® Itanium® Processor 9330	1.46	90	1.25	2000	4	20	8	596	155
Intel® Xeon® Processor L5530	2.4	45	1.35	731	4	8	8	263	60
Intel® Xeon® Processor E5540	2.53	45	1.35	731	4	8	8	263	80
Intel® Xeon® Processor X5570	2.93	45	1.35	731	4	8	4	263	95
Intel® Core™ i7-960 Processor	3.2	45	1.375	731	4	8	8	263	130
Intel® Core™ i7-870 Processor	2.93	45	1.4	774	4	8	8	296	95
Intel® Core™ i7-860S Processor	2.53	45	1.375	774	4	8	8	296	82
IBM Power 5	1.65	65	1.2	276	2	32	2	389	167
IBM Power 6	4.7	65	1.3	790	2	32	2	341	570
SUN Ultra SPARC T1	1.4	45	1.3	300	8	16	32	379	72
SUN Ultra SPARC T2	1.6	65	1.5	500	8	12	64	342	84
SUN Ultra SPARC IV	1.5	130	1.35	295	2	64	2	300	90
SUN/Fujitsu SPARC64 VII	2.88	65	1.35	600	4	64	8	365	135
SUN Rock	2	65	1.35	321	16	32	256	396	250

Table 3 F-Test on Multicore CPUs

F-Test on Power vs. Die size			F-Test on Power vs. Clock Freq		F-Test on Power vs. Process size	
	Power	Die size	Power	Clock Freq max	Power	Process size
Mean	103.35294	202.88235	103.35294	3.0723529	103.35294	49.235294
Variance	867.99265	10808.36	867.99265	0.0980191	867.99265	168.06618
Observations	17	17	17	17	17	17
Df	16	16	16	16	16	16
F	0.0803075		**8855.3403**		5.1645885	
P(F<=f) one-tail	3.72E-06		1.69 9E-28		0.0010455	
F Critical one-tail	0.4285438		2.3334836		2.3334836	
F-Test on Power vs. voltage			**F-Test on Power vs.**		**F-Test on Power vs.**	
	Power	Voltage (Min)	Power	Voltage(Max)	Power	Transistors
Mean	103.35294	1.0720588	103.35294	1.3829412	103.35294	549.58824
Variance	867.99265	0.0871158	867.99265	0.0161721	867.99265	49915.632
Observations	17	17	17	17	17	17
Df	16	16	16	16	16	16
F	9963.664		53672.365		0.0173892	
P(F<=f) one-tail	6.62E-29		9.34E-35		4.21E-11	
F Critical one-tail	2.3334836		2.3334836		0.4285438	
F-Test on Power vs. Cores			**F-Test on Power vs.Cache size**		**F-Test on Power vs. Threads**	
	Power	Cores	Power	Cache size	Power	Threads
Mean	103.35294	3.4117647	103.35294	5.8235294	103.35294	3.4117647
Variance	867.99265	2.3823529	867.99265	17.373162	867.99265	2.3823529
Observations	17	17	17	17	17	17
Df	16	16	16	16	16	16
F	**364.34259**		49.961697		**364.34259**	
P(F<=f) one-tail	1.99E-17		1.25E-10		1.99E-17	
F Critical one-tail	2.3334836		2.3334836		2.3334836	

Table 4 F-Test on Multithreaded CPUs

F-Test on Power vs. Die size			F-Test on Power vs. Clock Freq		F-Test on Power vs. Process size	
	Power	Die size	Power	Clock freq	Power	Process siz
Mean	144.875	361.25	144.875	2.358125	144.875	67.5
Variance	15045.45	11279.133	15045.45	0.7557896	15045.45	826.66667
Observations	16	16	16	16	16	16
Df	15	15	15	15	15	15
F	1.333919		19906.93		18.200141	
P(F<=f) one	0.2919144		1.90E-29		5.80E-07	
F Critical one	2.4034471		2.4034471		2.4034471	
F-Test on Power vs. Voltage			**F-Test on Power vs. Voltage**		**F-Test on Power vs. Transistors**	
	Power	Voltage min	Power	Voltage ma	Power	Transistors
Mean	144.875	1.0131688	144.875	1.33125	144.875	759.125
Variance	15045.45	0.0798255	15045.45	0.0052083	15045.45	270404.65
Observations	16	16	16	16	16	16
Df	15	15	15	15	15	15
F	**188479.3**		**2888726.4**		0.0556405	
P(F<=f) one	9.05E-37		1.16E-45		6.32E-07	
F Critical one	2.4034471		2.4034471		0.4160691	
F-Test on Power vs. Cores			**F-Test on Power vs. Cache size**		**F-Test on Power vs. Threads**	
	Power	Cores	Power	Cache size	Power	Threads
Mean	144.875	4.5625	144.875	21.1875	144.875	26.4375
Variance	15045.45	13.0625	15045.45	363.09583	15045.45	3997.7292
Observations	16	16	16	16	16	16
Df	15	15	15	15	15	15
F	**1151.805**		41.436581		3.7634991	
P(F<=f) one	3.60E-20		1.79E-09		0.0073163	
F Critical one	2.4034471		2.4034471		2.4034471	

