Temperature and Process Variations aware Power Gating of Functional Units

Deepa Kannan†, Aviral Shrivastava†, Vipin Mohan‡, Sarvesh Bhardwaj‡, Sarma Vrudhula‡

† Compiler and Microarchitecture Laboratory,
School of Computing and Informatics, Arizona State University, Tempe, AZ 85281 USA
{ deepa.kannan, aviral.shrivastava}@asu.edu
‡VLSI Electronic Design Automation Laboratory
School of Computing and Informatics, Arizona State University, Tempe, AZ 85281
{vipin.mohan, sarvesh.bhardwaj, vrudhula}@asu.edu

Abstract

Technology scaling has resulted in an exponential increase in the leakage power as well as the variations in leakage power of fabricated chips. Functional units (FUs), like Integer ALUs are regions of high power density and significantly contribute to the variation in the whole processor power consumption. Hence, it is important to reduce both the power consumption and the variation in power consumption of the FUs. Among existing FU power reduction techniques, power gating (PG) has been most effective. In this paper, we introduce a leakage sensor inside the FUs and propose a temperature and process variation aware power gating scheme, Leakage Aware Power Gating (LA-PG). Our experimental results demonstrate that LA-PG results in 22% reduction in mean and a 25% reduction in standard deviation of the ALU energy consumption when compared to existing power gating techniques, without significant performance penalty.

1. Introduction

Ever increasing performance demand of electronic devices has been the primary driving force behind aggressive technology scaling. Two important consequences of technology scaling are, the increase in leakage power, and increase in variation in the characteristics of manufactured devices. Leakage power is projected to contribute more than 40% of total power budget in processors fabricated in 65 nm technology and beyond [18]. Unlike dynamic power, leakage power is highly sensitive to variations in gate dimensions as well as the operational temperature.

High variation in the power consumption results in significant overestimation of the specification, leading to increased design time/effort and results in significant loss of parameterized yield [4, 3]. *Hence, reducing both the total power, and the variation in the power consumption of FUs is an important problem.*

Leakage power is a very important concern for functional units (FUs) such as Integer ALUs, Floating point ALUs and Multipliers which are significant contributors to the total energy consumption of the processor [7]. In addition, FUs being regions of high activity, are among the hottest regions on the chip. Therefore reducing both the leakage power, and the variation in leakage power of FUs is an important research problem.

Among the existing techniques to reduce the leakage power of FUs, power gating is one of the most promising approaches. [5]. Power gating is a technique which reduces leakage by shutting off the power supply to a unit during periods of inactivity. However existing power gating mechanisms [10, 17] do not consider dependence of leakage on temperature and process variations.

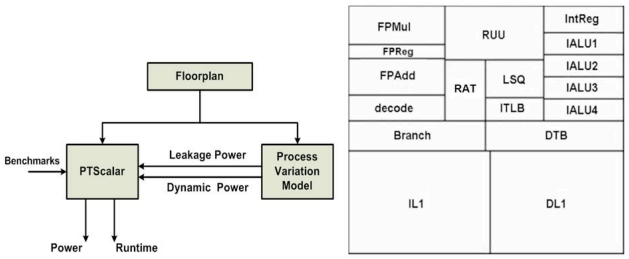
In this paper, we propose to introduce a leakage sensor in FUs, and develop a temperature and process variations aware power gating technique. We present a power gating technique based on the IPC and propose Leakage Aware Power Gating (LA-PG) scheme, which is both temperature and process variations aware, to decide on which FUs are to be power gated. Our technique, LA-PG results in 22% reduction in the average, and 25% reduction in the standard deviation of the total ALU energy consumption, without any performance loss, as compared to existing power gating techniques.

2. Experimental Setup

Microarchitecture Model: Our simulation framework is depicted in Figure 1A. We perform our experiments on

1063-9667/08 $25.00 © 2008 IEEE
DOI 10.1109/VLSI.2008.83

the ALPHA DEC 21364 processor. This is a 4-wide superscalar processor, whose floorplan is shown in Figure 1B.



A. Simulation Framework

B. Floorplan of ALPHA DEC 21364 Processor

Figure 1. Simulation Setup

The power, performance and temperature modeling of alpha processor is done using a modified version of sim-outorder of the PTScalar toolset. PTScalar is a coupled power and thermal simulator built over SimpleScalar [6]. We execute several benchmarks from the MiBench [9], and Spec 2000 [1] suite.

