01 Jul 2007

Leakage Minimization Technique for Nanoscale CMOS VLSI

Kyung Ki Kim

Yong-Bin Kim

Minsu Choi
Missouri University of Science and Technology, choim@mst.edu

Nohpill Park

Recommended Citation

K. K. Kim et al., "Leakage Minimization Technique for Nanoscale CMOS VLSI," *IEEE Design and Test of Computers*, vol. 24, no. 4, pp. 322-330, Institute of Electrical and Electronics Engineers (IEEE), Jul 2007.
The definitive version is available at https://doi.org/10.1109/MDT.2007.111

Leakage Minimization Technique for Nanoscale CMOS VLSI

Kyung Ki Kim and Yong-Bin Kim
Northeastern University

Nohpill Park
Oklahoma State University, Stillwater

Minsu Choi
University of Missouri-Rolla

Editor's note:
Leakage current is a major drawback in nanoscale CMOS. This article introduces a novel, practical approach that is amenable to CAD, to heuristically estimate this current during standby mode.
—Fabrizio Lombardi, Northeastern University

Because of the continued scaling of technology and supply-threshold voltage, leakage power has become more significant in power dissipation of nanoscale CMOS circuits. Therefore, estimating the total leakage power is critical to designing low-power digital circuits.

In nanometer CMOS circuits, the main leakage components are the subthreshold, gate-tunneling, and reverse-biased junction band-to-band-tunneling (BTBT) leakage currents. As transistor geometries decrease, it is necessary to reduce the supply voltage to avoid electrical breakdown and obtain the required performance. However, to retain or improve performance, it is necessary to reduce the threshold voltage (V_{TH}) as well, which results in an exponential increase of subthreshold leakage. To control short-channel effect and increase the transistor driving strength in deep-submicron (DSM) circuits, gate-oxide thickness also becomes thinner as technology scales down. The aggressive scaling in the gate oxide results in a tunneling current through the oxide, which is a strong exponential function of the oxide thickness and the voltage magnitude across the oxide. In scaled devices, the higher substrate doping density and the application of "halo" profiles cause significantly large reverse-biased junction BTBT leakage currents through the drain- and source-substrate junctions.[1–3]

This is a serious problem in portable electronic systems that operate mostly in sleep mode. To minimize leakage power dissipation, researchers have proposed several circuit techniques such as multi-threshold-voltage CMOS (MTCMOS) and variable-threshold-voltage CMOS (VTCMOS) using variable substrate bias voltage. However, these techniques require significant circuit modification and performance overhead for leakage reduction.[1,3]

Another technique with little or no overhead is the input pattern control technique based on the stack effect: the amount of leakage current of a nanometer CMOS circuit varies depending on the input pattern. However, it is an NP-hard problem to determine the input pattern that sets up the minimum leakage current during standby mode without any hardware overhead or architecture change. Researchers have proposed several techniques to generate the minimum leakage test pattern and to solve the NP-hard problem,[4–6] but none of these techniques explicitly considers the interactions among the subthreshold leakage, gate-tunneling leakage, body effect, and fan-out effect. Although some articles have been published on modeling subthreshold leakage and gate-tunneling leakage, they neglect the interactions among the three leakage components. They also do not consider the fan-out effect in the leakage current.[4–9]

0740-7475/07/$25.00 © 2007 IEEE Copublished by the IEEE CS and the IEEE CASS

Therefore, this problem requires a better understanding and a more accurate model of leakage currents for input pattern control in nanometer CMOS circuits.

In this article, we investigate all the possible leakage current components and propose a novel macromodeling technique to characterize the minimum leakage current of each individual leaf cell, considering fan-out effect, stack effect, and the interaction between gate-leakage and subthreshold currents. We developed a heuristic approach to find the input pattern for minimum leakage current during standby mode of nanoscale VLSI chips. Our approach uses the functional dependencies in VLSI and the controllability of its nodes.

Leakage power model and analysis

In the *off* state, the main components of leakage current are subthreshold leakage (I_{SubTH}), gate-induced drain leakage (I_{GIDL}), gate-tunneling leakage (I_{GATE}), and band-to-band tunneling (I_{BTBT}), as Figure 1a shows. Gate-tunneling leakage (I_{GATE}) is the major component during the *on* state, as Figure 1b shows.[10]

The GIDL is a current from the drain to the substrate caused by the high electric field between the gate and the drain; thin gate-oxide thickness and a high supply voltage increase GIDL. The gate-tunneling leakage is a current flowing into the transistor's gate by a tunneling effect; thin gate-oxide thickness and a high supply voltage also increase gate-tunneling leakage. The subthreshold leakage is a weak inversion conduction current of the CMOS transistor when V_{GS} is less than V_{TH}. It increases exponentially because of the reduced threshold voltage, and it is a main leakage component in the case of a high forward body bias. Finally, the BTBT leakage is a current by electron tunneling across the reverse-biased pn junction between the drain or source and the CMOS transistor's substrate. Therefore, in the case of a high reverse body bias, the BTBT leakage becomes a major portion of the total leakage current. Kuroda et al. show that the subthreshold leakage current and the BTBT leakage are more sensitive to the applied body bias than the other two leakage components.[10] The minimum leakage current is obtained when the subthreshold leakage current is equal to the BTBT leakage.

