Emerging Power Management Tools for Processor Design

D. T. Blaauw, A. Dharchoudhury, R. Panda, S. Sirichotiyakul, C. Oh, and T. Edwards
Motorla, Inc.
Austin, TX
blaauw@udsl.sps.mot.com

1. ABSTRACT

Power management is an increasing concern for processor design. In this paper, we presented an overview of traditional power simulation tools and discussed two emerging power management design technologies: power distribution integrity analysis and standby current measurement and optimization. We present methods for accurate peak current simulation, which is needed for power grid integrity analysis, and discuss the generation and compression of the simulation vectors. Also, static approaches for calculating an upper-bound on the maximum peak current are presented. Standby leakage current is state dependent and we present methods for calculating both the average and maximum leakage current. Finally, optimization methods for minimizing the leakage current by either assigning a standby state to the circuit or by using a dual-Vt process are discussed.

1.1 Keywords

low power CAD, power distribution, standby leakage

2. INTRODUCTION

With the growing demand for portable applications, low power processor design is increasingly common. In addition to power requirements, processors also have very stringent performance requirements. These conflicting goals present the designer with a challenging problem. In order to effectively reach an optimum trade-off between performance and power, a number of mature design technologies are needed. The most prominent and mature low power design tool is a power simulator. Power simulation can be performed at the transistor level, gate level, or RTL level. Each additional level of abstraction increases the performance of the tool but reduces the accuracy of the power estimate.

The drive for lower power, as well as process shrink, have lead to aggressive reduction of the supply voltage. With the reduction of the supply voltage, standby leakage current and power grid integrity have become prominent issues in processor design. Accurate power grid analysis places a number of new requirements on traditional power simulation tools. Traditional power simulation calculate the average power or maximum power over one or more clock cycles. Power simulation for power grid analysis is based on instantaneous power analysis, which requires a much higher accuracy simulation. Also, obtaining simulation vectors that exercise the worst case instantaneous power event in a design is a new and challenging problem.

For designers to meet the increasingly stringent leakage current requirements for today's portable processors, mature leakage measurement tools and leakage optimization technologies are needed. Since standby leakage current is a function of the circuit state in standby mode, methods for calculating the average and maximum leakage of a circuit are proposed.

The remainder of this paper is organized as follows: In Section 2, we present an overview of traditional power simulation tools. In Section 3, we discuss power grid analysis and the requirements it places on power estimation. In Section 4, we discuss standby leakage measurement and optimization approaches. Finally, in section 5, we present conclusions and plans for future work.

3. Power Simulation for Processor Design

Power simulation can be performed at the transistor, gate, or RTL level. When the level of abstraction of the simulation is increased, the run time performance of the simulator also increases, while the accuracy of the power estimate is reduced. Running Spice on a transistor-level description of the circuit gives the highest level of accuracy. However, the simulation time required for Spice scales super-linear with the size of the circuit. Therefore, Spice simulation can only be applied to small circuits. Nevertheless, Spice simulation is still one of the most extensively used tools for power and performance analysis in the design of low power processors.

In order to address the capacity issues of Spice, a number of Spice-like transistor-level simulation tools have been introduced. These tools rely on simplified device models and circuit partitioning techniques[1] to obtain run time performance which is linear with the size of the circuit. For moderately sized designs, these simulators perform between one to two orders of magnitude faster than traditional Spice simulation[2]. The accuracy for average power estimation is between 5%-10% with respect to Spice. The advantage of fast transistor-level simulators is that they allow simulation of large circuit blocks that cannot be reasonably simulated with Spice. It is, therefore, possible to perform block power characterization and power budgeting for modules on the chip. Fast transistor level power simulators have become an integral and indispensable part of the design methodology for processor designs.

For gate-level power simulation, a gate-level simulator is used to determine the frequency of different input and output switchings for each gate. A precharacterized power model

ISLPED98, Monterey, CA, USA
© 1998 ACM 1-58113-059-7/98/0008..$5.00

for each standard cell gate is then used to calculate the power dissipated by each type of signal transition. The challenge of gate level power estimation lies in the construction of accurate power models. Typical gate level power simulators obtain an accuracy of 10%-15% with respect to fast transistor level simulators, while their performance is improvement by approximately 1 to 2 orders of magnitude[3]. Probabilistic approaches for calculating the switching activity of nodes have been developed which dramatically increase the performance of the power estimation, but also reduce its accuracy. One limitation of gate level power estimation techniques is that the entire design must consists of cells with precharacterized power and simulation models. Therefore, gate level power simulation fits well within ASIC designs flows. In typical processor designs, a significant portion of the design uses custom design techniques for which it is difficult and time consuming to generate reliable power models.

