

A LEAKAGE ESTIMATION AND REDUCTION TECHNIQUE FOR SCALED CMOS LOGIC CIRCUITS CONSIDERING GATE-LEAKAGE

Hafijur Rahman, and Chaitali Chakrabarti

Center for Low Power Electronics
Department of Electrical Engineering
Arizona State University
hafij@asu.edu, chaitali@asu.edu

ABSTRACT

Leakage power reduction is extremely important in the design of scaled CMOS logic circuits. The dominant leakage mechanism of such circuits is the gate leakage. This paper first describes a fast leakage estimation technique based on biasing states for both gate leakage and sub-threshold leakage. Next, it describes a leakage reduction method based on the selective insertion of control points. Simulations on a set of examples show that this method results in the average leakage being 28.7% of the leakage of the baseline circuit whose inputs have already been subjected to the minimum leakage vector (MLV).

1. INTRODUCTION

Technology scaling enables us to integrate huge number of transistors on chip for higher performance. This comes at the price of increase in both static and dynamic power consumption. For scaled technologies, leakage power reduction is an essential design component. There are different mechanisms that contribute to leakage power. These include sub-threshold leakage, gate leakage, BTBT leakage, etc. [1]. Gate leakage is clearly the dominant component if the gate oxide thickness is less than 20 Å. So, leakage reduction for scaled technologies should focus more on gate leakage reduction.

This paper describes techniques to estimate and reduce the leakage power (including gate leakage) in CMOS circuits. The main contributions are as follows:

1) The proposed leakage estimation method includes forward and backward gate leakage for both PMOS and NMOS devices, along with the sub-threshold leakage. Average error using this estimation method is as low as 1.8% with respect to SPICE results.

2) The proposed leakage reduction method is based on selective insertion of control points. This method achieves 71.3% average leakage reduction with 31.9% average increase in delay even if the minimum leakage vector (MLV) is applied to the primary inputs of the circuit.

The rest of the paper is organized as follows: Section 2 describes the various leakage mechanisms. Section 3 presents the proposed leakage power estimation technique. Section 4 introduces the control point insertion based technique for leakage power reduction. Section 5 concludes the paper.

2. LEAKAGE MECHANISM REVIEW

The two major leakage components in scaled devices are - 1) sub-threshold leakage and 2) gate leakage. The threshold voltage scaling in short channel devices results in exponential increase in sub-threshold current [1]. The sub-threshold current mainly depends on drain to source voltage, the gate to source voltage, and threshold voltage. The gate-oxide thickness scaling, required to reduce the short channel effect, causes an increase in gate leakage due to direct tunneling through the gate oxide. Gate leakage current becomes a serious concern when the gate oxide thickness is scaled down to 20 Å and below.

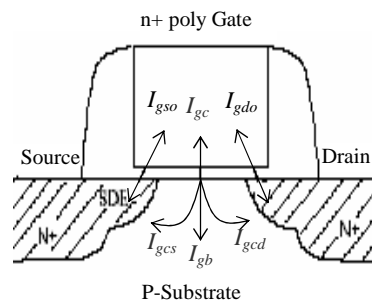


Figure 1: Illustration of gate tunneling components of a short channel NMOS device

The major components of gate tunneling in a scaled MOSFET device are shown in Figure 1 [2]: (1) gate to source /drain overlap region current (edge direct tunneling (EDT)) components (I_{gso} & I_{gdo}), (2) gate to channel current (I_{gc}), part of which goes to source (I_{gcs}) and rest goes to drain (I_{gcd}). Each component is dominant or important in different regions of operations of MOSFET. In gate leakage analysis, the forward and backward gate leakage is expressed as,

$$I_{fgate} = I_{gdo} + I_{gso} + I_{gcd} + I_{gcs} \quad \dots(1)$$

$$I_{bgate} \approx I_{dgo} \quad \dots(2)$$

There are two other leakage mechanisms. Gate induced drain leakage (GIDL) [3] arises due to minority carrier emission to the 'incipient inversion layer'. Reverse biased drain to substrate and source to substrate junction Band-to-Band Tunneling (BTBT) leakage current [1] depends on substrate bias. These two leakage mechanisms are not taken in our analysis due to their small contribution to overall leakage.

3. LEAKAGE ESTIMATION

Leakage estimation for a large circuit is complicated by the state dependence of both the gate tunneling current and sub-threshold current. A leakage estimation method has been proposed in [4] that considers forward gate leakage but ignores gate current for PMOS transistor as well as the backward gate current. Rao et.al. in [5] have proposed a gate leakage estimation method based on six different biasing states of MOS devices. The method considers both forward and backward gate leakage. Our work is an extension of [5]: we consider four possible biasing states, ($S_1 - S_4$) instead of the six biasing states to estimate both gate and sub-threshold leakage. Figure 2 shows the four states for NMOS devices. The other two states in [5] $\langle 0 \ 0 \ V_{dd} \rangle$ and $\langle V_{dd} \ 0 \ V_{dd} \rangle$ are applicable to only PMOS devices and are identical to states S_3 and S_4 for NMOS devices. The biasing states of each transistor in a circuit can be determined by switch level simulation.