Gate-tunneling leakage current

Gate-tunneling leakage is a current flowing (tunneling) into the transistor's gate. With an increase of gate-oxide thickness, the tunneling drops exponentially. This is given by

$$I_{\text{gate\_tunneling}} = (AC)(WL)e^{-B\left(\frac{T_{ox}}{V_{GS}}\right)^{\alpha}} \quad (1)$$

where $A = q^3/(8\pi h\phi_b)$, $B = 8\pi\,(2m_{ox})^{1/2}\phi_b^{3/2}/(3hq)$, $C = (V_{GS}/T_{ox})^2$, and α is a parameter that ranges from 0.1 to 1, depending on the voltage drop across the oxide. Here, h is Planck's constant, and ϕ_b is the barrier height for electrons and holes in the conduction and valence band.

In nanometer CMOS technology, the gate-tunneling leakage current is expected to increase to more than twice the subthreshold leakage current. Figure 2a shows the gate-tunneling leakage currents produced by a nanoscale n-channel MOS transistor. As Figure 2a shows, the gate-tunneling current consists of four components: gate-to-channel tunneling (I_{GC}), gate-to-drain edge tunneling (I_{GD}), gate-to-source edge tunneling (I_{GS}), and gate-to-body tunneling (I_{GB}). The gate tunneling's magnitude depends on the applied voltage V_{GS}. For NMOS, four possible states exist, depending on the voltages of the three terminals

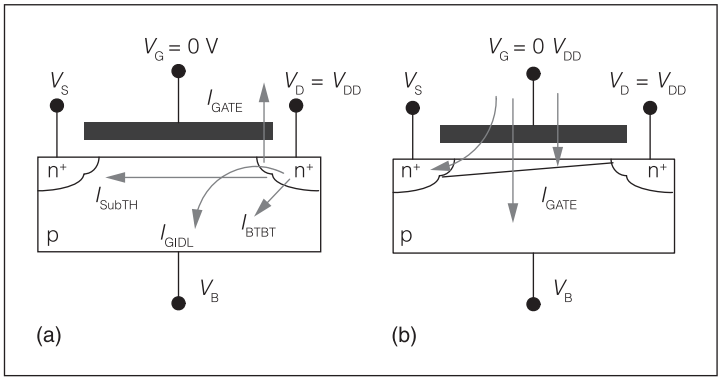

Figure 1. Leakage current in a nanoscale CMOS circuit: off-state (a) and on-state (b) leakage components.

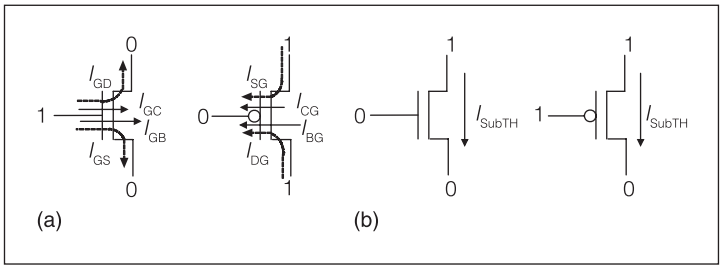

Figure 2. The gate-tunneling and subthreshold leakage current in NMOS and PMOS transistors: maximum gate-tunneling leakage current state (a) and maximum subthreshold leakage current state (b).

