Journal of
Low Power Electronics
Vol. 2, 1–10, 2006

CMOS Leakage and Glitch Minimization for Power-Performance Tradeoff

Yuanlin Lu and Vishwani D. Agrawal*

Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849, USA

(Received: 22 June 2006; Accepted: 13 October 2006)

A mixed integer linear programming (MILP) technique simultaneously minimizes the leakage and glitch power consumption of a static CMOS circuit for any specified input to output delay. Using dual-threshold devices the number of high-threshold devices is maximized and a minimum number of delay elements are inserted to reduce the differential path delays below the inertial delays of incident gates. The key features of the method are that the constraint set size for the MILP model is linear in the circuit size and power-performance tradeoff is allowed. Experimental results show 96%, 40%, and 70% reductions of leakage power, dynamic power, and total power, respectively, for the benchmark circuit C7552 implemented in the 70 nm BPTM CMOS technology.

Keywords: Dual-Threshold CMOS Circuits, Dynamic Power, Leakage Reduction, Low Power Design, Glitch-Free Design, Mixed Integer Linear Programming (MILP).

1. INTRODUCTION

In the past, the dynamic power has dominated the total power dissipation of CMOS devices. However, with the continuous trend of technology scaling, leakage power is becoming a main contributor to power consumption. To reduce leakage power, several techniques have been proposed, including transistor sizing, multi-V_{th}, dual-V_{th}, optimal standby input vector selection, stacking transistors, dual V_{dd}, etc. Among these, the dual-V_{th} assignment is an efficient technique for decreasing leakage power. Its basic idea is to utilize the timing slack of non-critical paths to assign high V_{th} to gates on those paths to decrease the leakage. There are heuristic algorithms[8, 12, 20–24] that search for an optimal solution of dual-V_{th} assignment. For example, the *backtrace algorithm*[21, 22] can determine a dual-V_{th} assignment for a possible solution without guaranteeing an optimal one (see example of Fig. 10). Because the backtrace search direction for non-critical paths is from primary outputs to primary inputs, the gates close to the primary outputs have a higher priority for high V_{th} assignment, even though their leakage power savings may be smaller than those of gates close to the primary inputs. Wang et al.[20] treat the dual-V_{th} assignment as a constrained 0-1 programming problem with non-linear constraint functions. They use a heuristic algorithm based on circuit graph enumeration to solve this problem. Although their *swapping algorithm* tries to avoid local optimization, a global optimization is still not guaranteed.

By describing both the objective function and constraints as linear functions, linear programming (LP) can easily get a globally optimum solution. Nguyen et al.[11] use LP to minimize the leakage and dynamic power by gate sizing and dual-threshold voltage device assignment. The optimization work is separated into several steps. An LP is first used to distribute slack to gates with the objective of maximizing total power reduction. Then, another independent algorithm resizes gates and assigns threshold levels. This means that the LP still needs the assistance of a heuristic algorithm to complete the optimization.[11] Gao and Hayes[5] use mixed integer linear programming (MILP) to optimize the total power consumption by dual-threshold assignment and gate sizing.

The techniques cited above[5, 8, 11, 12, 20, 22, 23] have not considered the glitch power, which can account for 20%–70% of the dynamic switching power.[4] To eliminate these unnecessary transitions, a designer can adopt techniques of hazard filter[2, 25–28] and path balance.[3, 14, 29] In Hazard filtering, gate sizing or transistor sizing is used to increase a gate inertial delay which can filter the glitches. An obvious disadvantage of hazard filtering, when used alone, is that it may increase the circuit delay due to the increase of the gate delay. Alternatively, any given performance can be maintained by path delay balancing, although the area overhead and additional power consumption of the inserted delay elements can become a major concern.

*Author to whom correspondence should be addressed.
Email: vagrawal@eng.auburn.edu

In the present research, a new MILP model is proposed to minimize leakage power by dual-V_{th} assignment and simultaneously eliminate dynamic glitch power by inserting zero-subthreshold delay elements to balance path delays. To our knowledge, no previous work on optimizing dynamic and static power has adopted such a combined approach. This MILP method is specifically devised with a set of constraints whose size is linear in the number of gates. Thus, large circuits can be handled. Although theoretical worst-case complexity of MILP is exponential, actual complexity depends on the nature of the problem. A discussion about this point is presented at the end of Subsection 6.1. To deal with the complexities of delay models and leakage calculation, two look up tables for the delay and leakage current for both low and high threshold versions are constructed in advance for each cell. This greatly simplifies the optimization procedure.

To further reduce power, other approaches such as gate sizing can be easily implemented by extending our cell library and look up tables. However, a dual-V_{dd} technique may require additional considerations beyond the delay look up tables for low V_{dd} and high V_{dd}.

This paper is organized as follows. Section 2 presents the necessary background knowledge about subthreshold leakage, delay, and glitches. Section 3 proposes the mixed integer linear programming for power minimization. Sections 4 and 5 discuss the implementation of delay elements for glitch elimination and the superiority of MILP, respectively. In Section 6, experimental results are presented and discussed. A conclusion is given in Section 7. Some work from this paper has appeared in a recent presentation by the authors.[9]

2. BACKGROUND

2.1. Leakage and Delay

The leakage current of a transistor is mainly the result of reverse biased PN junction leakage and subthreshold leakage. Compared to the subthreshold leakage, the reverse bias PN junction leakage can be ignored. The subthreshold leakage is the weak inversion current between source and drain of an MOS transistor when the gate voltage is less than the threshold voltage.[24] It is given by:[7]

$$I_{sub} = I_{s0} \exp\left(\frac{V_{gs} - V_{th}}{nV_T}\right)\left(1 - \exp\left(\frac{-V_{ds}}{V_T}\right)\right) \quad (1)$$

$$I_{s0} = \mu_0 C_{ox} \frac{W_{eff}}{L_{eff}} V_T^2 e^{1.8} \quad (2)$$

where μ_0 is the zero bias electron mobility, n is the subthreshold slope coefficient, V_{gs} and V_{ds} are the gate-to-source voltage and drain-to-source voltage, respectively, V_T is the thermal voltage, V_{th} is the threshold voltage, C_{ox} is the oxide capacitance per unit area, and W_{eff} and L_{eff} are the effective channel width and length, respectively. Due to the exponential relation between V_{th} and I_{sub}, an increase in V_{th} sharply reduces the subthreshold current.

Table I. Leakage currents for low and high V_{th} NAND gates.