At the RTL level, significant trade-offs between the performance, area, and power of a design can be made. Therefore, there is a greater opportunity to reduce the power of a design at the RTL level than at any other level of design. However, fast and accurate power estimation at the RTL level is difficult. Two main approaches have been successfully used. The first approach uses a library of pre-characterized power models for different RTL level components, such as muxes, adders, registers, etc[4]. Control logic is quickly synthesized with low effort and simulated with a gate level simulator. Typically, this approach results in a 20% to 40% error in the estimated power relative to that of a fast transistor level simulation. The second approach predicts power based on the complexity of the Boolean function measured in terms of an entropy function. The entropy function is related to the signal probabilities[5]. This approach has very fast run times but has an inferior accuracy compared to the first approach. Even though the accuracy of RTL level power simulation is not sufficient for predicting the power of a design, RTL estimation can be used to evaluate the relative power of two RTL implementations. This can give the designers valuable information regarding which RTL implementation is preferable for a design, and result in an ultimately much more low power design.

4. Power Grid Integrity Analysis for Processor Design

In recent years, the integrity of the power grid supplying Vdd and Gnd to the devices on the chip has become a significant concern for chip designers. Due to the resistance of the interconnect, a small voltage drop develops as the power grid supplies current to the circuitry on the chip[6]. Since the current drawn by the devices fluctuates with time, the voltage delivered to the devices fluctuates. The voltage drop and voltage fluctuation results in a number of problems. First, the reduced voltage has a direct impact on the performance of the design. This can lead to degraded or unreliable performance. Secondly, the voltage fluctuations in the power grid inject noise into the signal lines of the circuit, causing possible functional failure. Thirdly, a poorly-designed power grid will result in very high current densities in the metal lines, causing electro-migration and reliability concerns. In addition to the resistance of the metal grid of the power supply network, the inductance of the package also needs to be considered and, in the near future, the inductance of the power grid itself needs to be considered as well.

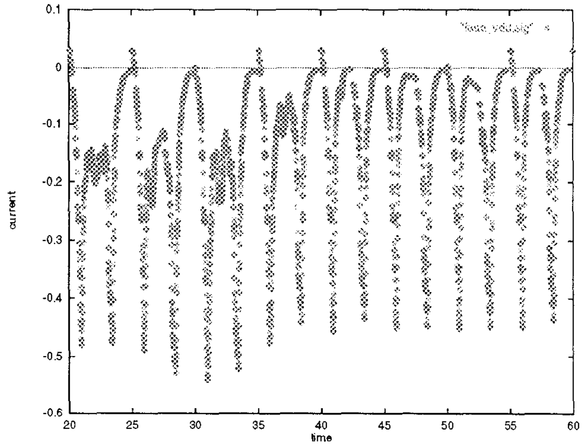


Figure 1. Current profile of floating point unit from the PPC 750.

The voltage at which processors operate is steadily decreasing with each process shrink. As a result, the current needed to supply the chip with power is increasing. Also, the average current density is increasing since the line widths are decreasing with process shrink. Inductance is becoming a more prominent effect with increased frequency. Therefore, power grid integrity is becoming a more significant problem with process evolution, and quality CAD solutions for analyzing and improving power grid integrity are needed.

The difficulty of analyzing the power grid stems from the size of the problem. Fundamentally, the power grid analysis is a global problem. The common approach to power grid analysis is to separate the non-linear devices from the linear interconnect of the grid itself. First, the non-linear devices are simulated with a fast power simulator, and the current of each gate is recorded. Then, these currents are represented by a time-varying current source in a simulation of the power grid using a linear solver[6]. The fast power simulation is performed with the assumption that the circuit has a clean Vdd and Gnd power supply. This leads to an over-estimation of the current drawn by the devices relative to when the voltage is supplied by the power grid. However, as long as the voltage drop in the power grid is small, the conservatism is manageable.