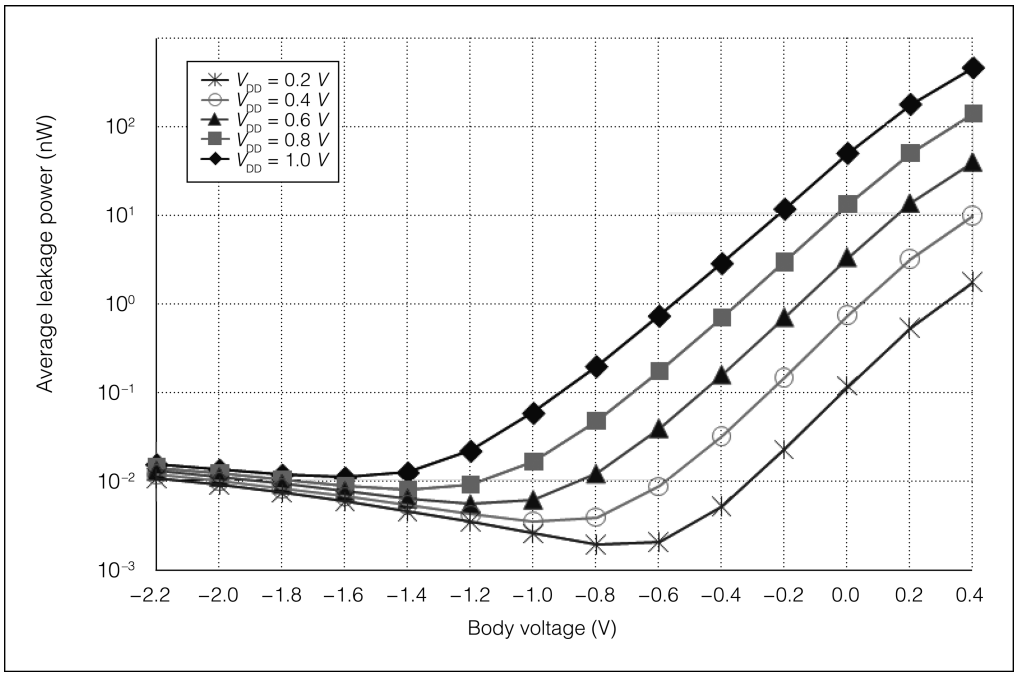

Figure 3. Leakage power as a function of body-bias voltage and supply voltage.

(drain, gate, source): (1, 0, 0), (1, 0, 1), (0, 0, 1), and (0, 1, 0). The leakage current under the (0, 1, 0) state is the highest due to the strong inversion. For a p-channel MOS transistor, the current direction and the voltages are the opposite of the NMOS transistor (as Figure 2a shows). Because holes must pass a higher barrier to tunnel, the PMOS tunneling current is less than the NMOS tunneling current.[11,12]

Subthreshold leakage current

Even though the transistor's gate voltage decreases below V_{TH}, a small current still flows between the source and drain terminals, as Figure 2b shows.[1–3] The subthreshold leakage current is given by

$$I_{\text{subthreshold}} = I_0 e^{-B\left(\frac{V_{GS}-V_{TH}}{\eta kT/q}\right)}\left(1 - e^{-\frac{V_{DS}}{kT/q}}\right) \qquad (2)$$

where $I_0 = \mu_0 C_{ox}(W/L)(kT/q)^2(1 - e^{1.8})$, W and L are the transistor channel width and length, μ_0 is the low field mobility, C_{ox} is the gate-oxide capacitance, k is Boltzmann's constant, q is the electronic charge, and N is the subthreshold swing factor.

BTBT leakage and body effect of nanoscale CMOS gates

In a high electric field (greater than 10^6 volts per centimeter), electrons tunnel across the reverse-biased pn junction of drain and source substrates, in what is known as junction BTBT. If both n and p regions are heavily and halo-doped shallow junctions, the BTBT significantly increases and becomes a major contributor to the total leakage current.[3]

Figure 3 shows the effect of body-bias voltage and supply voltage on leakage power for a 45-nm NMOS transistor. Figure 3 also shows that as supply voltage and body-bias voltage decrease, leakage power decreases because of I_{SubTH} reduction. On the other hand, leakage current starts increasing as the body bias decreases from −0.9 to −2.2 V because the increase of I_{BTBT} becomes dominant over the subthreshold leakage current. Therefore, the optimal body-bias voltage to minimize the power dissipation is determined by the relationship between I_{SubTH} and I_{BTBT}.

Minimum leakage pattern generation

Because of stack effect, fan-out effect, and the interaction between gate-leakage and subthreshold currents, the leakage power of a circuit under standby mode depends on its input pattern. However, it is an NP-hard problem to determine the input pattern. An accurate heuristic approach is required to solve the NP-hard problem for better accuracy.

Stack effect

When there are two or more stacked transistors, the subthreshold leakage is reduced. This reduction depends on the choice of the input pattern during standby periods because it determines the number of *off* transistors in the stack. Turning off more than one transistor in a stack forces the intermediate node voltage to go to a higher value than 0.[12] This causes a negative V_{GS}, a negative V_{BS} (more body effect), and a V_{DS} reduction (less DIBL) in the top transistor closest to the outputs, thereby reducing the subthreshold leakage current flowing through the stack consider-

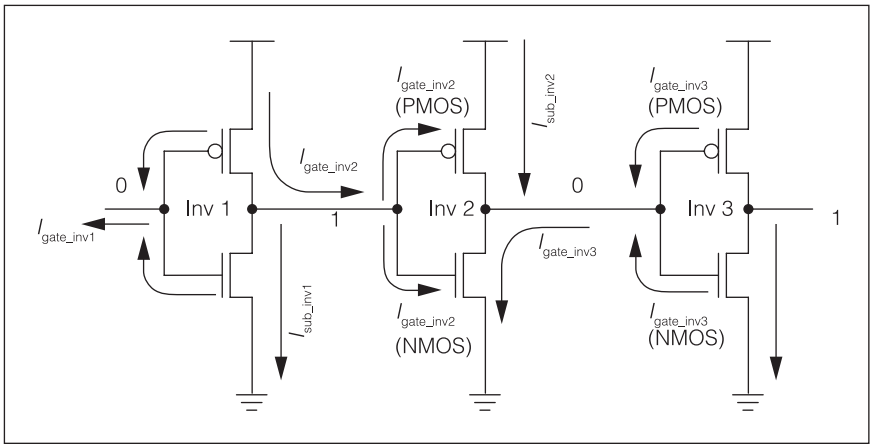

Figure 4. Leakage current flows in nanoscale CMOS circuits.

ably, which is known as the *stack effect*. The leakage current decreases monotonously with the number of stacked *off* transistors.