	I_{leak} (nA)		
Input vector	Low V_{th}	High V_{th}	Reduction (%)
00	1.7360	0.0376	97.8
01	10.323	0.2306	97.8
10	15.111	0.3433	97.7
11	17.648	0.3169	98.2

Our Spice simulation results on the leakage current of a two-input NAND gate are given in Table I for 70 nm BPTM CMOS technology[1] ($V_{dd} = 1$ V, Low $V_{th} = 0.20$ V, High $V_{th} = 0.32$ V). The leakage current of a high V_{th} gate is only about 2% of that of a low V_{th} gate. If all gates in a CMOS circuit could be assigned the high threshold voltage, the total leakage power consumed in the active and standby modes can be reduced by up to 98%, which is a significant improvement. However, according to the following equation, the gate delay increases with the increase of V_{th}.

$$T_{pd} \propto \frac{CV_{dd}}{(V_{dd} - V_{th})^{\alpha}} \quad (3)$$

where α equals 1.3 for short channel devices.[17] Table II gives the delays of NAND gates obtained from Spice simulation when the output fans out to varying numbers of inverters. We observe that by increasing V_{th} form 0.20 V to 0.32 V, the gate delay increases by 30%–40%.

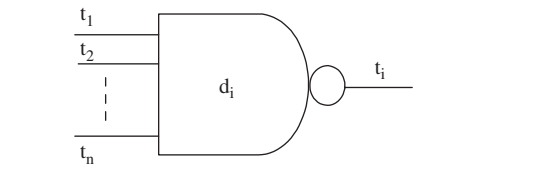
We can make tradeoffs between leakage power and performance, leading to a significant reduction in the leakage power while sacrificing only some or none of circuit performance. Such a tradeoff is made in MILP. Results in Section 6.1 show that the leakage power of all ISCAS85 benchmark circuits can be reduced by over 90% if the delay of the critical path is allowed to increase by 25%.

2.2. Glitch Elimination Techniques

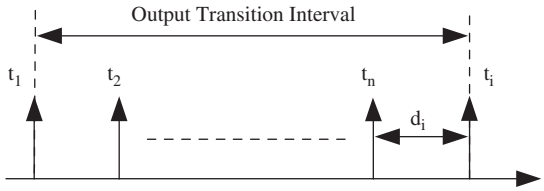
When transitions are applied at inputs of a gate, the output may have multiple transitions before reaching a steady state (Fig. 1 and Fig. 2(a)). Among these, at most one is an essential transition, and all others are unnecessary transitions often called *glitches* or *hazards*. Because switching power consumed by the gate is directly proportional to the number of output transitions, glitches reportedly account for 20%–70% dynamic power.[4] Agrawal et al.[3] prove that a combinational circuit is minimum transient energy design, i.e., there is no glitch at the output of any gate, if the

Table II. Delays of low and high V_{th} NAND gates.

	Gate delay (ps)		
Number of fanouts	Low V_{th}	High V_{th}	% increase
1	14.947	21.150	41.5
2	22.111	30.214	36.6
3	29.533	39.171	32.6
4	37.073	48.649	31.2
5	44.623	58.466	31.0

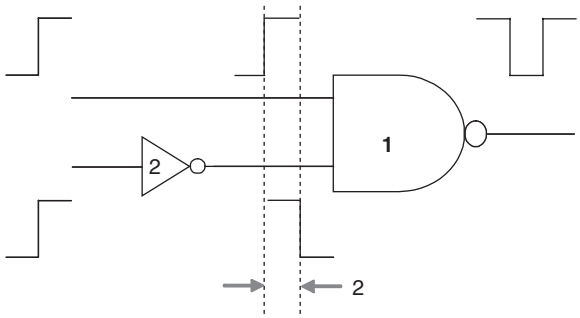


(a) one n-input NAND gate

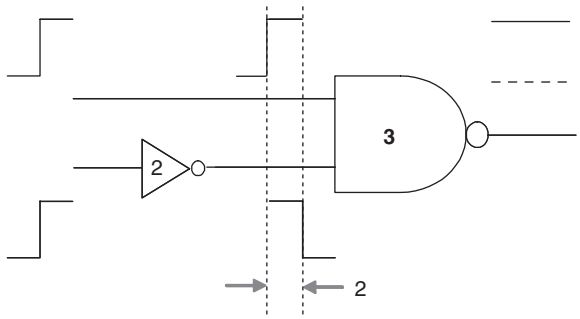


(b) Transition timing for the inputs and output of gate in (a)

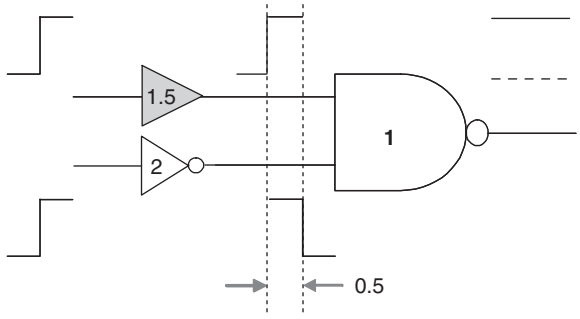
Fig. 1. Output timing window for an *n*-input NAND gate.



(a) Glitch at the output of one NAND gate



(b) Glitch elimination by hazard filtering



(c) Glitch elimination by path delay balancing

Fig. 2. Glitch elimination methods.

difference of the signal arrival times at every gate's inputs remains smaller than the inertial delay of the gate. This condition is expressed by the following inequality:

$$t_n - t_1 < d_i \tag{4}$$

where we assume t_1 is the earliest arrival time at inputs, t_n is the most delayed arrival time at another input, and d_i is gate's inertial delay, as illustrated in Figure 1. The interval $t_n - t_1$ is referred to as the gate output timing window.[14] To satisfy inequality,[4] we can either increase the inertial delay d_i (*hazard filtering*) or decrease the path delay difference $t_n - t_1$ (*path balancing*). Figures 2(b) and (c) illustrate these procedures for the gate of Figure 2(a). Hazard filtering, when used alone, can increase the overall input to output delay. Path balancing does not increase the delay but requires insertion of delay elements. A combination of the two procedures can give an optimum design.[3, 29]

3. AN MILP FOR POWER MINIMIZATION

We use a mixed integer linear programming (MILP) model to determine the optimal assignment of V_{th} while maintaining any given performance requirement on the overall circuit delay. To minimize the total leakage the MILP assigns low V_{th} to the largest possible number of gates while controlling the critical path delays. Unlike the heuristic algorithms,[8, 12, 20, 22, 23] the MILP gives us a globally optimal solution as discussed in Section 5.

To eliminate the glitch power, additional MILP constraints determine the positions and values of the delay elements to be inserted to balance path delays within the inertial delay of the incident gates. We can easily make a tradeoff between power reduction and performance degradation by changing the constraint for the maximum path delay in the MILP model.

3.1. Variables

Each gate is characterized by four variables:

X_i: assignment of low or high V_{th} to gate i is specified by an integer X_i which can only be 0 or 1. A value 1 means that gate i is assigned low V_{th}, and 0 means that gate i is assigned high V_{th}. Each gate has two possible values of delays, D_{Li} and D_{Hi}, corresponding to low and high thresholds, respectively.

T_i: latest time at which the output of gate i can produce an event after the occurrence of an input event at primary inputs of the circuit.

t_i: earliest time at which the output of gate i can produce an event after the occurrence of an input event at primary inputs of the circuit.