4.1 Power estimation for power grid analysis

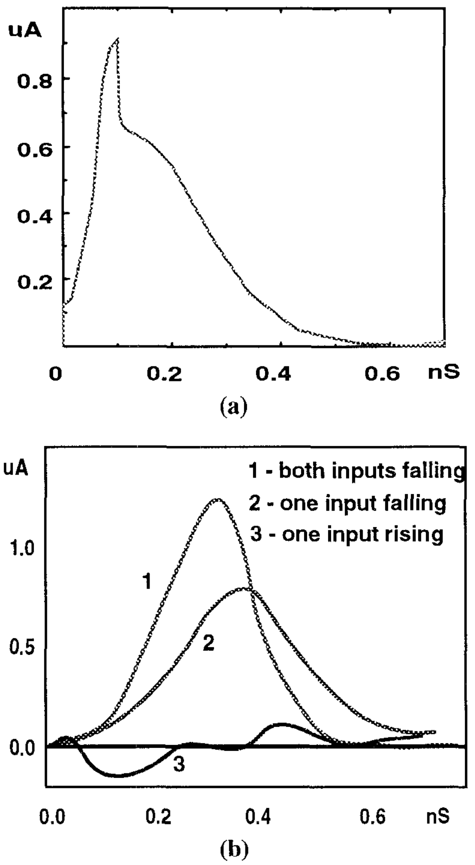
The difficulty in performing reliable power grid analysis lies in obtaining accurate worst case estimates of the peak current. In Figure 1, the current drawn by a floating point unit from the PPC 750 processor is shown. The clock cycle time is 5ns and the two phase clock creates a current peak every 2.5ns. The peak current is approximately 5 time as large as the average current and determines the worst case voltage drop in the power grid. There are two approaches for estimating the peak current: dynamic and static. The dynamic method uses simulation vectors and traditional power simulators to obtain current profiles. The static method calculates a conservative upper bound on the current profile and does not require user supplied simulation vectors.

4.2 Dynamic simulation approaches for power grid analysis

Power simulation for power grid integrity analysis differs significantly from that of traditional 'power analysis. Traditional power simulation calculates either the average or maximum power of a circuit over one or more clock cycles. Therefore, the exact shape and timing of the current pulses is not critical. For power grid analysis, the voltage drop across the power grid corresponds to the worst case instantaneous power event in the chip. This makes the shape and exact timing of each current pulse in the circuit of critical importance, and the simulation must have a much higher timing resolution than what is needed for traditional power estimation. For instance, when running a commercial fast power simulator in the standard accuracy mode on a small circuit block for the PPC 750, we obtained a 10% accuracy for the average power (or current) with respect to Spice simulation. However, the magnitude of the peak current measured in this same simulation had a deviation of 50% compared to that of Spice. Also, the timing of the peak had significant inaccuracy. In order to obtain an accurate peak current estimation, we had to run the power simulator in a high accuracy mode which required more than 10 times the simulation time required for the standard simulation mode.

The accuracy requirements for peak current estimation stems from the short time duration of the current pulse induced by a switching gate. Figure 2a shows the current drawn by a typical inverter in the PPC 750. The current pulse width at 50% of peak magnitude is 250ps, and the current pulse width at 70% of peak magnitude is only 50ps. To assess which current peaks overlap and to calculate the peak current of a block, the simulation must have a timing accuracy that is significantly better than 30ps. On the other hand, for average power calculation the simulation only needs to determine whether the gate switched and accurately determine the total charge that was dissipated. For power grid analysis the time varying current of each component must be explicitly stored and individually modeled as current sources. Therefore, the storage requirement for instantaneous peak power simulation is also much greater, and the user must ensure that all simulations were obtained with correlated input vectors.

Power simulation for power distribution network integrity analysis is typically performed at the device-level. However, gate-level simulators can significantly decrease the run time of the simulation. Each gate must be precharacterized with a switching current model. Figure 2a shows that the current profile of the gate is characterized by three constants: the slope of the leading and trailing ramp and the peak value of the pulse. The current profile of a gate is a function of the output load capacitance, input slope, and which combination of inputs is switching. Figure 2b shows that the peak current of a nand gate is significantly higher when both inputs switch simultaneously low than when only one input switches low. Therefore the current model for the gate must account for simultaneous switching, which significantly complicates the model. Another important phenomena that is significant for power grid analysis, but overlooked in gate power characterization is the feed-through current evident when one or both inputs rise in the Nand gate (curve number 3).



(a)

(b)

Figure 2. Vdd current for a 2 input inverter and nand gate.