Process Variation model: We model the variations in device features (gate length and threshold voltage) using the stochastic process corresponding to gate length using the Karhunen-Loève Expansion proposed in [3]. The process variation model generates the dynamic power and leakage power values for the ALUs in the processor corresponding to one die. We generate 1000 such die samples, which are fed into PTScalar for power, performance, and temperature modeling. The power numbers are scaled to correspond to 45nm technology.

3. Motivation

In this section, we perform experiments on the representative susan-corners benchmark from the MiBench suite [9], to demonstrate the need to reduce the FU energy consumption, as well as the variation in the FU energy consumption in the presence of temperature and process variations.

3.1. FUs are regions of high energy density

Figure 2 shows the total energy consumption (dynamic + leakage) of all the units sorted by their energy consumption for all the 1000 die samples. It can be observed from the plot that the total energy consumed by the Integer ALUs is 11.2% of the total processor energy.

In addition, FUs are one of the most active units in a processor, and therefore have very high energy density. This is exacerbated by the exponential relation of leakage on temperature. Figure 3 shows that ALUs have second highest energy density among all the units, only next to the IntRegFile.

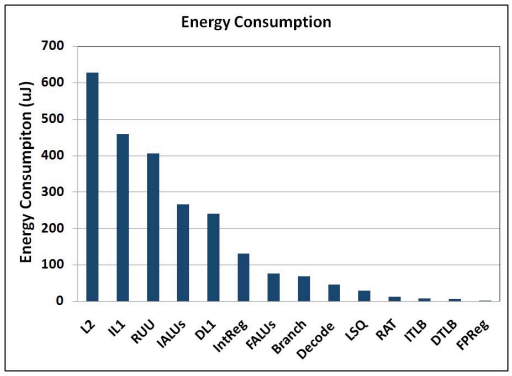


Figure 2. Energy consumption of all units in the alpha processor

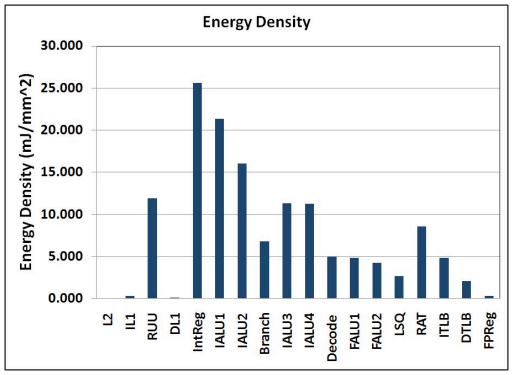


Figure 3. Energy density of all units in the alpha processor

3.2. FUs contribute significantly to variation in processor energy

Figure 4 plots the standard deviation of the energy consumption of each unit, across the 1000 die samples. The plot shows that ALUs have the highest variation in energy consumption. This is also due to the strong exponential dependence of leakage on temperature.

4. Related Work

Butts and Sohi [5] demonstrated that due to the exponential dependence of leakage on temperature, combinational logic has an order of magnitude larger leakage current relative to cache RAM transistors. Since Functional Units (FUs) are regions of high power density in the processor, techniques to reduce the leakage power of FUs were explored. Of various FU leakage reduction techniques, power gating [8, 10, 17] has proven to be the most effective for FU leakage reduction. These techniques address the question of how to implement power gating.

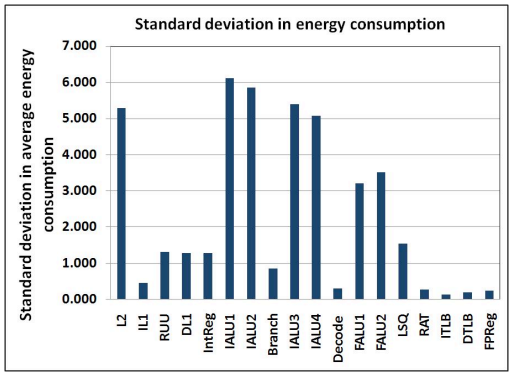


Figure 4. Standard deviation of the energy consumption of all units in the alpha processor

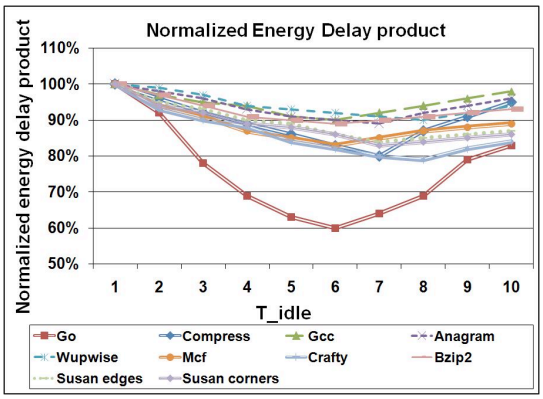


Figure 5. Energy delay product variation with t_{idle}

When to do power gating, has been approached from two directions, i) compiler-based solutions, and ii) hardware solutions. Compiler-based FU leakage reduction techniques were studied in [17]. But this technique requires that the entire code be examined off-line to identify suitable regions for turning the functional units off. Hardware based techniques for identifying the idle regions consume additional power throughout the execution.