$\Delta d_{i,j}$: delay of a possible delay element that may be inserted at the input of gate i from gate j.

Thus, an n input gate is characterized by $n+5$ quantities, i.e., n input buffer delay variables, two inertial delay constants, one [0, 1] integer variable, and two output timing window variables.

3.2. Objective Function

The objective function for the MILP is minimization of the sum of all gate leakage currents I_{leaki} and the sum of all inserted delays:

$$\text{Min}\left\{\sum_i I_{leaki} + \sum_i \sum_j \Delta d_{i,j}\right\} = \text{Min}\left\{\sum_i [X_i I_{Li} + (1 - X_i) I_{Hi}] + \sum_i \sum_j \Delta d_{i,j}\right\} \quad (5)$$

For a static CMOS circuit, the leakage power is

$$P_{\text{leak}} = V_{\text{dd}} \sum_i I_{leaki} \quad (6)$$

If we know the leakage currents of all gates, the leakage power can be easily obtained. Therefore, the first term in the objective functions of this MILP minimizes the sum of all gate leakage currents, i.e.,

$$\text{Min} \sum_i (X_i \cdot I_{Li} + (1 - X_i) I_{Hi}) \quad (7)$$

I_{Li} and I_{Hi} are the leakage currents of gate i with low V_{th} and high V_{th}, respectively. Recognizing that the subthreshold current of a gate depends on its input state, we make a leakage current look-up table of I_{Li} and I_{Hi} for all gates i through simulation. These look-up tables are similar to Table I and are used for power estimation by logic simulation as discussed in Section 6. For the MILP, we need one set of I_{Li} and I_{Hi} for each gate and the average values from the look-up tables can be used.

Besides the leakage power, we also minimize the glitch power, simultaneously. We insert minimal delays to satisfy the glitch elimination conditions at all gates. This leads to the second term in the objective function:

$$\text{Min} \sum_i \sum_j \Delta d_{i,j} \quad (8)$$

When implementing these delay elements, we use transmission gates with only the gate leakage. The two terms in the objective function, $\sum_i I_{\text{leaki}}$ and $\sum_i \sum_j \Delta d_{i,j}$, have different units and numerically $\sum_i I_{\text{leaki}}$ is 50 to 1000 times larger than $\sum_i \sum_j \Delta d_{i,j}$ in our examples of benchmark circuits. Therefore, the objective function of Eq. (5) puts greater emphasis on leakage power, assuming it to be the dominant contributor to the total power. Experimental results show that an objective function $\text{Min}\{A \times \sum_i I_{\text{leaki}} + B \times \sum_i \sum_j \Delta d_{i,j}\}$ with $A \rightarrow$ large constant and $B = 1$ generates the same results as those by the objective function of Eq. (5) in which the terms are left unweighted. In general, suitable weight factors A and B can be used to make tradeoffs between leakage power reduction and glitch power elimination.

3.3. Constraints

Constraints are imposed on each gate i with respect to each of its fanin j, where j refers to the gate providing the fanin:

$$T_i \geq T_j + \Delta d_{i,j} + [X_i \cdot D_{\text{Li}} + (1 - X_i) D_{\text{Hi}}] \quad (9)$$

$$t_i \leq t_j + \Delta d_{i,j} + [X_i \cdot D_{\text{Li}} + (1 - X_i) D_{\text{Hi}}] \quad (10)$$

$$X_i \cdot D_{\text{Li}} + (1 - X_i) \cdot D_{\text{Hi}} \geq T_i - t_i \quad (11)$$

where D_{Hi} and D_{Li} are the delays of gate i with high V_{th} and low V_{th}, respectively. With the increase in fanouts, the delay of the gate increases proportionately. Therefore, a look-up table is constructed by simulation and specifies the delays for all gate types for varying fanout numbers. D_{Li} and D_{Hi} for gate i are obtained from the look-up table whose entries are indexed by the gate type and the number of fanouts. As discussed in Subsection 2.2, constraints (9–11) ensure that the inertial delay of gate i is always larger than the delay difference of its input paths. This would be done by inserting the minimal number of delay elements while maintaining the critical path delay constraints.

We explain constraints (9–11) using the circuit shown in Figure 3. Here the numbers on gates are gate indexes and not the delays. Red (bold) lines show critical paths and two grey shaded triangles are delay elements possibly inserted on the input paths of gate 2. Similar delay elements are placed on all primary inputs and fanout branches throughout the circuit. Let us assume that all primary input (PI) signals on the left arrive at the same time. For gate 2, one input is from gate 0 and the other input is directly from a PI. Its constraints corresponding to inequalities (9–11) are:

$$T_2 \geq T_0 + \Delta d_{2,0} + [X_2 \cdot D_{\text{L2}} + (1 - X_2) D_{\text{H2}}] \quad (12)$$

$$T_2 \geq 0 + \Delta d_{2,\text{PI}} + [X_2 \cdot D_{\text{L2}} + (1 - X_2) D_{\text{H2}}] \quad (13)$$

$$t_2 \leq t_0 + \Delta d_{2,0} + [X_2 \cdot D_{\text{L2}} + (1 - X_2) D_{\text{H2}}] \quad (14)$$

$$t_2 \leq 0 + \Delta d_{2,\text{PI}} + [X_2 \cdot D_{\text{L2}} + (1 - X_2) D_{\text{H2}}] \quad (15)$$

$$[X_2 \cdot D_{\text{L2}} + (1 - X_2) D_{\text{H2}}] \geq T_2 - t_2 \quad (16)$$

Variable T_2 that satisfies inequalities (12) and (13) is the latest time at which an event (signal change) could occur at the output of gate 2. Variable t_2 is the earliest time at which an event could occur at the output of gate 2, and it satisfies both inequalities (14) and (15). Constraint (16) means that the difference of T_2 and t_2, which equals the delay difference between two input paths, is smaller than

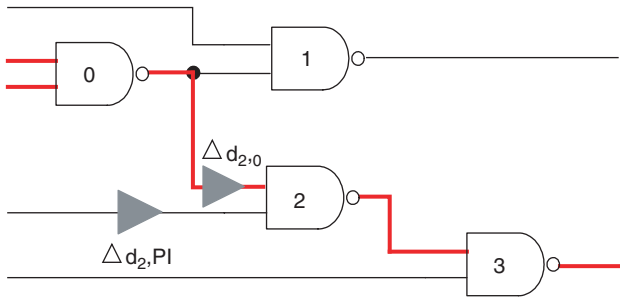


Fig. 3. Circuit for explaining ILP constraints.