4.3 Synthesis and compression of simulation vectors

An important issue in power simulation for power grid integrity analysis is how to obtain the chip level vectors which exercise the worst possible IR-drop event in the power grid. Current practises typically rely on the user to construct so-called 'hot loops', designed to maximize the average current dissipation by executing high activity instructions. These hot loops are used in the power simulation for the power grid integrity analysis in the expectation that they contain the worst case IR-drop event. However, the worst average power vector does not necessarily correspond to the worst instantaneous power vector. A number of methods for automatically generating power grid simulation vectors have been proposed[7, 8]. These methods use optimization algorithms such as genetic algorithms to automatically generate a vector that maximizes the instantaneous current of a block. One weakness of these algorithms is that they are typically applied to individual circuit blocks as opposed to the entire chip design. Because of this, the generated vectors for the blocks will not be correlated with one another, possibly under estimating the peak current prediction from the chip perspective.

High quality simulation vectors can be quite long, ranging between hundreds to thousands of clock cycles. Due to the size of the power grid, the linear solver requires significant run times (between 2 to 30 minutes) per time point for reasonably large processor designs (5 to 20 million devices). Simulating even 20 clock cycles at a resolution of 50 time points per clock cycle requires between 1 day to 2 weeks of simulation. Therefore, vector compression techniques are needed to manage the run time requirements for the power grid integrity analysis. The most common compression technique is to generate a single cycle current envelope for each simulated component of the circuit. While this will typically bring the analysis run time to an acceptable level, the compression inherently destroys the timing correlations between the circuit components, and introduces significant conservatism.

4.4 Static approaches for power grid analysis

An alternative to simulation based approaches is to take a static approach for power estimation. which provides an upper bound on the peak current value. The simplest static approach is to calculate the worst peak current for each circuit component (such as a gate) using Spice simulation, and sum over all components. No timing correlations are taken into account in this approach, and the result can be very conservative. For a G3 series processor, this approach resulted in a total chip peak current of 160A, compared to an average current under normal operation of 3A. The induced voltage drop of 630mV compares to a 34mV voltage drop under simulation of a designer supplied hot loop.

An improvement on this basic approach was proposed in [9]. The basic idea is to calculate activity windows for each gate during which the gate can switch either high or low. The gate is assumed to dissipate its peak current for the entire duration of the activity window. The number of activity windows provides a trade-off between the time complexity of the approach and the conservatism of the resulting peak current estimation. The logical correlations between timing windows can also be included. The approach promises to provide much better results than the basic static approach, though efficacy still needs to be demonstrated on industrial designs.

5. Leakage Calculation and Optimization Tools

With the increased use of battery operated devices, standby current is become an important constraint for chip designs. Most portable devices spend the majority of their time in standby mode, during which the system clock is inactive. In this mode, current drawn by the device is due to the static leakage current of the gates in the circuit. Although the magnitude of this current is several orders of magnitude lower than the active current during normal operation, the standby current typically dominates battery life due to the large portion of time that the device spends in standby mode.

With the reduction of the supply voltage, the threshold voltage (Vt) of the devices is reduced to maintain proper scaling for performance. With the reduction of Vt, the sub-threshold leakage current of the device increases exponentially, causing the leakage current of devices to be a major concern in new portable chip designs. The drain induced barrier lowering (DIBL) effect in short channel devices also increases the leakage of the device. Leakage therefore increases dramatically with process shrink while the leakage requirement for portable devices is becoming more stringent. This is causing an increasing focus on design technology for the analysis and optimization of leakage current in circuits.

5.1 Sources of leakage and leakage issues

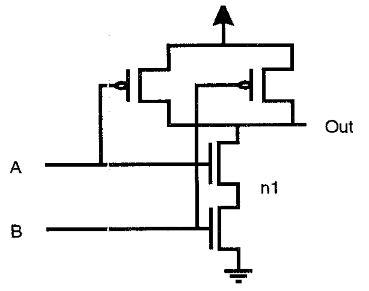
There are three main sources of leakage current in a CMOS design: sub-threshold channel leakage, junction leakage, and well leakage[10]:

- Sub-threshold leakage is the leakage current across the channel of the device when the device is intended to be off. The subthreshold current increases exponentially with a decrease in Vt. The subthreshold current is a function of terminal voltages and the transistor length and width.
- The junction leakage is the leakage current of the reverse bias diode at the drain and source junction.
- The well leakage is the leakage current of the reverse bias junction of the well/substrate interface. Since this leakage is not a function of any device parameters but only of process parameters, we will omit well leakage from the remainder of our discussion.

Figure 3 shows the leakage of a two input nand gate for high and low Vt devices. The results in Figure 3 indicate that junction leakage is more than one order of magnitude smaller than the subthreshold leakage and can be ignored.