Previous techniques for FU power gating in superscalar processors are idle-time based [8]. Whenever an FU is predicted to be idle for more than break-even time, the FU is power gated.

Previous works have attempted to design effective and accurate leakage current sensors [15, 12]. Our power gating scheme, reads the reading of the leakage sensor and power gates functional units in order to reduce both the power consumption, as well as the variation in the power consumption of the FUs. To the best of our knowledge, this is the first work in this direction.

5. Previous Approach: Idle Time-based Power Gating (IT-PG)

In the idle-time based power gating technique (we call it IT-PG), t_{idle} is the key parameter. The activity of FU is monitored, and if the FU is idle for more than t_{idle} cycles, the power supply to the FU is gated off. Once in a power gated state, the FU will be woken up (power gating is disabled) when an operation is issued to it. The parameter t_{idle} can be varied to obtain a tradeoff between performance and leakage savings.

Figure 5 plots the normalized energy delay product of all our benchmarks for varying values of t_{idle}. The average of energy delay product over all our benchmarks is the least for t_{idle} = 7. This is consistent with previously published results [10], who found the optimal value of t_{idle} as between 6-9 cycles, and therefore we choose $t_{idle} = 7$ for our comparison experiments.

6. Our Approach: Leakage Aware Power Gating (LA-PG)

Our first observation is that temperature increases are gradual, and like a plethora of previous works, we assume that appreciable temperature changes occur only at $10,000$ cycle granularity, and therefore it is reasonable to implement temperature dependent policies at this granularity [11, 16]. Our power gating mechanism is a two step process: we use the current IPC information to find out how many FUs to power gate, and then we use leakage sensor values to determine which FUs to power gate.

6.1. How many FUs to Power Gate?

At each decision moment (i.e., every 10,000 cycles), we compute the *average IPC*, or the average number of instructions that are *ready to be issued* every cycle. Note that this is different from the regular definition of IPC or Instructions Per Cycle, which is the number of instructions issued each cycle. However, due to it's close similarity to IPC, and since we do not use IPC otherwise in this paper, we call our approach as IPC based technique. The number of FUs to power gate is determined by comparing our computed average with a *threshold*. For a n FU configuration, we have $n - 1$ thresholds.

The *average IPC* is computed as an average of IPCs of the last *history* number of cycles. The value of *history* determines the accuracy of our power gating technique. Therefore, the history and the thresholds are the two key parameters of our IPC threshold-based power gating technique. Designers can vary these parameters to trade off power, performance, and architectural complexity. To determine

suitable values of history and thresholds, we simulated all the 10 benchmarks with IPC threshold-based power gating technique for several values of history and thresholds. We vary the history from 10 to 1000 cycles, and the threshold value for the case when a single ALU is in the active mode from 1.0 to 1.20 in steps of 0.01. The corresponding values for two and three ALUs to be in the active mode are varied from 2.0 to 2.20 and 3.0 to 3.20.

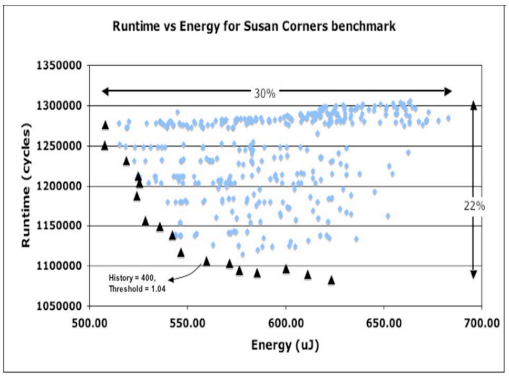


Figure 6. Runtime vs Energy showing the pareto optimal points for Susan Corners benchmark

Figure 6 shows the runtime vs energy plot for all history-threshold configurations for the representative susan-corners benchmark. The figure shows that a variation of 30% in the energy of the ALUs and a variation of 22% in the runtime is possible by power gating. We have identified and marked the pareto-optimal points by the dark triangles in the figure. A configuration is pareto optimal if it is not worse than any other configuration in both power and performance. Designers can choose any of these pareto-optimal design points to trade-off power and performance.