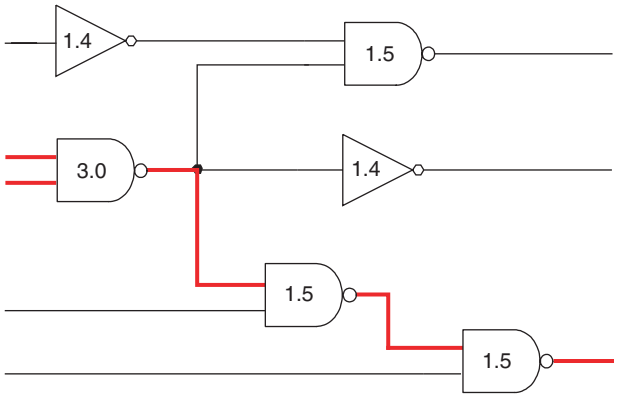


Fig. 4. An unoptimized circuit with high leakage and potential glitches.

gate 2's inertial delay, which may be either low V_{th} gate delay, D_{L2}, or high V_{th} gate delay, D_{H2}.

The critical path delay T_{max} is specified at primary output (PO) gates 1 and 3, as:

$$T_i \leq T_{max}, \quad i = 1, 3 \tag{17}$$

T_{max} can be the maximum delay specified by the circuit designer. Alternatively, the delay of the critical path (T_c) can be obtained from a linear program (LP) by assigning all gates to low V_{th}, i.e., $X_i = 1$ for all i. The objective function of this LP is minimization of the sum of T_k's where k refers to primary outputs. The critical path delay T_c is then the maximum of T_k's found by the LP.

If T_{max} equals to T_c, the actual objective function of the MILP model will be to minimize the total leakage current without affecting the circuit performance. By making T_{max} larger than T_c, we can further reduce leakage power with some performance compromise, and thus make a tradeoff between leakage power consumption and performance.

When we use this MILP model to simultaneously minimize leakage power with dual-V_{th} assignments and reduce dynamic power by balancing path delays with inserted delay elements, the optimized version for the circuit of Figure 4 is shown in Figure 5. In these figures the labels in or near gates are inertial delays.

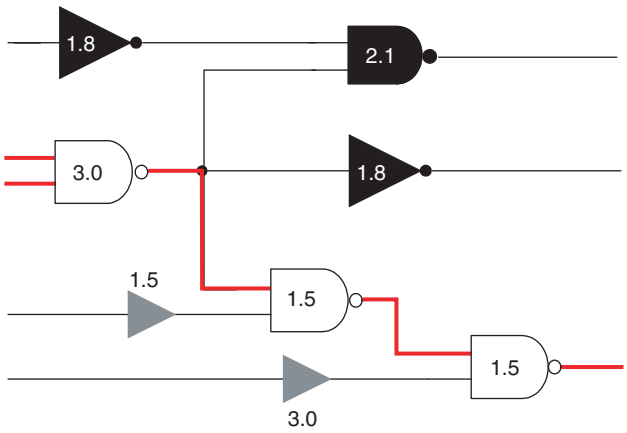


Fig. 5. Optimized glitch-free circuit with low leakage.

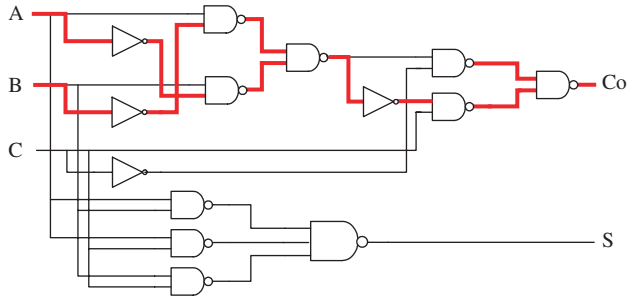


Fig. 6. A full adder circuit with all gates assigned low V_{th} (I_{leak} = 161 pA).

Three black shaded gates are assigned high V_{th}. They are not on critical paths (shown by red or bold lines) and their delay increase does not affect the critical path delay. Although delay elements were assumed to be present on all primary inputs and fanout branches, only two were assigned non-zero values. They are shown as grey triangles with delays of 1.5 and 3.0 units, respectively. To minimize the additional leakage and dynamic power consumed by these delay elements, we implement them by CMOS transmission gates. In Section 4, we will show that an always turned-on CMOS transmission gate can be treated as a zero-subthreshold leakage and low-dynamic-power-consumption delay element.[15, 16, 32]

A 14-gate full adder is used as a further illustration. Figure 6 is the original circuit with all low V_{th} gates. Critical paths are shown in red (or bold) lines. Figure 7 shows an MILP solution. All gates on non-critical paths were assigned high V_{th} (black shaded) to minimize leakage power. At the same time, three delay elements (grey shaded) are inserted to balance path delay to eliminate glitches. When the critical path delay is increased by 25%, the MILP gives the solution of Figure 8. Greater leakage power saving is achieved since some gates on the critical path are also assigned high V_{th}. All three circuits were implemented in the 70 nm BPTM CMOS technology[1] we mentioned in Section 2.1. The three delay elements use high-V_{th} devices and their design is described in the next section. The leakage average leakage currents for the circuits of Figures 6 (unoptimized), 7 (optimized with no critical path delay increase), and 8 (optimized with 25%

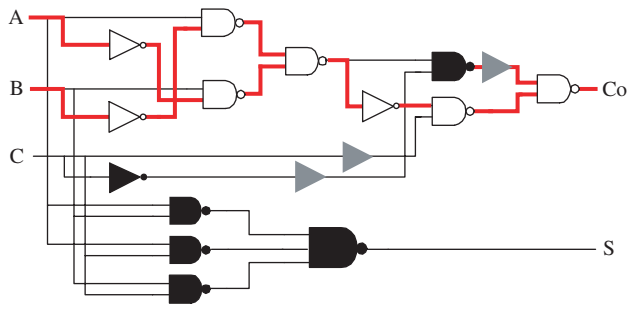


Fig. 7. Dual-V_{th} assignment and delay element insertion for $T_{max} = T_c$ (I_{leak} = 73 pA).

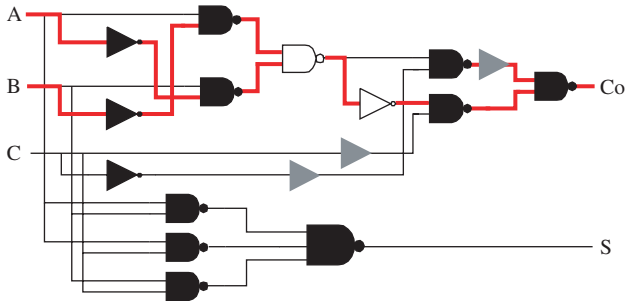


Fig. 8. Dual-V_{th} assignment and delay element insertion for $T_{max} = 1.25\ T_c$ ($I_{leak} = 16$ pA).

increase in critical path delay) were 161 pA, 73 pA, and 16 pA, respectively.