The leakage is strongly dependent on the state of the circuit. The highest leakage for this typical nand gate is as much as 7 times larger than the lowest leakage. The difference in leakage is accounted for as follows:

- In the maximum leakage state, both the Pmos transistors are turned off. In this state, the output of the gate is nom-



Circuit State		Leakage Current(nA)			
		High Vt		Low Vt	
A	B	Isub	Ijunc	Isub	Ijunc
1	1	0.4608	0.0028	13.592	0.0028
0	1	0.3175	0.0016	11.453	0.0016
1	0	0.2517	0.0040	8.7345	0.0041
0	0	0.0743	0.0017	2.0321	0.0018

Figure 3. Leakage current for a 2 input Nand gate.

inal 0 volts, and both Pmos transistors see the full Vdd drop across their channel. Also, the leakage current of the parallel devices is additive.

- In the second highest leakage state, the top Nmos transistor is off and the bottom Nmos device is on. In this state, the source node of the Nmos device is again nominal ground, while the drain is nominal Vdd. The leakage is equal to the leakage of the Nmos device with Vds = Vdd.
- In the third highest leakage state, the bottom Nmos transistor is off. A voltage drop across the top Nmos transistor reduces the Vds of bottom Nmos device and decreases the leakage of this state by 20% relative to the previous state. The first order approach to calculating the leakage in this state is to assume that the voltage at node n1 is Vdd-Vt, where Vt is the threshold voltage of the top Nmos device. However, we have not found this simple analysis invalid for current technologies, since the top transistor does not completely shut-off when Vgs < Vt. The leakage of the top transistor is in equilibrium with the leakage of the bottom transistor, which requires analysis with a circuit simulator.
- In the last and smallest leakage state, both Nmos devices are off. In this state the supply voltage is divided between the two leaking Nmos transistors. Again the total leakage is a function of the relative sizes and Vt's of the two Nmos transistors, and requires a non-linear simulation of the circuit.

Typically, the state will be partially know in standby mode. Control signals may be known, while data signals typically are unknown and depend on how the device entered sleep mode.

5.2 Leakage estimation

The primary tool that a designer needs to meet his standby current constraints is an accurate leakage estimation tool for large circuit blocks. Two types of leakage measurements can be obtained: the maximum leakage and the average leakage. The number of possible input state permutations is 2**N, where N is the number of input signals that are unknown in the sleep state. In general, the number of possible input state permutations will be very high and an exhaustive search of the input states is not feasible. In light of this, two possible approaches can be taken. A) A general search algorithm can be applied, such as a branch-and-bound or genetic algorithm[11], which will generate a low bound on the maximum current. B) The state correlations between the gates in the circuit are ignored, and the maximum leakage is calculated as the sum of the maximum leakage of each gate. This approach will yield an upper-bound on the maximum leakage.

Although the maximum leakage can help the designer find leakage hot-spots, the average leakage of a circuit provides a more meaningful measure for the expected battery life. A device will enter and exit standby mode many times over the life of one battery, each time having a random state for the free input signals. The total standby power that is drawn by a device over a long period of time is therefore directly related to the average leakage current of the circuit, not the maximum leakage current.

The average leakage current is calculated by taking the weighted sum of the leakage of each state state of a gate. Again, one approach is to ignore the state correlations of the gates, providing an approximate estimate of the average leakage. Alternatively, the state probabilities can be calculated using signal probabilities and corellations[12]. The user can provide input probabilities for the free input signals to improve the leakage estimation.

5.3 Design Technology for Leakage Reduction

The leakage current can be minimized in a circuit by controlling the state in standby mode[13]. The minimum leakage state of the circuit is calculated using techniques similar to those for maximum power calculation and is then assigned to the circuit in standby mode by adding logic to the latches in the design. The difficulty with this approach is that imposing a fixed sleep state requires insertion of extra logic in the circuit, which will add to the area, power and delay of the design. Also, the scope for reducing the leakage is limited with this approach.

An alternate approach for leakage reduction is to use a dual-Vt process and assign transistors a high or low Vt based on the circuit's leakage/performance requirements. A number of chips today cannot meet both their performance and leakage constraints with only a single Vt for all transistors. If a uniform high Vt is used, the leakage requirements are

	Delay (ns)	% Impr	Leakage (mA)	Increase
High Vt	-2.160	0	0.13	0 X
Low Vt	-1.530	29.2	5.46	42.00 X
Mixed Vt	-1.719	20.42	0.62	4.77 X

Figure 4. Leakage and performance with Dual-Vt optimization.

met but the performance of the part is lacking. On the other hand, if a uniform lower Vt is used, the performance goals of the chip are met but the leakage constraints of the design is exceeded. To overcome this problem, designers have started using dual-Vt processes, where transistors or gates can be assigned one of two threshold voltages.