We varied the values of history and thresholds for all 10 benchmarks, and computed the energy-delay product for each history-threshold configuration. We then compute the summation of the energy-delay product for all benchmarks for each history-threshold configuration. The optimal values of history and thresholds came out to be 400, 1.04, 2.04, and 3.04 respectively, and we use these values in estimating the effectiveness of our approach.

6.2. Which FUs to Power Gate?

In order to reduce the leakage, we want to power gate the FUs which have the highest leakage. An FU may have high leakage either because of process variations, or because it's temperature is high. Thus LA-PG is both temperature and process variation aware. Power gating the FU with the highest leakage, minimizes the FU power consumption; in addition it also reduces the variation in the leakages of FUs.

Introducing Leakage Sensors: We propose to introduce the leakage sensor proposed by Kim et al. [15] inside each FU and continuously measure the FU leakage during the chip operation. A single channel leakage sensor is shown in Figure 7A. M2 is the only transistor that is sensitive to the variation in leakage of the ALU due to the impact of temperature and process variations. Therefore, the accuracy of leakage sensor itself is not affected by process and temperature variations.

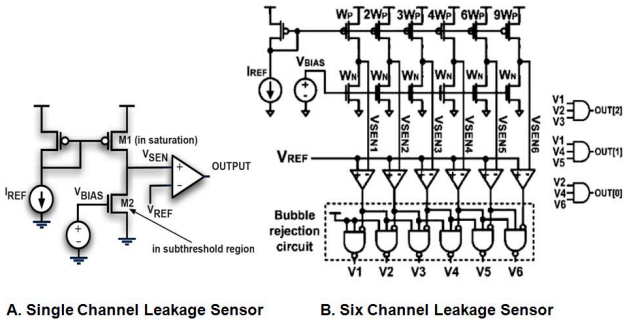


Figure 7. Six Channel Leakage sensors [15]

We explicitly model the area, power and inaccuracy introduced when converting the leakage sensor in our experimental setup. The overhead of using leakage sensors accounts to around $3-4\%$ reduction in the total power savings obtained using our LA-PG.

Leakage Sensor Placement: To find a good location for the leakage sensor, we compared the leakage of a device located at various locations (x_i, y_i) inside the ALU, and the average leakage of the ALU ($I_{av} = I_{S,T}/N$). We found that mean of the percentage difference between the average ALU leakage and the leakage of a device located at the center of the ALU for a sample of 1000 dies to be less than 1%. The maximum percentage error over the same set of samples was 7%. Thus a single leakage sensor located at the center of the ALU can provide accurate estimation of the leakage power of the entire ALU.

Microarchitectural Overheads: Figure 8 shows the implementation of the circuit required for our technique. A naive implementation could have high power and performance overheads. Therefore, we introduce several optimizations in the implementation. (i) We limit the range of IPC to be only from 0 to n_{issue}, instead of 0 to $n_{reorder}$, for a $n - issue$ superscalar processor with a re-order buffer size of $n_{reorder}$. In other words, if the number of instructions that are ready is more than n_{issue}, the IPC saturates at n_{issue}. This reduces the size of the microarchitectural overhead tremendously. (ii) Instead of adding 400 values, we add the IPC every 4^{th} cycle for a period of 512 cycles. This results in 128 samples of IPC over 512 cycles. On a 4-issue superscalar, the maximum value of the sum of the IPC over the entire sampling period will not exceed 512. Hence, a

9 bit adder is sufficient for this purpose and it can be implemented as very low-power $ripple\ carry\ adder$, and still meet the timing constraint. This further reduces the power consumption of the architectural overhead. The logic circuit required is a small combinational logic block that determines how many ALUs to power gate based on the IPC sum and the threshold values and which ALUs to power gate based on the 3 threshold values which are scaled to 133, 233 and 333, and the 3 bit output of the leakage sensor placed in each of the 4 ALUs.