4. DELAY ELEMENT IMPLEMENTATION

In our design, all the delay elements are implemented by transmission gates, whose obvious advantage is that they consume very little dynamic power because they are not driven by any supply rails.[10] They also have lower area overhead and leakage power consumption compared with the more conventional two-cascaded-inverter buffer.[15, 16, 18, 19] CMOS transmission gates are adopted in our design to avoid the voltage drop when signal passes through series transistors. The circuits in Figure 9 simulated for the subthreshold current by Smart-Spice were used to compare the leakage power dissipation in the two delay elements. In Figure 9(a), there is only a gate leakage path and no subthreshold leakage. The two transistors are always turned on. In two cascaded inverters of Figure 9(b), beside gate leakage, subthreshold paths always exist. Hence, we can treat a transmission-gate delay element as a zero-subthreshold-leakage delay element. The delay of a transmission gate is given by:[10]

$$t_p = \ln(2) R_{eq} C_L \tag{18}$$

Where R_{eq} is the equivalent resistance of a CMOS transmission gate, and C_L is load capacitance. By changing the widths and lengths of the transistors, we can change the delay of the transmission gate. We simulated the circuit of Figure 9(a) for nearly 80 transmission gates with transistors whose dimensions were varied. By subtracting the delay of the circuit in which the transmission gate was replaced by a short, we obtained the delay of the transmission gate. These data were arranged in a look-up table of delays versus transmission gate dimensions. For any required delay between two entries in the look-up table, the size of the transmission gate is determined by interpolation.

The is a deterministic approach in which the initial delay of a gate is assumed to have a fixed value. However, variations of process parameters, especially in nanometer technologies, can change gate delays and affect the path delay balancing, causing incomplete suppression of

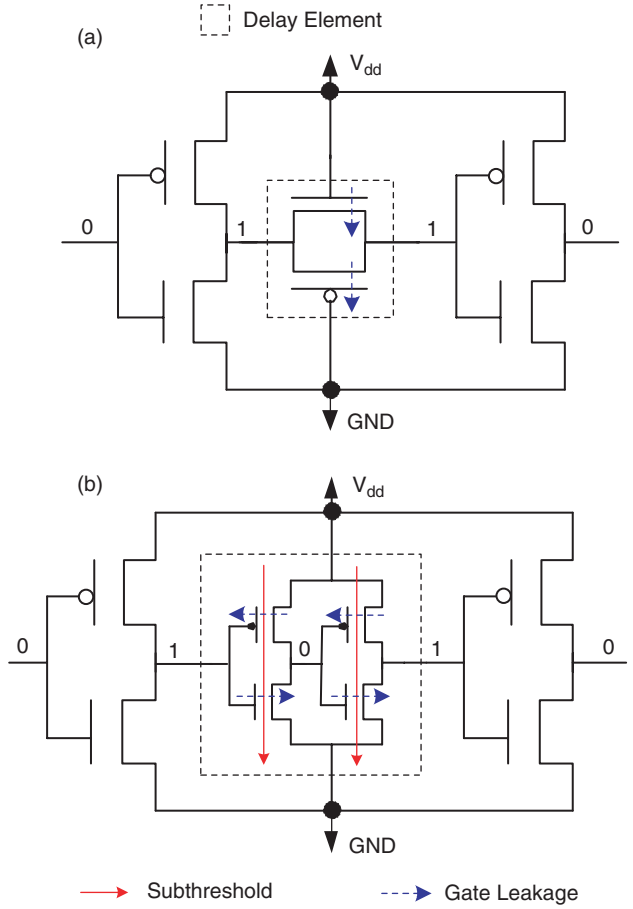


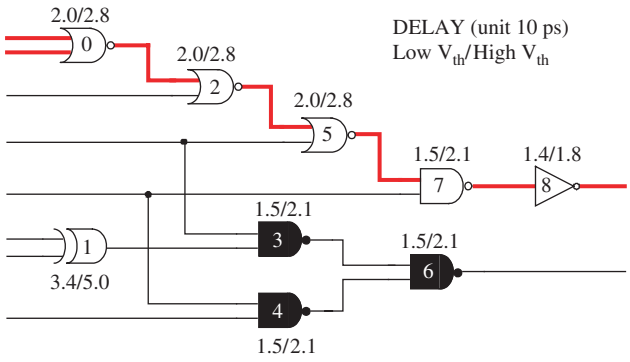
Fig. 9. Delay elements: (a) CMOS transmission gate and (b) cascaded inverters.

glitches. Hu[6, 30] proposes a statistical analysis to treat the gate delays as random variables with normal distributions. The results show that the power distribution due to the process variation can be reduced. Our deterministic MILP model can also be extended as a statistical MILP model to minimize the impact of the process variation on the glitch elimination and leakage power.[31]

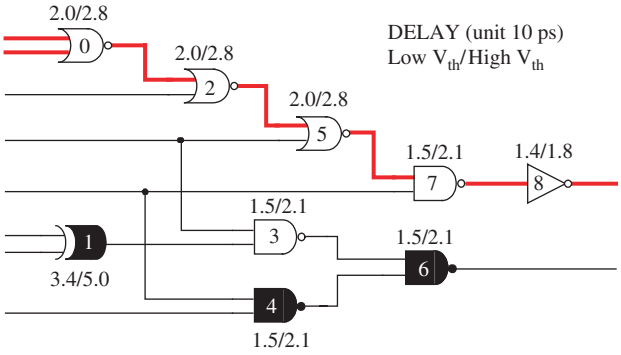
5. ILP AND HEURISTIC ALGORITHMS

In the introduction, we mentioned several heuristic algorithms[8, 12, 20, 22, 23] used for dual-V_{th} assignment. Due to the intrinsic limitation, heuristic algorithms normally aim at achieving a locally optimal solution. In MILP, the objective function and constraints are both linear, ensuring a global optimization. To illustrate the point, we examine the *backtrace algorithm*[22] as an example to show the advantage of the MILP.

In Figure 10, the XOR gate (gate 1) close to primary input has the largest leakage power reduction if assigned a high threshold. However, in Figure 10(a), the slacks for the non-critical paths are first consumed by gates 6, 3, and 4, which are closer to primary outputs. Hence, by the time the backtrace arrives at the XOR gate the slack



(a) Backtrace algorithm: optimized leakage current is 79.2 pA.



(b) MILP: optimized leakage current is 58.1 pA

Fig. 10. Comparison of MILP with backtrace heuristic algorithm.

has already been used up and it cannot be assigned high-V_{th}. In Figure 10(b), MILP considers leakage reduction and delay increase of each gate simultaneously, making sure that the best candidates (gates with the largest leakage reduction without violating the timing constrains) are selected. Due to the global optimization, MILP achieves 26% greater leakage power saving compared to the heuristic backtrace algorithms. Other heuristic algorithms have the similar problems, because the available slack for each gate must depend on the search direction or the selected cut[20] in the circuit graph. Thus, a global optimization cannot be guaranteed.

6. RESULTS

To study the increasingly dominant effect of leakage power, we use the BPTM 70 nm CMOS technology.[1] Low V_{th} for NMOS and PMOS devices are 0.20 V and −0.22 V, respectively. High V_{th} for NMOS and PMOS are 0.32 V and −0.34 V, respectively. We regenerated the netlists of ISCAS'85 benchmark circuits using a cell library in which the maximum gate fanin is 5. Two look-up tables for gate delays and leakage currents, respectively, of each type of cell were constructed using Spice simulation. A C program parses the netlist and generates the constraint set (see Section 3) for the CPLEX ILP solver in the AMPL software package.[13] CPLEX then give the optimal V_{th} assignment as well as the value and position of every delay element. The dynamic power is estimated by an event driven logic simulator that incorporates an inertial delay glitch filtering analysis.