The basic approach starts by setting all transistors to high Vt. The performance of the design is then improved by setting a few gates on the critical paths of the circuit to low-Vt. Since the majority of the gates remain at high-Vt, the leakage of the circuit remains low. More sophisticated methods for Vt optimization are needed in order to trade-off the leakage, area, and performance. Figure 4 shows the performance and leakage improvements possible by using a more advanced dual-Vt optimization approach. The first two rows show the performance and leakage of the circuit with all high and all low Vt transistors, respectively. The low Vt circuit implementation obtains a performance improvement of 29% over the high Vt implementation, but also increases the leakage by 42X. The third row shows a circuit implementation with mixed high and low Vt transistors yielding a performance improvement of 20%, while increasing the leakage by only 4.7X.

6. Conclusions

In this paper we have presented an overview of traditional power simulation tools, and discussed two emerging power management design technologies: power grid integrity analysis and standby current measurement and optimization. Power grid integrity analysis is becoming an increasingly important problem with the scaling of the supply voltage for processor designs. Also, power grid integrity analysis posses special requirements for power estimation, since the maximum instantaneous power is needed for worst case analysis. Requirements for the accuracy of the simulation and the generation and compression of the simulation vectors were discussed. Also, static approaches for calculating an upper-bound on the maximum peak current were presented. For standby current measurement, the state-dependent nature of the leakage current was discussed, as well as methods for calculating the average and maximum circuit leakage. Finally, optimization methods for minimizing the leakage current by either assigning a standby state to the circuit or by using a dual-Vt process were presented. Continuing reduction in feature sizes increases the need for further research and industrial application these of these emerging design technologies.

REFERENCES

[1] C. X. Huang, B Zhang, A-C Deng, and B, Swirski, "The design and implementation of PowerMill", in *Proc. International Symposium on Low Power Design*, pp. 105-120, 1995.

[2] S. Gavrilov, A. Glebov, S. Rusakov, D. Blaauw, L. Jones, and G. Vijayan, "Fast Power Loss Calculation for Digital Static CMOS Circuits", in *Proc IEEE European Design Automation Conference*, 1997.

[3] B. George, G. Yeap, M. G. Wloka, S. C. Tyler, and D. Gossain, "Power Analysis for Semi-Custom Design", in *Proc. Custom Integrated Circuits Conference*, pp. 249-252, May 1994.

[4] Toas User Manual, Motorola, Inc., 1996.

[5] Towards a high-level power estimation capability", in *Proc. International Symposium on Low Power Design*, pp. 87-92, 1995.

[6] David Blaauw, Abhijit Dharchoudhury, Rajendran Panda, David Bearden, and Bogdan Tutuianu, "Methodology for the Design and Analysis of Power Distribution Networks on the PowerPC Microprocessor", in *Proc. Design Automation Conference*, June 1998.

[7] M. S. Hsiao, E. M. Rudnick, J. Patel, "K2: An Estimator for Peak Sustainable Power of VLSI Circuits", in *Proc. International Symposium on Low Power Electronics and Design*, 1997.

[8] A. Krstic, K. T. Cheng, "Vector Generation for Maximum Instantaneous Current Through Supply Lines for CMOS Circuit", in *Proc. Design Automation Conference*, pp. 383-389, 1997

[9] H. Kriplani, F. Najm, I. Hajj, "Maximum Current Estimation in CMOS Circuits", in *Proc. Design Automation Conference*, 1992

[10] S. M. Sze, "Physics of Semiconductor Devices", New-York:John Wiley, 1981.

[11] Z. Chen, L. Wei, M. Johnson, K. Roy, "Estimation of Standby Leakage Power in CMOS Circuits Considering Accurate Modeling of Transistor Stacks", in *Proc. International Symposium on Low Power Electronics and Design*, 1998.

[12] S. Ercolani, M. Favalli, M. Damiani, P. Olivo, B. Ricco, "Testability Measures in Pseudorandom Testing", in *Trans. IEEE Trans. on Computer-Aided Design*, pp. 794-800, June 1993.

[13] J. P. Halter and F. Najm, "A gate-level leakage power reduction method for ultra-low-power CMOS Circuits", in *Proc. Custom Integrated Circuits Conference*, 1996.