Because of the transistor stack effect, a gate's leakage current depends on its input combination. An individual CMOS gate shows a variation in the leakage power for different input patterns. Only a few input patterns, defined as *dominant leakage states*, cause significant leakage. Therefore, 011, 101, 110, and 111 input patterns of a 3-input NAND gate are dominant leakage states. If the dominant leakage states of all input patterns are used to generate macromodels, the lookup table's size for the macromodels is reduced by ignoring the rest.

Figure 4 shows the static current paths that appear when the leakage current is considered in CMOS circuits. In the circuits, each inverter has a few paths of subthreshold and gate-tunneling leakage current. We can assume that Inv 2 is the device under test (DUT), and the input of Inv 1 is 0. Inv 2 has three leakage components, which depend on the fan-out structures of Inv 2:

- the gate-tunneling current I_{gate_inv2} starting from the PMOS of Inv 1,
- the subthreshold leakage of the turned-off state PMOS in Inv 2 (I_{sub_inv2}), and
- the gate-tunneling current I_{gate_inv3} starting from Inv 3.

Therefore, the total leakage current is the sum of I_{gate_inv2}, I_{sub_inv2}, and I_{gate_inv3}. However, when a cell's macromodel is generated, one leakage tunneling current (I_{gate_inv3}) should be removed to make sure the leakage components are not counted twice when the total leakage currents of Inv 3 are calculated. In Figure 4, I_{gate_inv3} is the gate-tunneling leakage of Inv 3; that is, only I_{gate_inv2} and I_{sub_inv2} are the leakage currents of Inv 2.

Fan-out effect

Depending on the primary input (PI) pattern, the subthreshold leakage current and gate tunneling are affected by the adjacent fan-in and fan-out logic circuits. Figure 5 illustrates the dependency of the leakage current on the fan-out structures. In Figure 5, the PI is logic 1, the number of fan-outs of inverter G2 is two, and the number of fan-outs of inverter G3 is three. First, the current I_{gG3} is the gate-tunneling leakage of inverter G3. In this circuit, I_{gG2} and I_{gG4} are the gate-tunneling leakage currents of G2 and G4, respectively. The directions of the three currents converge into the input of inverter G3.

The sum of the gate leakage currents at node N3 is a function of gate G1's fan-out and the subthreshold currents of G2, G3, and G4. The 0 state voltage at node N3 increases as the fan-out of G1 increases, which in turn reduces the gate-leakage current of G2, G3, and G4 because the voltages between the inputs and outputs of those gates are reduced. The gate-leakage currents of G2, G3, and G4 are also a function of their subthreshold currents because the subthreshold currents affect the voltage between those gates' inputs and outputs. Considering these fan-out effects, I_{gG3} is about one-third of the gate-tunneling leakage when G1 has only one fan-out. Consequently, the subthreshold current is influenced by the number of fan-outs of the previous driver. However, the fan-out of inverter G3 cannot have a significant effect on the leakage current of inverter G3. As the number of fan-outs for G3 increases, G3's output voltage decreases, which then reduces its subthreshold and gate-tunneling leakage currents.

Therefore, the total leakage of inverter G3 is affected by the fan-outs of G1 and G3. Hence, it is necessary to consider the interaction of each leakage current component in both previous stages and the