6.1. Leakage Power Reduction

The results of the leakage power reduction for ISCAS'85 benchmark circuits are shown in Table III. Here the objective of the MILP was set to minimization of leakage alone. All $\Delta d_{i,j}$ variables were forced to be 0 and constraint 11 was suppressed. The numbers of gates in column 2 are for our gate library and differ from those in the original benchmark netlists. T_c in column 3 is the minimum delay of the critical path when all gates have low V_{th}. This was determined by the LP discussed in Subsection 3.3 in the paragraph following Eq. (17). Column 4 shows the total leakage current with all gates assigned low V_{th}. Column 5 shows the optimized circuit leakage current with gate V_{th} reassigned according to the MILP optimization. Column 6 shows the leakage reduction (%) for optimization without sacrificing any performance. Column 9 shows the leakage reduction with 25% performance sacrifice.

From Table III, we see that by V_{th} reassignment the leakage current of most benchmark circuits is reduced by more than 60% without any performance sacrifice (column 6). For several large benchmarks leakage is reduced by 90% due to a smaller percentage of gates being on

Table III. Leakage reduction alone due to dual-V_{th} assignment (27 °C).

				Optimized ($T_{max} = T_c$)			Optimized ($T_{max} = 1.25\ T_c$)		
Circuit name	Number of gates	T_c (ns)	Unoptimized I_{leak} (μA)	I_{leak} (μA)	Leakage reduction (%)	Sun OS5.7 CPU s	I_{leak} (μA)	Leakage reduction (%)	Sun OS5.7 CPU s
C432	160	0.751	2.620	1.022	61.0	0.42	0.132	95.0	0.3
C499	182	0.391	4.293	3.464	19.3	0.08	0.225	94.8	1.8
C880	328	0.672	4.406	0.524	88.1	0.24	0.153	96.5	0.3
C1355	214	0.403	4.388	3.290	25.0	0.1	0.294	93.3	2.1
C1908	319	0.573	6.023	2.023	66.4	59	0.204	96.6	1.3
C2670	362	1.263	5.925	0.659	90.4	0.38	0.125	97.9	0.16
C3540	1097	1.748	15.622	0.972	93.8	3.9	0.319	98.0	0.74
C5315	1165	1.589	19.332	2.505	87.1	140	0.395	98.0	0.71
C6288	1177	2.177	23.142	6.075	73.8	277	0.678	97.1	7.48
C7552	1046	1.915	22.043	0.872	96.0	1.1	0.445	98.0	0.58

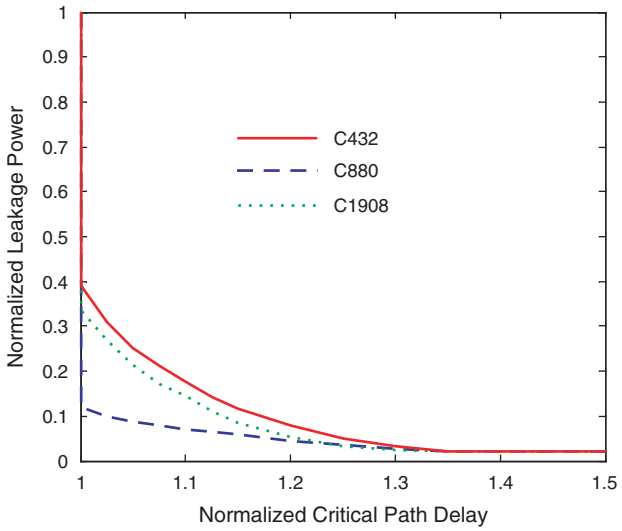


Fig. 11. Tradeoffs between leakage power and performance.

critical paths. However, for some highly symmetrical circuits, which have many critical paths, such as C499 and C1355, the leakage reduction is less. Column 9 shows that the leakage reduction reaches the highest level, around 98%, with some performance sacrifice.

The curves in Figure 11 show the relation between normalized leakage power and normalized critical path delay in a dual-V_{th} process. Unoptimized circuits with all low V_{th} gates are at point (1, 1) and have the largest leakage power and smallest delay. With optimal V_{th} assignment, leakage power can be reduced sharply by 60% (from point (1, 1) to point (1, 0.4)) to 90% (from point (1, 1) to point (1, 0.1)), depending on the circuit, without sacrificing any performance. When normalized T_{max} becomes greater than 1, i.e., we sacrifice some performance, leakage power further decreases with a slower decreasing trend. When the delay increase is more than 30%, the leakage reduction saturates at about 98%. Thus, Figure 11 provides a guide for making tradeoffs between leakage power and performance.

The CPU times shown in columns 7 and 10 of Table III are for the MILP. Although in the worst case, the solution time of MILP is exponential in the problem size, in practice, it is determined by the nature of the problem. From the data in columns 7 and 10 of Table III, it is really hard to express any relation between the CPU time and the problem size, such as the number of gates in the circuit. For example, MILP solution time for the 1046-gate C7552 is only 1.1 CPU seconds that is much less than 140 CPU seconds used for the 1165-gate C5315. Even for the same size problems, different constraints require varying solution times. Consider the 1177-gate C6288 circuit as an example. When the timing constraints for primary outputs (POs) are relaxed by 25%, CPU time decreases from 277 CPU seconds to 7.48 CPU seconds. As a result, MILP formulation may still solve some very large size circuits and provide a possibly better solution to dual-V_{th} assignment problem through global optimization.

6.2. Leakage, Dynamic Glitch, and Total Power Reduction

The leakage current increases with temperature because V_T (thermal voltage, kT/q) and V_{th} both depend on the temperature. Our Spice simulation shows that for a 2-input NAND gate with low V_{th}, when temperature increases from 27 °C to 90 °C, the leakage current increases by a factor of 10. For a 2-input NAND gate with high V_{th}, this factor is 20.

The leakage in our look-up table is from simulation for a 27 °C operation. To manifest the dominant effect of the leakage power, we estimate the leakage currents at 90 °C by multiplying the total leakage current obtained from CPLEX[13] by a factor between 10 and 20 as determined by the proportion of low to high threshold transistors.

The dynamic power is estimated by a glitch filtering event driven simulator, and is given by

$$P_{dyn} = \frac{E_{dyn}}{T} = \frac{0.5 \cdot C_{inv} \cdot V_{dd}^2 \cdot \sum_i T_i \mathrm{FO}_i}{1000(1.2 \cdot T_c)} \qquad (19)$$

where C_{inv} is the gate capacitance of an inverter, T_i is the number of transitions at the output of gate i when 1,000 random vectors are applied at PIs, and FO_i is the number of fanouts for gate i. Vector period is assumed to be 20% greater than the critical path delay, T_c. By simulating each

Table IV. Leakage, glitch, and total power reduction for ISCAS'85 benchmark circuits (90 °C).