4.4 Cache size and Design

We also consider cache design as a factor that affects the power consumption of a processor. Often, a larger cache size and increased cache associativity also causes decreases the cache miss rate, which in turn decreases the total cache access time, and thus reducing the total power consumption of the processor.

We have considered the following cache size and its associativity in processors:

- Intel Itanium Processor 9300 series, Intel Core i7 Processor, and Intel Pentium G6950 Processor use shared L3 cache to enhance cache communications in multi-core. Intel Core2 Duo Processor use L2 shared cache for performance enhancement.
- In IBM Power 5 processors, each processor core has a separate L1 instruction and data cache. L3 cache is a 36 MB victim cache of the L2 cache and, it is shared by all the hardware threads of both the processor cores of the POWER5 chip. IBM Power 6 has L1 I-cache (instruction), D-cache (data) size of 64KB, L2 and L3 victim-cache and shared by all the hardware threads of both processor cores. It has 4-way set associativity I-cache for fast address translation.
- SUN Ultra SPARC T1, T2 and IV have L2 cache and 16-way set associativity. Sun Rock processor has the cores in clusters and shares the two I-cache and D-cache. SUN/Fujitsu SPARC64 VII has sectored L1 2-way instruction and data cache and index hash sectored L2 10-way cache line.
- Dual-core AMD Phenom II X4 has L1 and L2 cache per core and L3 cache shred between all cores and the core of the Quad-Core AMD OpteronTM processor code named Barcelona introduce a unique on-die L3 cache for latency reduction.

In Tables 1 and 2, cache sizes implemented for some of the common processors are depicted (*). In Tables 1, 2 and 3, 4 we have taken a various power causing factors in order to find the most important power influencing factors. After our regression and F-Test we found that the key influencing factors as voltage and frequency as claimed in section 6.

5 Architectural Design Techniques of Power Efficient Processors

On the architectural level, the multicore processor is decomposed into major components such as cores, NoCs, caches, memory controllers and clocking [35]. Previous research on high performance computing was motivated by performance demands. However, the recent trend in low-powered devices is forcing computer architects to re-evaluate the architectural features and micro-architecture techniques for energy efficiency in general purpose and multicore CPUs. In the statistical analysis carried out, we identified three power limiting factors: clock frequency, voltage and number of cores. In the next two sub-sections, we dwell upon some of the power saving techniques and their implementation in recent multicore processors like Intel Nehalem and AMD Istanbul processors. This paper reports on a study of power management techniques in multithreaded and multicore CPUs focusing on the Intel Nehalem and AMD Istanbul as case studies.