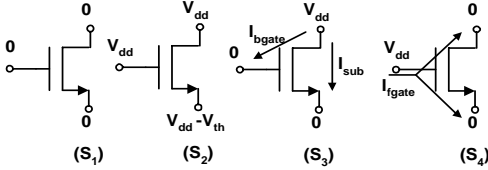


Figure 2: The four possible bias conditions for a NMOS transistor in a circuit

States S_1 and S_2 result in no gate leakage current and negligible sub-threshold current.

State S_3 results in both sub-threshold current and gate tunneling current. For this biasing state, the gate leakage current is the backward gate current that depends upon the position of the transistor in the stack. If this transistor is on the top of the stack (as in Figure 3(a)), V_{dg} equals to V_{dd} and the backward gate leakage current becomes maximum. If this transistor is placed other than on the top of the stack (as in Figure 3(b)), V_{dg} equals to $V_{dd} - V_{th}$, and the backward gate leakage current will be lower than the previous case. The gate leakage current for this biasing state can be expressed as,

$$I_{bgate} = I_{bgateT} \times S_p \times S_s \quad \dots(3)$$

where I_{bgateT} is the backward gate leakage current for $V_{dg}=V_{dd}$, S_p is the position factor which depends upon the position of the devices in a stack and S_s is the size of transistor.

For state S_3 , the sub-threshold current can be approximated by,

$$I_{sub} = I_{subS} \times S_f \times S_s \quad \dots(4)$$

where, I_{subS} is the sub-threshold leakage current for a single OFF transistor in a stack. S_f is the stack factor, which represents the number of OFF transistor in a stack.

State S_4 results in only the forward gate current. This biasing state contributes to gate leakage for all possible positions in a stack except the case in which there is at least one non-conducting transistor both above and below this transistor in the stack as shown in Figure 3(c). In this case (Figure 3(c)), the internal node V_a and V_b are no longer near ground potential. There is significant gate leakage current at the gate terminal, but its contribution is neglected since there is no path the drain (or source) to ground terminal and gate leakage for this case can be

ignored. The forward gate leakage current for state S_4 can be expressed as,

$$I_{fgate} = I_{fgateb} \times S_p \times S_s \quad \dots(5)$$

where I_{fgateb} is the forward gate current, S_p is the position factor. S_p equals to 0 for the case shown in Figure 3(c) and in other cases, it is 1.

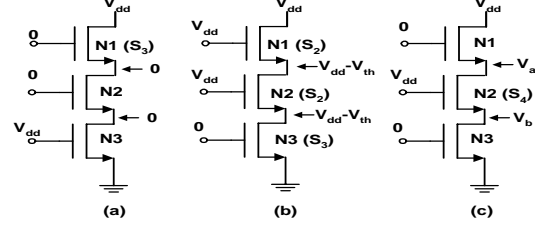


Figure 3: (a) S_3 on the top of the stack (b) S_3 in the bottom of the stack (c) S_4 in between two non-conducting transistors

The total leakage current of circuit can be found by adding the leakage components for all bias states of a circuit, i.e. $I_L = \sum_i I_g(S_i)$.

To demonstrate the accuracy of this method, the sub-threshold and gate leakage estimate obtained by this method for all possible states of a sample three input NAND gate with feature size of $65nm$ and 17\AA gate oxide are tabulated in Table 1 and also compared with SPICE simulation results. The input pattern of this gate determines the different bias states of its transistors. To determine the gate leakage and sub-threshold leakage of individual PMOS and NMOS devices, SPICE simulation was done using Berkeley Predictive SPICE model (BPSIM4) for $65nm$ technology that includes the parameters for gate tunneling. The error ranges for 0-7.3% with an average error of 1.81%. Leakage for a three input NAND gate with feature size of $45nm$ and 14\AA gate oxide resulted in an error in the range of -0.76% - 6.83%, with an average error of 1.89%. In both these cases, large errors occur in states '010' and '100'. These large errors occur due to the assumption that all the internal nodes attain full logic values (either V_{dd} or 0). This is fairly good assumption for all cases except the case shown in Figure 3(c).