		Leakage power (μW)			Glitch power (μW)			Total (leakage + glitch) power (μW)		
Circuit name	Number	All low V_{th}	Dual V_{th}	Reduc. (%)	Dual V_{th}	Delay Optim.	Reduc. (%)	All low V_{th}	Dual V_{th} + Del Opt.	Reduc. (%)
C432	160	35.77	11.87	66.8	101.0	73.3	27.4	136.8	85.2	37.7
C499	182	50.36	39.94	20.7	225.7	160.3	29.0	276.1	200.2	27.5
C880	328	85.21	11.05	87.0	177.3	128.0	27.8	262.5	139.1	47.0
C1355	214	54.12	39.96	26.3	293.3	165.7	43.5	347.4	205.7	40.8
C1908	319	92.17	29.69	67.8	254.9	197.7	22.4	347.1	227.4	34.5
C2670	362	115.4	11.32	90.2	128.6	100.8	21.6	244.0	112.1	54.1
C3540	1097	302.8	17.98	94.1	333.2	228.1	31.5	636.0	246.1	61.3
C5315	1165	421.1	49.79	88.2	465.5	304.3	34.6	886.6	354.1	60.1
C6288	1177	388.5	97.17	75.0	1691.2	405.6	76.0	2079.7	502.8	75.8
C7552	1046	444.4	18.75	95.8	380.9	227.8	40.2	825.3	246.6	70.1

Table V. Number of delay elements for optimization.

Circuit	Gates #	Δdi #
C432	160	160
C499	182	128
C880	328	303
C1355	214	112
C1908	319	313
C2670	362	330
C3540	1097	1258
C5315	1178	1198
C6288	1189	1307
C7552	1046	845

gate's number of transitions, we can estimate the glitch power reduction.

To demonstrate the projected dominant effect of leakage power in a sub-micron CMOS technology, we compare the leakage power and dynamic power at 90 °C in Table IV. "All low V_{th}" means the unoptimized circuit that has all low threshold gates, and "Dual V_{th}" means the optimized circuit whose V_{th} has been optimally assigned for minimum leakage. Column 6 gives the glitch power of the optimized design, which is further reduced as shown in column 7 when glitches are eliminated. We observe that for 70 nm BPTM CMOS technology at 90 °C, unoptimized leakage power (column 3) of some large ISCAS'85 benchmark circuits can account for about one half or more of the total power consumption (column 9). With V_{th} reassignment, the optimized leakage power of most benchmark circuits is reduced to less than 10%. With further glitch (dynamic) power reduction, total power reductions for all circuits are more than 50%. Some have a total reduction of up to 70%.

However, the area overhead due to the inserted delay elements is large. From Table V, we observe that the number of delay elements (Δdi #) is almost equal to the number of gates (Gates #), except for C1355. If we assume that the average number of transistors in a gate is 4 (e.g., consider a 2-input NAND gate), and each delay element implemented by a transmission gate has 2 transistors, the area overhead will be around 50% due to delay element insertion. The main reason is that our cell library has some complex gates, for example, AOI (AND-OR-INVERT) gates whose fanin number may be up to 5. Some NAND or NOR gates can also have up to 4 inputs. As a result, it is very possible that more than one delay buffer is inserted for a gate. The solution is to use a simpler and smaller cell library which will be used in our following research.

7. CONCLUSION

A new technique to reduce the leakage and dynamic glitch power simultaneously in a dual-V_{th} process is proposed in this paper. A mixed integer linear programming (MILP) model is generated from the circuit netlist and the AMPL CPLEX[13] solver determines the optimal V_{th} assignments for leakage power minimization and the delays and positions for inserting delay elements for glitch power reduction. Experimental results for ISCAS'85 benchmarks show reductions of 20%–96% in leakage, 28%–76% in dynamic (glitch), and 27%–76% in total power. We believe some of the other techniques, such as gate sizing and dual power supply can also be incorporated in the MILP formulation. Ongoing work incorporating process variation in this power reduction technique will be the topic of a future publication.[31] The transmission gate delay elements avoid the comparatively larger capacitive dissipation and subthreshold leakage inherent in the alternative design of two-inverter type of delay elements. However, the gate leakage of the transmission gate delay element could become a concern and will require further investigation.

Acknowledgments: The authors are thankful to reviewers for useful comments and for bringing to their notice similar work being done by other researchers.

References

1. BPTM: Berkeley Predictive Technology Model. http://www-device.eecs.berkeley.edu/~ptm/.
2. V. D. Agrawal, Low power design by hazard filtering. *Proc. 10th International Conference on VLSI Design* **(1997)**, pp. 193–197.
3. V. D. Agrawal, M. L. Bushnell, G. Parthasarathy, and R. Ramadoss, Digital circuit design for minimum transient energy and a linear programming method. *Proc. 12th International Conference on VLSI Design* **(1999)**, pp. 434–439.
4. A. P. Chandrakasan and R. W. Brodersen, Low Power Digital CMOS Design, Kluwer Academic Publishers, Boston **(1995)**.
5. F. Gao and J. P. Hayes, Total power reduction in CMOS circuits via gate sizing and multiple threshold voltages. *Proc. Design Automation Conference* **(2005)**, pp. 31–36.
6. F. Hu, Process-variation-resistant dynamic power optimization for VLSI circuits, Ph.D. thesis, Auburn University, Auburn, Alabama **(2006)**.
7. J. T. Kao and A. P. Chandrakasan, Dual-threshold voltage techniques for low-power digital circuits. *IEEE J. Solid-State Circuits* **(2000)**, Vol. 35, pp. 1009–1018.
8. M. Ketkar and S. S. Sapatnekar, Standby power optimization via transistor sizing and dual threshold voltage assignment. *Proc. International Conference on Computer-Aided Design* **(2002)**, pp. 375–378.
9. Y. Lu and V. D. Agrawal, Leakage and dynamic glitch power minimization using integer linear programming for V_{th} assignment and path balancing. *Proc. the International Workshop on Power and Timing Modeling, Optimization and Simulation* **(2005)**, pp. 217–226.
10. N. R. Mahapatra, S. V. Garimella, and A. Tarbeen, An empirical and analytical comparison of delay elements and a new delay element design. *Proc. IEEE Computer Society Workshop on VLSI* **(2000)**, pp. 81–86.
11. D. Nguyen, A. Davare, M. Orshansky, D. Chinney, B. Thompson, and K. Keutzer, Minimization of dynamic and static power through joint assignment of threshold voltages and sizing optimization. *Proc. the International Symposium on Low Power Electronics and Design* **(2003)**, pp. 158–163.
12. P. Pant, R. K. Roy, and A. Chatterjee, Dual-threshold voltage assignment with transistor sizing for low power CMOS circuits. *IEEE Transactions on VLSI Systems* **(2001)**, Vol. 9, pp. 390–394.