5.1 Overview of Intel Nehalem Architecture

For several decades, the use of increasing number of transistors has led to faster computation, but power-hungry processors. In the recent Nehalem architecture shown in Figure 1 [28], as in earlier generation of dual core and core 2 processor series, Intel continued the use multiple cores on a single die. Nehalem architecture uses multiple cores to improve power and memory management. In its micro-architecture design, a newly shared L3 cache is added and is shared across all cores. This lead to significant performance enhancement as it reduced the traffic to the processor cores. Nehalem's L3 cache shown in Figure 1 can be fully shared, and all applications can use entire cache. It is designed to avoid unnecessary core searches to reduce latency, thus improves the performance. A new second level processor cache called Translation Look aside Buffer (TLB) is introduced to improve the speed of virtual address translation. Power savings in the Nehalem processor are attributed to various factors such as clock frequency, voltage and micro-architecture level techniques called branch prediction, out-of-order execution and cache organization. It is also evident from our statistical analysis that clock frequency and voltage play a vital role in power consumption. We found that the recent developments on power savings in the Nehalem processor also used adaptive techniques, called adaptive clocking system (clock frequency), to reduce the power and improve clock skew margins. It has a scalable performance and power efficient architecture enabled by adaptive duty cycle and adaptive frequency system (AFS) for low power at same frequency. It uses PLL (Phase Locked Loop) as shown in Figure 2 for clock distribution, and filter DLL for higher sampling frequencies [24]. Quick Path Interconnect (QPI) technology shown in Figure 1 features the new system micro architecture that consists of integrated memory controller (IMC), DDR, and PCI to deliver better performance, optimized memory organization to avoid unnecessary memory traffics [11].

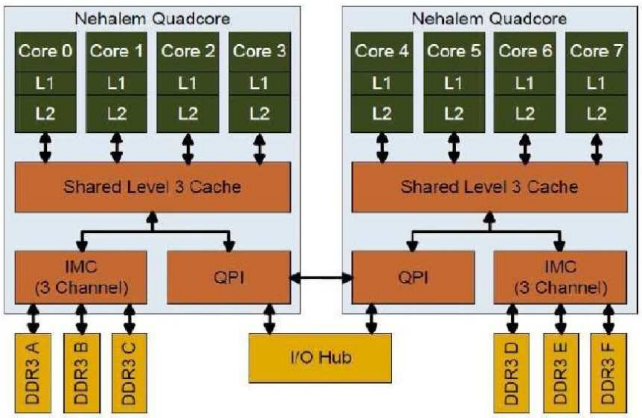


Fig. 1. Nehalem Eight- Core Architecture

5.1.1 Micro-architectural Design Features

McPAT [36] is the first modeling framework which completely describes the multiocre/manycore processor from the architectural perspectives. McPAT's authors believed that the architectural factors also important to model the efficient power, area and timing framework. In this way we also explored/reviewed the architectural design factors and its impact on power consumption of the recent advanced processors.

- Branch prediction: It is a technique which is used to extract greater performance on pipelined processors. Mispredicted branches increase the stall cycles, thus increases energy consumption. An enhanced branch prediction technique called second level Branch Target Buffer (BTB) to handle branch mispredictions and by adopting pointers for call and return instructions are used in Nehalem to gain significant performance enhancement and less power consumption.
- Out-of-order execution: Greater parallelism can be achieved in software code by increasing the amount of instructions that can be executed in out-of-order. In Nehalem, the increased size of the out-of-order window and scheduler is used to make use of instruction cycles efficiently. The size of the other buggers in the core was also increased to ensure the performance improvement.
- Cache and memory organization: The memory system is a significant source of power consumption. Intel Nehalem processor has a new cache organization protocol called MESIF (Modified, Exclusive, Shared, Invalid, Forward) in order to avoid unnecessary memory traffics [22] [20]. This reduces frequent instruction accesses from main memory and decreases power consumption. In Nehalem, triple-channel on-chip memory controller shown in Figure 1 is used to provide more predictable processor performance that will run the processor faster with less power consumption. In earlier generation processors, off-chip memory controller is used for reducing memory latency access.

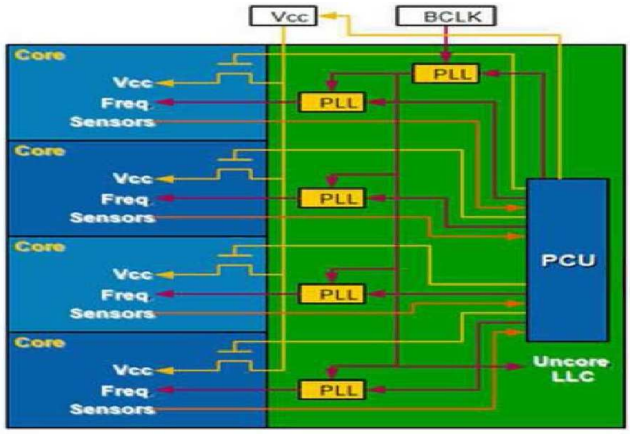


Fig. 2. Nehalem Architecture Power Control Unit

An intelligent way for power management is to monitor power consumption in multi-processor systems and identify those that are not fully utilized. Nehalem architecture includes integrated power gates that enables idling cores to reduce to near-zero power and is independent of other operating cores. Depending on the present workload of the processor, a feature called automated low power state maintains the processor and memory to lowest available power states which greatly enhances the power management in Nehalem [17].