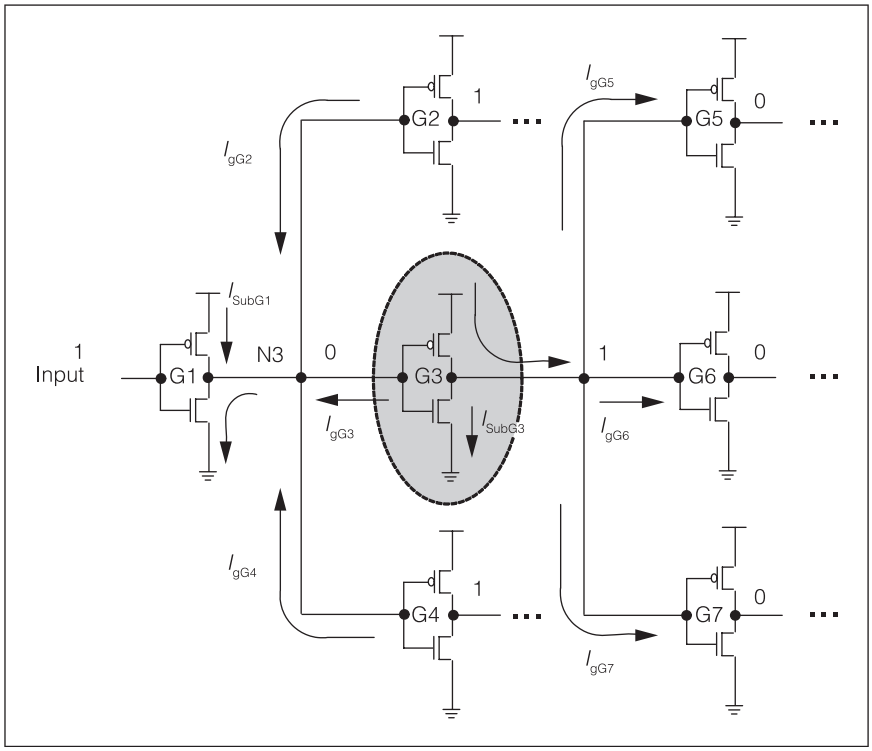

Figure 5. Fan-out effect for the G3 gate. The leakage currents depend on the fan-out structures of the G3 gate. [*W*/*L* of PMOS is (90 nm)/(45 nm), and *W*/*L* of NMOS is (45 nm)/(45 nm), where *W* and *L* are the transistor channel width and length.]

next stages for an accurate leakage estimate in nanoscale CMOS circuits. However, the effects of the leakage current components beyond one logic level from the DUT are negligible.

Figure 6 presents the fan-out effect on the leakage current for inverter G3 shown in Figure 5. The leakage currents are measured at inverter G3 in Figure 5. The number of fan-outs of G1 varies from 1 to 5, as does the number of fan-outs of G3. Figure 6a and Figure 6b show the subthreshold leakage and gate tunneling, respectively, when the input of inverter G3 is 1. Figure 6c and Figure 6d show the subthreshold leakage and gate tunneling, respectively, when the input of inverter G3 is 0.

As expected, the number of fan-outs of G1 affects the leakage current. For the 0 input to G3, the fan-outs of G3 significantly affect the leakage current, but less than the fan-outs of the previous driver. We measured the smallest total leakage (0.73 μA) for the 1 input with five fan-outs of G1 and five fan-outs of G3. We see the highest total leakage (2.33 μA) for the 0 input with one fan-out of G1 and one fan-out of G3. If we do not consider the fan-out effect when modeling the leakage current, the smallest total leakage is 3.34 μA under the 0 input, and the largest total leakage is 6.19 μA under the 1 input. In large circuits, depending on the applied input vector, leakage of different logic gates moves in different directions (some increase and some decrease). Because of this, in such circuits, the net change in overall leakage due to the fan-out effect tends to be reduced by averaging out the individual gate's leakage current change.

Input-pattern generation

Based on the fan-out effect in leakage current, we developed the macromodel for a leaf cell (inverter, NAND, and NOR gates) based on an Hspice simulation, where the controlling variables are the number of fan-outs, the cell size, and the input pattern, considering the stack effect under fixed V_{DD}, V_{TH}, T_{ox}, and temperature. On the basis of the accurate macromodel of the cells, we implemented a heuristic approach to generate the minimum leakage test pattern. The leakage of each cell in the circuit depends on the input pattern applied to the circuits. Researchers have proposed several techniques to generate the input pattern for minimum leakage current and solve the NP-hard problem.[4–6] An easy way to solve the problem is to use the functional dependencies in the circuits and the controllability of the nodes. In this article, we improve the methodology to estimate the accurate leakage current with a fast simulation time.

The functionality of the cells in circuits determines the states of the internal nodes for any given input vector. A cell is *dominated* if its input pattern causing the minimum leakage current is a subset of the minimum leakage input pattern of the other cells. A cell is *conflicting* if the input pattern causing the minimum leakage conflicts with other cells' input vectors that cause the minimum leakage.

Before finding the optimal input pattern to reduce leakage power dissipation, the functional dependencies between cells should be searched, and each cell's dominated and conflicting cells should be listed in the order of the weight function, which is given by