13. R. Fourer, D. M. Gay, and B. W. Kernighan, AMPL: A Modeling Language for Mathematical Programming, The Scientific Press, South San Francisco, California **(1993)**.

14. T. Raja, V. D. Agrawal, and M. L. Bushnell, Minimum dynamic power CMOS circuit design by a reduced constraint set linear program. *Proc. 16th International Conference on VLSI Design* **(2003)**, pp. 527–532.

15. T. Raja, V. D. Agrawal, and M. L. Bushnell, Design of variable input delay gates for low dynamic power circuits. *Proc. the International Workshop on Power and Timing Modeling, Optimization and Simulation* **(2005)**, pp. 436–445.

16. T. Raja, V. D. Agrawal, and M. L. Bushnell, Transistor sizing of logic gates to maximize input delay variability. *J. Low Power Electronics* **(2006)** Vol. 2, pp. 121–128.

17. T. Sakurai and A. R. Newton, Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas. *IEEE J. Solid-State Circuits* **(1990)**, Vol. 25, pp. 584–594.

18. S. Uppalapati, Low power design of standard cell digital VLSI circuits, Master's thesis, Rutgers University, New Brunswick, New Jersey **(2004)**.

19. S. Uppalapati, M. L. Bushnell, and V. D. Agrawal, Glitch-free design of low power ASICs using customized resistive feedthrough cells. *Proc. 9th VLSI Design and Test Symposium*, Bangalore **(2005)**, pp. 41–48.

20. Q. Wang and S. B. K. Vrudhula, Static power optimization of deep submicron CMOS circuits for dual V_T technology. *Proc. International Conference on Computer-Aided Design* **(1998)**, pp. 490–496.

21. L. Wei, Z. Chen, M. Johnson, and K. Roy, Design and optimization of low voltage high performance dual threshold CMOS circuits. *Proc. Design Automation Conference* **(1998)**, pp. 489–494.

22. L. Wei, Z. Chen, K. Roy, M. C. Johnson, Y. Ye, and V. K. De, Design and optimization of dual-threshold circuits for low-voltage low-power applications. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* **(1999)**, Vol. 7, pp. 16–24.

23. L. Wei, Z. Chen, K. Roy, Y. Ye, and V. De, Mixed-V_{th} (MVT) CMOS circuit design methodology for low power applications. *Proc. Design Automation Conference* **(1999)**, pp. 430–435.

24. L. Wei, K. Roy, and V. K. De, Low voltage low power CMOS design techniques for deep submicron ICs. *Proc. 13th International Conf. VLSI Design* **(2000)**, pp. 24–29.

25. M. Hashimoto, H. Onodera, and K. Tamaru, A power optimization method considering glitch reduction by gate sizing. *Proc. the International Symposium on Low Power Electronics and Design* **(1998)**, pp. 221–226.

26. E. Jacobs and M. Berkelaar, Using gate sizing to reduce glitch power. *Proc. of the PRORISC/IEEE Workshop on Circuits, Systems and Signal Processing* **(1996)**, pp. 183–188.

27. C. V. Schimpfle, A. Wroblewski, and J. A. Nossek, Transistor sizing for switching activity reduction in digital circuits. *Proc. European Conference on Theory and Design* **(1999)**.

28. A. Wroblewski, C. V. Schimpfle, and J. A. Nossek, Automated transistor sizing algorithm for minimizing spurious switching activities in CMOS circuits. *Proc. the International Symposium on Circuits and Systems* **(2000)**, pp. 291–294.

29. S. Kim, J. Kim, and S.-Y. Hwang, New path balancing algorithm for glitch power reduction. *IEE Proceedings G—Circuits, Devices and Systems* **(2001)**, Vol. 148, pp. 151–156.

30. F. Hu and V. D. Agrawal, Input-specific dynamic power optimization for VLSI circuits. *Proc. the International Symposium on Low Power Electronics and Design* **(2006)**.

31. Y. Lu and V. D. Agrawal, Statistical leakage and timing optimization for submicron process variation. *Proc. 20th International Conference on VLSI Design* **(2007)**.

32. T. Raja, V. D. Agrawal, and M. L. Bushnell, Variable input delay CMOS logic for low power design. *Proc. 18th International Conference on VLSI Design* **(2005)**, pp. 596–604.

Yuanlin Lu

Yuanlin Lu *is currently a Ph.D. student in the Electrical and Computer Engineering Department at Auburn University, Auburn, Alabama, USA, under the guidance of Professor Vishwani D. Agrawal. Before joining Auburn University, she received her B.S. and M.S. degrees in Engineering from Southeast University, China in 1999 and 2002, respectively. Her research interests include low-power design, VLSI design, and CAD.*

Vishwani D. Agrawal

Vishwani D. Agrawal *is the James J. Danaher Professor of Electrical and Computer Engineering at Auburn University, Alabama. He has over thirty years of industry and university experience, working at Bell Labs, Murray Hill, NJ; Rutgers University, New Brunswick, NJ; TRW, Redondo Beach, CA; IIT, Delhi, India; EG&G, Albuquerque, NM; and ATI, Champaign, IL. His areas of expertise include VLSI testing, low-power design, and microwave antennas. He obtained his B.E. degree from the University of Roorkee, Roorkee, India, in 1964; M.E. degree from the Indian Institute of Science, Bangalore, India, in 1966; and Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign, in 1971. He has published over 250 papers, has coauthored five books, and holds thirteen United States patents. His textbook, Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits (Springer), co-authored with M. L. Bushnell, was published in 2000. He is the founder and Editor-in-Chief (since 1990) of the Journal of Electronic Testing: Theory and Applications, and a past Editor-in-Chief (1985–87) of the IEEE Design & Test of Computers magazine. He is the founder and Consulting Editor of the Frontiers in Electronic Testing Book Series of Springer. He is a co-founder of the International Conference on VLSI Design, and the International Symposium on VLSI Design and Test (VDAT), held annually in India. He has served on numerous conference committees and is a frequently invited speaker. He was the invited Plenary Speaker at the 1998 International Test Conference, Washington, D.C., and the Keynote Speaker at the Ninth Asian Test Symposium, held in Taiwan in 2000. He served on the Board of Governors (1989 and 1990) of the IEEE Computer Society, and in 1994 chaired the Fellow Selection Committee of that Society. He has received eight Best Paper Awards and one Honorable Mention Paper Award. He received the Lifetime Achievement Award from the VLSI Society of India in 2006, the Harry H. Goode Memorial Award of the IEEE Computer Society for "innovative contributions to the field of electronic testing" in 1998, and the Distinguished Alumnus Award of the University of Illinois at Urbana-Champaign, "in recognition of his outstanding contributions in design and test of VLSI systems" in 1993. Dr. Agrawal is a Fellow of the ACM, IEEE, and IETE-India. He has served on the Advisory Boards of the ECE Departments of the University of Illinois, New Jersey Institute of Technology, and the City College of the City University of New York.*