5.2 Overview of AMDs Istanbul Architecture

Istanbul processor is based on the earlier generation of quad-core architecture. Though the structure of the cores has not been modified, the new 6-core Istanbul provides a significant performance improvement with the same power consumption due to the use of a new L3 cache shared access across the processor cores [18]. The main features of Istanbul processor are: Use of snoop filter, Higher interconnect bandwidth, Lower cache coherency latency

5.2.1 Micro-architectural Design Features

- Branch prediction: An improved data cache support for two 64-bit operations per cycle, a

latency of 3 cycles, and the instruction cache has advanced branch pre-diction implemented in Istanbul's processor for performance enhancement and less energy consumption.

- Out-of-order execution: The Istanbul processor has three pairs of integer execution units and address generation units. A 32-entry integer scheduler takes cares of the integer computations. Similarly, for floating point units, a FPU (floating-point unit) stack map receives the floating point instruction from the instruction control unit and a 120 element FPU scheduler takes care of floating point ordering and computations. The stack manipulation is in the instruction streaming efficiently handled by stack optimizer to improve the instructions per cycle.

- Cache and memory organization: An efficient cache memory technology is used in Istanbul, called smart fetch technology, wherein the operating and halt states of the system are identified and before an inactive core enters a halt state, the processor shares its L1 and L2 cache contents with L3 caches. An independent dynamic core technique is used where clock frequencies of individual cores can be adjusted depending on the requirement of applications.

An efficient power management technology is introduced in Istanbul, called cool core technology, wherein the unused parts of the processor including L3 cache, can be powered off. Dual dynamic power management enables independent power supply to the cores.

6 Results and Discussion

Fig.3 & 4 shows that the results from regression and F-test analysis based on the power consumption factors considered for multithreaded and multicore CPUs are shown in Tables. Though many factors are involved in power consumption of CPUs, we have focused on major factors and performed the statistical analysis to study their sensitivity. Table 2&4 indicates the key influencing factors as voltage and frequency on multithreaded CPUs. We have considered the number of threads as one of the factors influencing power consumption on multithreaded CPU, as multithreading provides significant performance advantage over the conventional techniques [31] [2] [32]. Further, in multi-threaded CPU there is a relative variation in performance and Power as the number of thread varies. Table 1 and 3 suggests that the key factors are voltage, frequency and number of cores on multicore CPUs. In multicore CPUs it is clearly seen that, if the cores are simplified, power consumption decreases linearly [4].