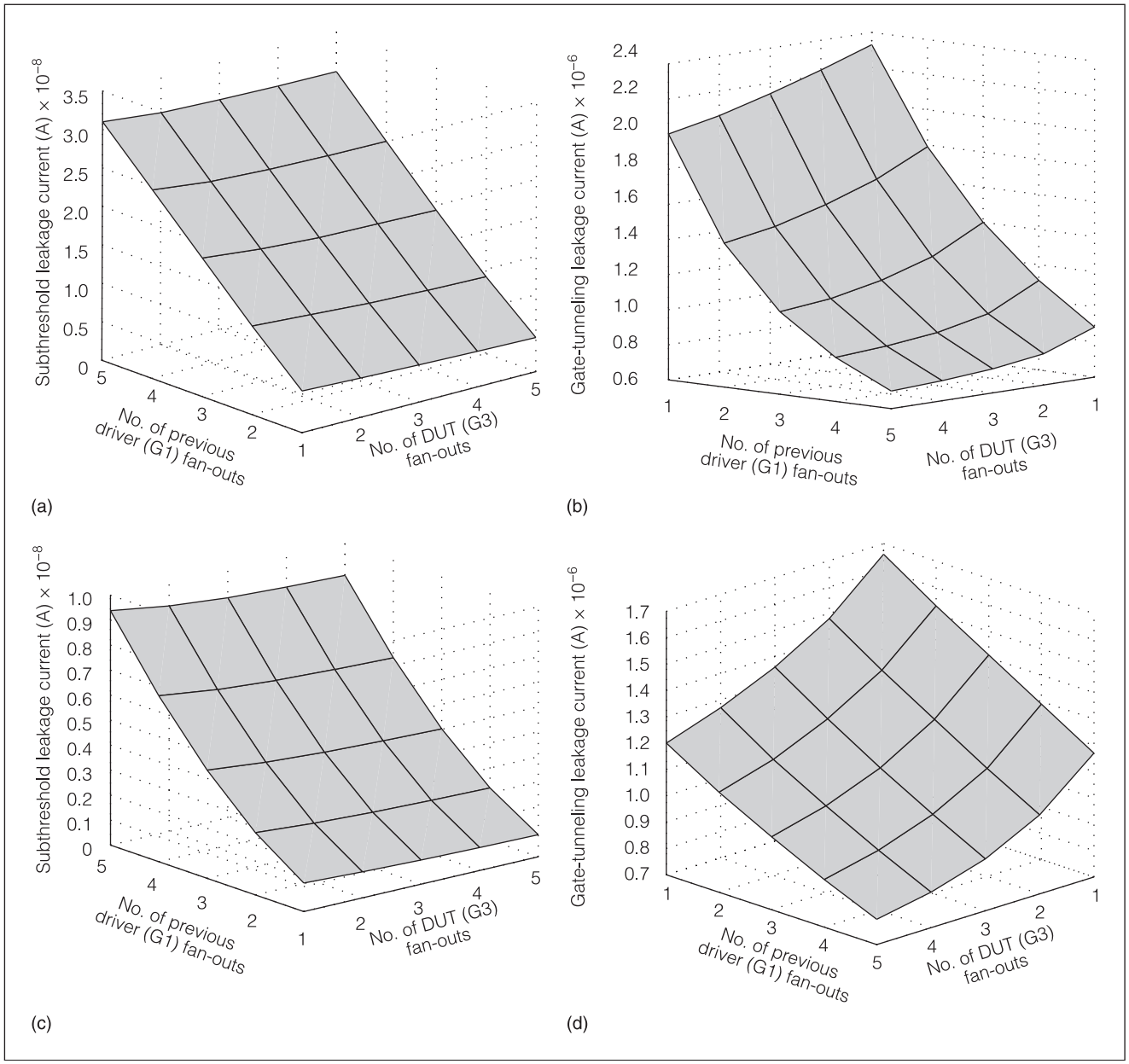

Figure 6. Leakage current variation due to fan-out effect in BSIM4 45-nm technology: subthreshold leakage current with input 1 (a), gate-tunneling leakage current with input 1 (b), subthreshold leakage current with input 0 (c), and gate-tunneling leakage current with input 0 (d).

$$W_c(G_i) = \sum_i (MLK_{cc}(G_i)) - \sum_i (MLK_{dc}(G_i)) - MLK_{fG} - MLK(G_i) \quad (3)$$

where $W_c(G_i)$ is the weight of cells G_i; and MLK_{cc}, MLK_{dc}, and MLK_{fG} are the mean leakage for conflicting cells, dominating cells, and fan-out of G_i, respectively.

Once the list is determined, the cell that has the least weight function will be selected. If the cell satisfies the functional constraints for minimum leakage current, the PI patterns controlled by the cell are determined. After finding the proper input patterns, the cell is removed from the list, and at the same time the cell's dominated and conflicting cells are removed. This procedure is repeated until there are no cells in the list or only undefined cells remain. If the

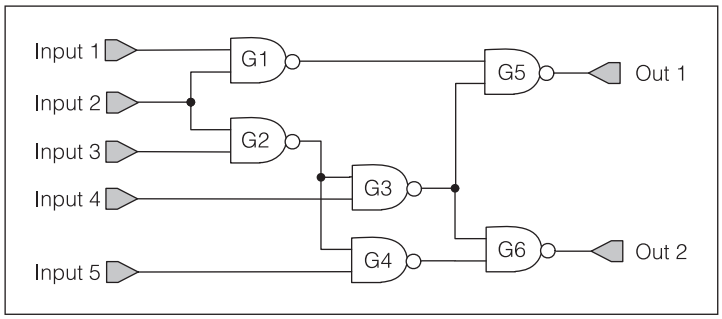

Figure 7. ISCAS85 benchmark circuit C17, which is used as an example to explain our proposed heuristic algorithm.

undefined cells are found, proper patterns must be assigned, considering the conditions for low leakage current, because these cells have no dominated or conflicting cells.