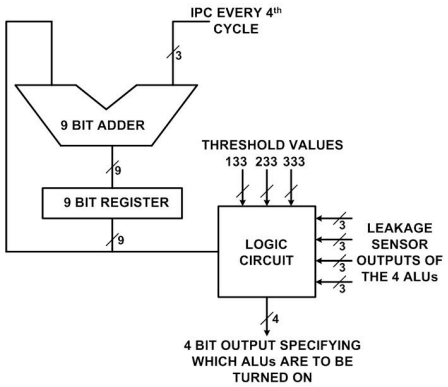


Figure 8. Microarchitectural enhancements for the IPC Threshold based power gating technique

We synthesized this logic using Synopsys Design Compiler and implemented it in Cadence Spectre toolset (Virtuoso Schematic editor) using TSMC 0.25um CMOS deep submicron process, and scaled the numbers to 45 nm. We also synthesized the logic for the IT-PG technique for comparison purposes. This logic has an area overhead of less than 3% and energy overhead of $< 0.15\%$, as compared to the architectural overhead of idle-time based technique. We include the energy overhead due to our logic in the power computations using PTscalar in all our simulations.

7. Experiments

7.1. LA-PG reduces ALU energy consumption

Figure 9 plots the mean of the ALU energy consumption computed over 1000 sample dies, for LA-PG, normalized to IT-PG, for all the 10 benchmarks. The 11^{th} bar to the extreme right denote the average energy reduction achieved over all the benchmarks.

The figure shows that LA-PG decreases the average energy consumption by 22% as compared to the IT-PG. The performance penalty of applying our IPC threshold-based power gating techniques is less than 2%. This performance loss is lesser than the performance loss of IT-PG, which is 2.2%, as compared to the case with no power gating. Another important observation from the graph is that the energy reductions are quite uniform across the benchmarks. Hence, the effectiveness of our technique is consistent through the benchmark spectrum.

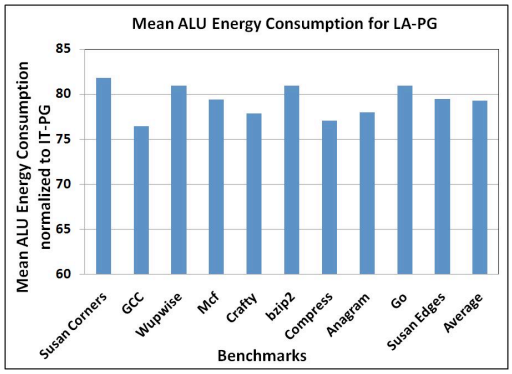


Figure 9. Mean ALU energy consumption by IPC threshold based techniques

7.2. LA-PG mitigates process variation

In the presence of process variation, the power gating priorities assigned for one die may not be the best for the other dies. For the same priorities, the variation in the total ALU energy consumption in different dies may be quite significant. We simulate all the 10 benchmarks with IT-PG and LA-PG techniques, for 1000 die samples.

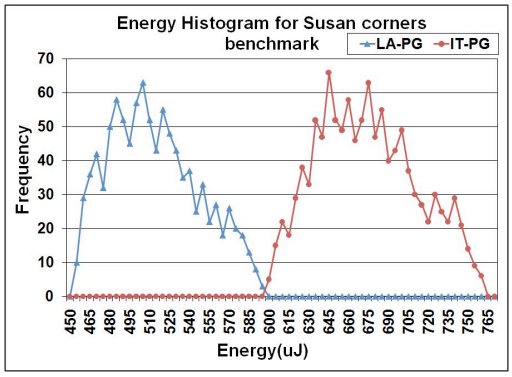


Figure 10. Energy histogram for various power gating techniques for susan corners benchmark

Figure 10 plots the energy histogram for IT-PG and LA-PG techniques, for susan corners benchmark for 1000 die samples. The rightmost curve (lines connected by circles)

corresponds to the energy distribution for IT-PG. The mean and standard deviation of the energy distribution are 675.18 μJ and 33.76 μJ. For processors that will incorporate leakage energy sensors, LA-PG technique is very effective. It reduces the mean and standard deviation to 521.98 μJ and 23.2 μJ respectively, as shown by its energy histogram depicted by the leftmost curve (lines joined by triangles). As compared to IT-PG, LA-PG reduces the energy consumption by 22% and reduces the standard deviation by 25%.