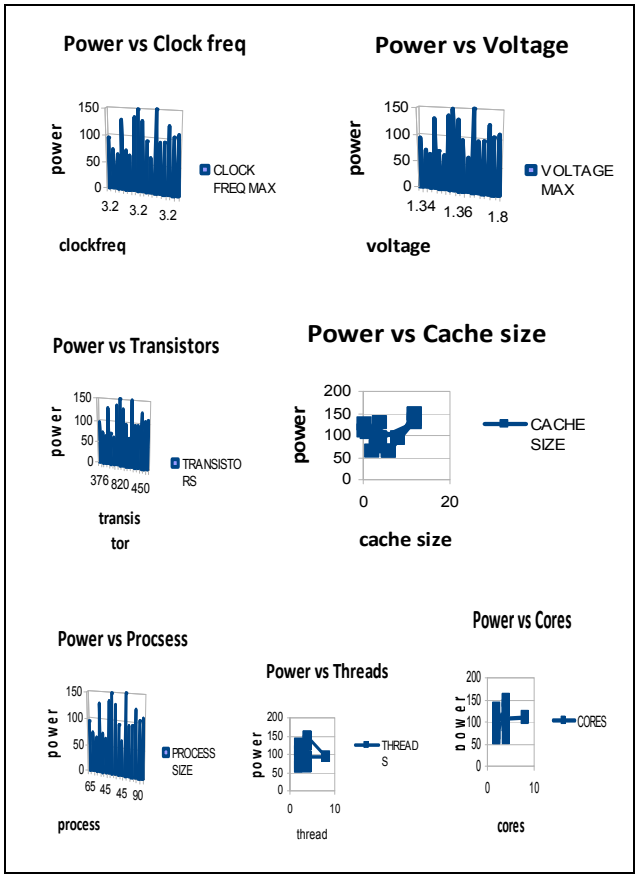
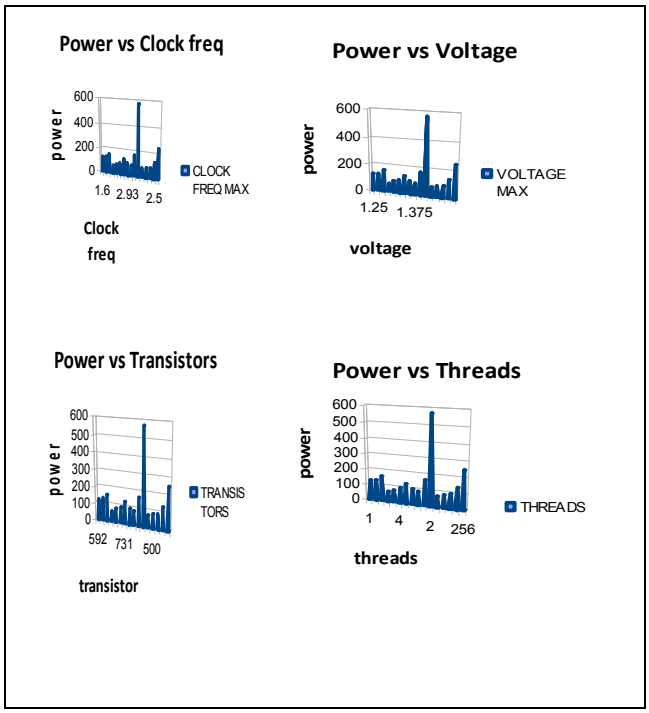


Fig. 3 Relationship between Power vs. Various power consumption factors of Multicore CPUs



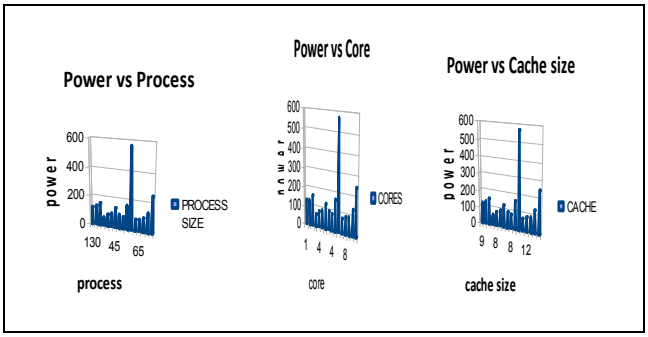


Fig. 4 Relationship between Power vs. Various power consumption factors of Multithreaded CPUs

As can be seen from our analysis, the power consumptions are highly dependent on the different domains in which the processors are meant to be used. The architectural design principles incorporated into the processors are, however, surprisingly equal. The large differences in the power consumption are mainly assumed to be due to the clock frequencies, since higher clock frequency demands higher power consumption. When increasing the clock frequency, in general, the voltage used to drive the processor must also be increased.

7 Conclusion and Future Enhancements

With the emergence of high performance computing with low powered devices, it is important to understand the micro-architectural design and factors affecting power and their characteristics among multithreaded and multicore CPUs. In this paper, we have identified some of the major power affecting factors and provided a statistical view of these factors contributing to the power consumption in evidence with various power, area and timing simulator tools such as Wattch [35], McPAT [36] and CACTI [37]. We have also investigated/reviewed how these factors influence the power saving techniques implemented in recent advanced processors such as Intel Nehalem and AMD Istanbul. In general, all these modeling tools and techniques are insight into general view on power, area and timing. But, in our novel approach Table 1, 2, 3 and 4 we strongly provided the statistical solution of on-chip major power influencing factors and also the authors strongly believes that it can be used in near future by the researchers may focus only on these particular factors and reduce the clock frequency and voltage in order to achieve low power on multicore processor design.

8 Acknowledgement

Vijayalakshmi Saravanan would like thank the EURECA project and the reviewers for their time and valuable suggestions.