The example ISCAS85 benchmark circuit C17 shown in Figure 7 explains this proposed heuristic approach in detail. As we described at the beginning of this subsection, a macromodel is generated for all the cells in the library, and the macrocells contain all the leakage current information for all the possible input and fan-out cases. For example, the two-input NAND gate in Figure 7 has the leakage currents shown in Table 1 for different inputs and a fan-out of 2.

In Table 1, Input 1 represents the input to the transistor closest to the output. For a NAND gate, the total leakage is the lowest when the inputs are (1, 0) and the gate leakage current is dominant over the subthreshold current. If the subthreshold current is larger than the gate leakage, then the total leakage is lowest when the inputs are (0, 0). In this article, we consider nanoscale CMOS technology that has a larger gate-tunneling current than the subthreshold current. The primary-input patterns for minimum leakage current and the lists of dominating and conflicting cells are created using the test controllability lists of the C17 circuit. The possible input patterns can be obtained from the macromodel of each cell.

The detailed procedure is as follows: On the basis of the leakage current information of each cell from the macromodel and the controllability for the C17 circuit, PI patterns that minimize leakage current are generated, then the conflicting and dominated cell lists are identified, as Table 2 shows. However, the input vector (1, 0) to gate G1 conflicts with the G2 and G5 inputs. Therefore, the input pattern (1, 0) must be changed to the input pattern that causes the next-least leakage current, which is (0, 0), as Table 1 shows. This is a trade-off between the minimum leakage current and the conflicting input vectors. Consequently, the minimum leakage input patterns for G1 and G2 are changed to 00XXX and X00XX, respectively. The conflicting cells for G1 and G2 are removed from the list, and G1 among the conflicting cells for G5 is also removed because G1 no longer conflicts with G5. At the same time, the weight function for each cell must be updated using Equation 3, and a cell with the least weight function must be identified. In this case, G1 and G2 are the cells with the least weight function, so the two cells are removed from the list, and the 000XX input pattern is determined by combining 00XXX and X00XX. G4 is the next cell that has the least weight function among the remaining cells. If G4 is removed from the list, G4 and its conflicting cell G6 are also removed, and the 000X0 pattern is determined. Finally, G3 and G5 are left. Because the weight function of G5 is smaller than G3, G5 is removed from the list, and the input pattern becomes 00010, which is the final input pattern.

Table 1. Total leakage current for a two-input NAND gate.

Input 1	Input 2	Total leakage current (μA)
0	0	0.33
0	1	1.01
1	0	0.15
1	1	1.61

Experimental results

We implemented our minimum leakage test pattern generator for nanometer CMOS gates in Hspice and C language and ran it on a 500-MHz UltraSparc-IIe with 500 Mbytes of memory. We were able to demonstrate the algorithm using the results from various ISCAS85 benchmark circuits. Each benchmark circuit was designed using Hspice in a 45-nm BSIM4 model. The minimum leakage current was measured using Monte Carlo simulations with 50 repetitions in Hspice. Monte Carlo simulations let us iteratively evaluate a deterministic model using many sets of random numbers as inputs. These simulations generated many random inputs, and we ran Hspice simulations for the circuit with those inputs; we then selected the least leakage from the results. In addition, we compared the results of our proposed methodology with and without the fan-out effect.

Table 3 compares the simulation results for our proposed method with other simulations. Our method's accuracy is within a 4% difference of the Hspice

Table 2. Input patterns for minimum leakage

Cell	Minimum leakage input pattern of each cell: Input 1	Input 2	Minimum leakage input pattern: A1	A2	A3	A4	A5	Conflicting cells
G1	1	0	1	0	X	X	X	G2, G5
G2	1	0	X	1	0	X	X	G1
G3	1	0	X	X	0	0	X	G5
G4	1	0	X	X	0	X	0	G6
G5	1	0	0	X	0	1	X	G1, G3, G6
G6	1	0	X	X	X	0	1	G4, G5

Table 3. Leakage estimation results for benchmark circuits.

Circuit	No. of gates	Estimated leakage current (μA): Without fan-out effect	With fan-out effect	Hspice minimum leakage (μA)	Error rate (%)	CPU runtime: Our work (ms)	Hspice (s)
C432	160	3.75	1.49	1.53	2.70	0.89	9.45
C499	202	10.48	6.78	7.00	3.20	5.60	40.34
C880	383	9.67	5.34	5.40	1.20	3.50	28.88
C1355	546	16.54	7.43	7.21	3.90	5.87	40.41
C1908	880	15.53	6.84	6.57	4.00	6.65	35.98
C2670	1,193	18.43	11.14	10.75	3.50	9.88	64.83
C5315	2,307	35.12	29.54	29.18	1.22	16.30	185.98
C6288	2,388	33.48	26.57	25.88	2.60	42.57	987.45
C7552	1,916	35.67	28.57	29.38	2.83	23.84	235.87

simulation results. In addition, the simulation time of our method is far faster than that of the Hspice simulation.