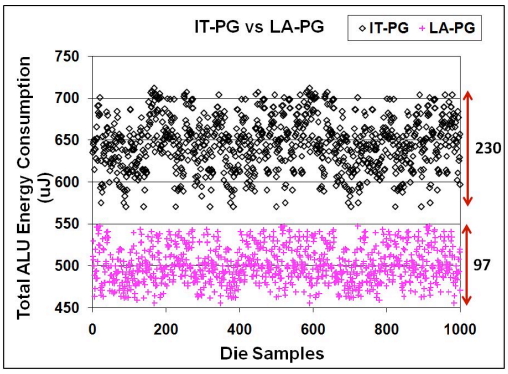


Figure 11. IT-PG vs LA-PG

Figure 11 plots another view of comparison between the ALU power consumption for IT-PG and LA-PG in 1000 die samples for susan corners benchmark. The same two observations can be made from this graph: (i) all the LA-PG points are lower than the IT-PG points, and (ii) the width of the vertical band in which points of LA-PG lie is lesser than the width of the band in which the points of IT-PG lie. The difference between the lowest and highest energy dies is around 230 μJ in the case of IT-PG when compared to 97 μJ in the case of LA-PG.

8. Summary

The exponential dependence of leakage on the temperature and device dimensions has made leakage an increasingly important concern in the nano-design era. In this paper, we presented an IPC threshold based power gating technique that reduces the energy consumption and the variation in the total ALU energy consumption across dies. Our LA-PG is both temperature and process variation aware, through a leakage sensor in the FUs. LA-PG reduces the mean ALU energy consumption by 22% and reduces the standard deviation in the ALU energy consumption by 25%, without any performance penalty, as compared to existing techniques.

Acknowledgement

We would like to thank Microsoft Corporation for their generous support.

References

[1] SPEC2000 Benchmarks, www.spec.org/benchmarks/html, 2000.

[2] A. Abdollahi, F. Fallah, and M. Pedram. Leakage current reduction in cmos vlsi circuits by input vector control, 2004.

[3] S. Bhardwaj, S. Vrudhula, P. Ghanta, and Y. Cao. Modeling of intra-die process variations for accurate analysis and optimization of nano-scale circuits. In *Proc. of IEEE/ACM Design Automation Conference*, 2006.

[4] D. Boning and S. Nassif. Models of process variations in device and interconnect, 2000.

[5] J. A. Butts and G. S. Sohi. A static power model for architects. In *Micro33*, pages 191–201, 2000.

[6] D.Burger and T.Austin. The simplescalar tool set version 3.0, 1997.

[7] J. Deeney. Reducing power in high-performance microprocessors. In *International Symposium on Microelectronics*, 2002.

[8] D. Duarte, Y.-F. Tsai, N. Vijaykrishnan, and M. J. Irwin. Evaluating run-time techniques for leakage power reduction. In *VLSI Design*, pages 31–38, 2002.

[9] M. R. Guthaus, J. S. Ringenberg, D. Ernst, T. M. Austin, T. Mudge, and R. B. Brown. MiBench: A free, commercially representative embedded benchmark suite. In *IEEE Workshop in workload characterization*, 2001.

[10] Z. Hu, A. Buyuktosunoglu, V. Srinivasan, V. Zyuban, H. Jacobson, and P. Bose. Microarchitectural techniques for power gating of execution units. In *Proc. of ISLPED*, pages 32–37, New York, NY, USA, 2004. ACM Press.

[11] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan. Hotspot: A compact thermal modeling methodology for early stage vlsi design. *IEEE Transactions on Component Packaging and Manufacturing Technology*, 14(5), 2006.

[12] J. Hurst and A. Singh. A differential built-in current sensor design for high speed iddq testing. *vlsid*, 00:419, 1995.

[13] J. Kao and A. Chandrakasan. Dual-threshold voltage techniques for low-power digital circuits, 2000.

[14] S. Kaxiras, Z. Hu, and M. Martonosi. Cache decay: Exploiting generational behavior to reduce cache leakage power. In *Proc. of ISLPED*, pages 240–251, 2001.

[15] C. H. Kim, K. Roy, S. Hsu, R. Krishnamurthy, and S. Borkar. A Process Variation Compensating Technique with an On-Die Leakage Current Sensor for nanometer Scale Dynamic Circuits. *IEEE Transactions on VLSI*, 14(6):646–649, 2006.

[16] W. Liao, L. He, and K. Lepak. Ptscalar version 1.0, 2004.

[17] S. Rele, S. Pande, S. Onder, and R. Gupta. Optimizing static power dissipation by functional units in superscalar processors. In *Computational Complexity*, pages 261–275, 2002.

[18] J. W. Tschanz, S. G. Narendra, Y. Ye, B. A. Bloechel, S. Borkar, and V. De. Dynamic sleep transistor and body bias for active leakage power control of microprocessors. *IEEE Journal of Solid State Circuits*, 38, Nov 2003.