References

1. Chang A. VLSI Data path Choices: Cell-Based Versus Full-Custom, *SM Thesis*, Massachusetts Institute of Technology, February 1998.
2. Robert Alverson, David Callahan, Daniel Cummings, Brian Koblenz, Allan Porterfield, and Burton Smith. The Tera computer system. In ICS '90: *Proceedings of the 4th international conference on Supercomputing*, pages 1–6, New York, NY, USA, 1990. ACM.
3. Azeez J. Bhavnagarwala, Blanca L. Austin, Keith A. Bowman, and James D. Meindl. A minimum total power methodology for projecting limits on CMOS GSI. *IEEE Trans. Very Large Scale Integer. Syst.*, 8(3):235–251, 2000.
4. Shekhar Borkar, Tanay Karnik, Siva Narendra, Jim Tschanz, Ali Keshavarzi, and De.Vivek. Parameter variations and impact on circuits and microarchitecture. In *DAC '03: Proceedings of the 40th annual Design Automation Conference*, pages 338–342, New York, NY, USA, 2003. ACM.
5. Anantha P. Chandrakasan and Robert W.Brodersen. Minimizing Power Consumption in Digital CMOS Circuits. *Proceedings of the IEEE, 83(4):498–523, 1995.*
6. David Chinnery and Kurt Keutzer. Closing the Gap between ASIC & Custom: Tools and Techniques for High-Performance ASIC Design. 2002.
7. J. Cong, C. Koh, and K. Leung. Simultaneous buffer and wire sizing for performance and power optimization. In *ISLPED '96: Proceedings of the 1996 inter-national symposium on Low power electronics and design, pages 271–276, Piscataway, NJ, USA, 1996. IEEE Press.*
8. AMD Corporation. "AMDGlobal Provider of Innovative Microprocessor Solutions details", available at *http://www.amd.com/us-en/processors/,* 2008.
9. IBM Corporation. "IBM Products details", available at *http://www.ibm.com/products/us/en/, 2008.*
10. Intel Corporation. "Intel Processors" details, available at *http://www.intel.com/products/processor/. Technical report.*
11. Intel Corporation. An Introduction to the Intel Quick-Path Interconnect, *Document Number: 320412-001US*, January 2009.
12. Sun Corporation. "Sun Microelectronics Products details", available at *http://www.sun.com/processors/,* 2008.
13. Montanaro et al. A 160-MHz, 32-b, 0.5-W CMOS RISC microprocessor. *Digital Tech. J., 9(1):49–62, 1997.*
14. Narendra.S et al. Comparative Performance, Leakage Power and Switching Power of Circuits in 150 nm PD-SOI and Bulk Technologies Including Impact of SOI History Effect. *Symposium on VLSI Circuits, Digest of Technical Papers*, pages 217–218, 2001.
15. Richardson N. et al. The iCORETM 520MHz Synthesizable CPU Core: Chapter 16 of Closing the Gap between ASIC and