It is not easy to compare our method with alternate approaches, because most of the previously published research requires significant circuit modification and too much performance overhead and they do not consider fan-out effects. Some of the previous research requires little or no overhead, but they are NP-hard problems. Our heuristic approach solves the previous approaches' problems, considering all the leakage components and the relationships between them. The best way to compare the efficiency of the algorithms is to compare the complexity of the algorithms. The proposed approach's complexity is $O(n^2)$, whereas the other algorithms' complexities are far higher, and the simulation time of our approach is faster than the previous approaches by at least a factor of 5.

THE REMAINING ISSUE is to consider process, voltage, and temperature (PVT) variations for a more accurate macromodel, because the leakage current below 90-nm technology is sensitively affected by PVT variations. ■

■ References

1. K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand, "Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits," *Proc. IEEE*, vol. 91, no. 2, Feb. 2003, pp. 305-327.
2. A. Agarwal et al., "Leakage Power Analysis and Reduction: Models, Estimation and Tools," *IEE Proc. Computers & Digital Techniques*, vol. 152, no. 3, May 2005, pp. 353-368.
3. S.G. Narendra and A. Chandrakasan, *Leakage in Nanometer CMOS Technologies, Series on Integrated Circuits and Systems*, Springer, 2005.
4. Y. Xu and Z. Luo, "Minimum Leakage Pattern Generation Using Stack Effect," *Proc. 5th Int'l Conf. ASIC*, IEEE Press, 2003, pp. 1239-1242.
5. F. Gao and J.P. Hayes, "Exact and Heuristic Approaches to Input Vector Control for Leakage Power Reduction,

Computer Aided Design," *Proc. Int'l Conf. Computer-Aided Design* (ICCAD 04), IEEE CS Press, 2004, pp. 527-532.
6. X. Chang et al., "SoC Leakage Power Reduction Algorithm by Input Vector Control," *Proc. Int'l Symp. System-on-Chip*, IEEE Press, 2005, pp. 86-89.
7. Y. Xu, Z. Luo, and X. Li, "A Maximum Total Leakage Current Estimation Method," *Proc. Int'l Symp. Circuits and Systems* (ISCAS 04), vol. 2, IEEE Press, 2004, pp. 757-760.
8. Y. Xu et al., "Average Leakage Current Macromodeling for Dual-Threshold Voltage Circuits," *Proc. 12th Asian Test Symp*, (ATS 03) IEEE CS Press, 2003, pp. 196-201.
9. R. Kumar and C.P. Ravikumar, "Leakage Power Estimation for Deep Submicron Circuits in an ASIC Design Environment," *Proc. Conf. Asia South Pacific Design Automation / VLSI Design* (ASP-DAC/VLSID 02), IEEE CS Press, 2002, pp. 45-50.
10. T. Kuroda et al., "Variable Supply-Voltage Scheme for Low-Power High Speed CMOS Digital Design," *IEEE J. Solid-State Circuits*, vol. 33, no. 3, Mar. 1998, pp. 454-462.
11. S. Yang et al., "Accurate Stacking Macro-modeling of Leakage Power in Sub-100 nm Circuits," *Proc. 18th Int'l Conf. VLSI Design* (VLSID 05), IEEE CS Press, 2005, pp. 165-170.
12. D. Lee et al., "Analysis and Minimization Techniques for Total Leakage Considering Gate Oxide Leakage," *Proc. 40th Design Automation Conf.* (DAC), ACM Press, 2003, pp. 175-180.

Kyung Ki Kim is a PhD candidate in the Department of Electrical and Computer Engineering at Northeastern University. His research interests include high-speed, low-power VLSI design; analog VLSI circuit design; electronic CAD; and ATE system design. Kim has a BS and an MS in electronics engineering from Yeungnam University, Kyungsan, South Korea.

Yong-Bin Kim is an associate professor in the Department of Electrical and Computer Engineering at Northeastern University. His research focuses on high-speed, low-power VLSI circuit design. Kim has a BS from Sogang University, Seoul, an MS from the New Jersey Institute of Technology, and a PhD from Colorado State University, all in electrical engineering. He is senior member of the IEEE.

Minsu Choi is an associate professor in the Department of Electrical and Computer Engineering at the University of Missouri-Rolla. His research interests include computer architecture and VLSI, embedded systems, fault tolerance, testing, quality assurance, reliability modeling and analysis, configurable computing, parallel and distributed systems, instrumentation and measurement, and computational nanotechnology. Choi has a BS, an MS, and a PhD in computer science from Oklahoma State University, Stillwater. He is a member of Sigma Xi and of the Golden Key National Honor Society.

Nohpill Park is an associate professor in the Computer Science Department at Oklahoma State University, Stillwater. His research interests include computer architecture, defect- and fault-tolerant systems, testing and quality assurance of digital systems, parallel and distributed computer systems, multichip-module systems, programmable digital systems, and SoCs. Park has a BS and an MS in computer science from Seoul National University and a PhD in computer science from Texas A&M University.

■ Direct questions and comments about this article to Yong-Bin Kim, Dept. of Electrical and Computer Engineering, Northeastern University, 442 Dana Research Center, 360 Huntington Ave., Boston, MA 02115; ybk@ece.neu.edu.