Custom. In *Proceedings of the 39th annual Design Automation Conference, 2002.*
16. Singh D. et al. Power-Conscious CAD Tools and Methodologies: a Perspective. *Proceedings of the IEEE, 83(4):570–594, 1995.*
17. Gavrichenkov. First Look at Nehalem Microarchitecture. 2008. *http:// www.xbitlabs.com/articles/cpu/ display/nehalem-microarchitecture. html.*
18. Paul G. Howard. Six-Core AMD Opteron Processor. 2009. *http://www.microway.com/ pdfs/microway_istanbul_whitepaper_ 2009-07.pdf.*
19. M. S. Hrishikesh, Doug Burger, Norman P. Jouppi, Stephen W. Keckler, Keith I. Farkas, and Premkishore Shivakumar. The optimal logic depth per pipeline stage is 6 to 8 FO4 inverter delays. In ISCA '02: *Proceedings of the 29th annual international symposium on Computer architecture*, pages 14–24, Washington, DC, USA, 2002. IEEE Computer Society.
20. Herbert H. J. Hum and James R. Goodma. Forward State for [] Use in Cache Coherency in a Multiprocessor System. *United States Patent*, 6,922,756, July 2000.
21. Shinichiro Mutoh; Satoshi Sigematsu; Yasuyuki Mat-suya; Hideki Fukuda; Takao Kaneko and Junzo Ya-mada. 1-V Power Supply High-Speed Digital Cir-cuit Technology with Multithreshold-Voltage CMOS. *IEEE Journal of Solid State Circuits*, 31(11):1795– 1802, 1996.
22. Brian W. Knotts. Coherent Copy back Protocol for Multi-Level Cache Memory Systems. *United States Patent,* 5,671,391, September 1997.
23. L. Bisdounis ; D. Gouvetas ;O. Koufopavlou. A comparative study of CMOS circuit design styles for low-power high-speed VLSI circuits. *International Journal of Electronics,* 84(6):599–613, 1998.
24. Nasser Kurd and Praveen Mosalikanti. Power-efficient scalable multi-core and high-speed IO clocking architecture. In *Proceedings of 2009 CMOS Emerging Technologies Workshop*, Vancouver, BC, Canada, 2009.
25. Tadahiro Kuroda. Low-Power, High-Speed CMOS VLSI Design. In ICCD '02: *Proceedings of the 2002 IEEE International Conference on Computer Design: VLSI in Computers and Processors (ICCD'02)*, page 310, Washington, DC, USA, 2002. IEEE Computer Society.
26. Dongwoo Lee. and David Blaauw. Static leakage reduction through simultaneous threshold voltage and state assignment. In DAC '03: *Proceedings of the 40th annual Design Automation Conference*, pages 191– 194, New York, NY, USA, 2003. ACM.
27. Dongwoo Lee, Wesley Kwong, David Blaauw, and Dennis Sylvester. Simultaneous Subthreshold and Gate-Oxide Tunneling Leakage Current Analysis in Nanometer CMOS Design. In ISQED '03: *Proceedings of the 4th International Symposium on Quality Electronic Design, page 287, Washington*, DC, USA, 2003. IEEE Computer Society.
28. Daniel Molka, Daniel Hackenberg, Robert Schone, and Matthias S. Muller. Memory performance and cache coherency effects on an Intel Nehalem multiprocessor system. In PACT '09: *Proceedings of the 2009 18th International Conference on Parallel Architectures and Compilation Techniques*, pages 261–270, Washington, DC, USA, 2009. IEEE Computer Society.
29. Sumitkumar N. Pamnani, Deepak N. Agarwal, Gang Qu, and Donald Yeung. Low Power System Design by Combining Software prefetching and dynamic voltage scaling. *Journal of Circuits, Systems, and Computers*, 16(5):745–767, 2007.
30. Dennis Sylvester and Kurt Keutzer. Getting to the bottom of deep submicron. In ICCAD '98: *Proceedings of the 1998 IEEE/ACM international conference on Computer-aided design*, pages 203–211, New York, NY, USA, 1998. ACM.
31. Dean M. Tullsen, Susan J. Eggers, and Henry M. Levy. Simultaneous multithreading: Maximizing on-chip parallelism. *Proceedings of 22nd Annual International Symposium on Computer Architecture*, pages 392–403, 1995.
32. Dean M. Tullsen, Susan J. Eggers, Joel S. Emer Y, Henry M. Levy, Jack L. Lo, and Rebecca L. Stamm Y. Exploiting choice: Instruction fetch and issue on an implementable simultaneous multithreading processor. In *Proceedings of the 23rd Annual International Symposium on Computer Architecture,* pages 191–202, 1996.
33. Lee and Brooks, "Accurate and Efficient Regression Modeling for Micro architectural Performance and Power Prediction", *ASPLOS'06*.
34. Matthew Curtis-Maury, et. al., Prediction-Based Power-Performance Adaptation of Multithreaded Scientific Codes, *TPDS (October 2008 issue)*.
35. D.Brooks, V.Tiwari, and M.Martonosi, " Wattch: a framework for architectural-level power analysis and optimization," in *ISCA,2000*.
36. Sheng Li, Jung Ho Ahn, Richard D. Strong, Jay B. Brockman, Dean M.Tullsen, Norman P. Joupp," McPAT: An Integrated Power, Area, and Timing modeling Framework for Multicore and Manycore Architectures," *MICRO'09*.
37. S.Thoziyoor, J. Ahn, M.Monchiero, J. Brockman and N.Jouppi, "A Comprehensive Memory Modeling Tool and Its application to the Design and Analysis of Future Memory Hierarchies," in *ISCA, 1996.*