Table 1: Estimated total leakage for 3 input NAND gate with 17\AA gate oxide

Input State	Estimated Leakage (nA)			SPICE (nA)	%error
	I_{gate}	I_{sub}	I_{total}		
000	5.78	0.47	6.25	6.257	0.11%
001	19.67	0.78	20.45	20.46	0.05%
010	5.69	0.78	6.47	6.98	7.30%
011	33.54	2.50	36.04	36.156	0.32%
100	2.44	0.78	3.22	3.45	6.67%
101	16.22	2.50	18.72	18.73	0.05%
110	1.69	2.50	4.19	4.19	0.00%
111	43.572	10.8	54.372	54.38	0.01%

4. LEAKAGE CONTROL

Due to the transistor stacking effect, the leakage of a circuit depends upon its input combination [6]. Thus the leakage power consumption during the standby mode can be significantly

reduced if the inputs to the circuit are chosen carefully. The circuit inputs that result in minimum leakage is referred to in the literature as the minimum leakage vector (MLV). There are several methods for finding MLV. These include branch and bound based techniques, a greedy heuristic based on the calculation of ‘Leakage Observability’ [6] and an approach that finds the input pattern that maximizes the number of the ‘OFF’ transistor in all stacks across the circuit [7].

For large circuits, the variance in the leakage energy for different input combinations is not very large. Application of the MLV thus does not result in significant reduction in leakage energy. Greater reduction can be achieved if the state of the gates deep in the circuit can be manipulated. One way of achieving it is by control point insertion, a method first proposed in [9]. Our approach consists of first finding the MLV, and then adding control points to a selective number of gates.

4.1. Control Point Insertion Method

In order to insert control points to the circuit, gates are selected in proper sequence so that the gates that contribute greater leakage are selected first. The drawback of this approach is the additional area and delay penalty. Thus a balance has to be achieved between the leakage energy reduction and the area and delay increase.

4.1.1. Gate Level Strategy

While the sub-threshold current depends on number of ‘OFF’ transistor in a stack, the gate leakage current depends on the position of ‘OFF’ transistors in a stack. Since in CMOS logic circuits, the leakage current for NMOS devices are much greater than those of PMOS devices, an effective solution of leakage reduction is to add an NMOS transistor as a control transistor at the bottom of the NMOS transistor stack (near the ground). If the NMOS control transistor is off, for some input states both the PMOS and NMOS stacks are turned OFF. In that case, the output voltage of the gate is near $V_{dd}/2$ (depending upon the size of the PMOS and NMOS transistors). This output value causes a short circuit in the next gate level. For this reason, an extra PMOS transistor is added in parallel to the PMOS transistor stacks to force the output to V_{dd} when the control input is set at ‘0’. Figure 4 shows an example where a NMOS control transistor is added at the bottom of NMOS transistor stack and a PMOS transistor is added in parallel to the PMOS stack in a 2-input NAND gate. In standby mode, this NMOS control transistor increases the number of ‘OFF’ transistor in the stack and effectively reduces both the gate and sub-threshold leakage.

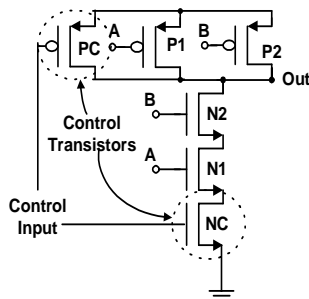


Figure 4: Illustration of control point insertion in a 2-input NAND gate

4.1.2. Circuit Level Strategy

Control points cannot be added to all the gates in the circuit because of the increased area and delay penalty. A careful selection of gates is necessary such that the leakage reduction is maximized subject to area and delay constraints. Our algorithm chooses the gates based on how effective they are in reducing the leakage energy.

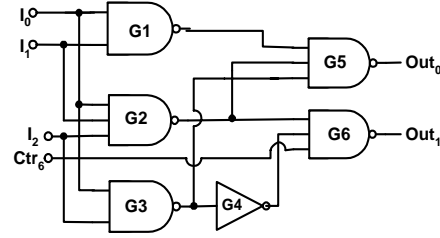


Figure 5: An example circuit to demonstrate the calculation of ‘Leakage Sensitivity’

We define ‘Leakage Sensitivity’ of a gate i (LS_i) as the amount of leakage reduction due to the insertion of a control point to gate i . So, ‘Leakage Sensitivity’ of a gate i , $LS_i = \sum \text{leakage before control point insertion} - \sum \text{leakage after control point insertion}$. To illustrate this concept, we use the example logic circuit shown in Figure 5. Suppose, MLV for this circuit is $\{I_2 I_1 I_0\} = \{110\}$. The leakage currents of six gates are-

$$L_{\text{gate1}} = 4.77 \text{ nA}, L_{\text{gate2}} = 4.52 \text{ nA}, L_{\text{gate3}} = 4.77 \text{ nA}, \\ L_{\text{gate4}} = 20.73 \text{ nA}, L_{\text{gate5}} = 62.15 \text{ nA}, L_{\text{gate6}} = 22.76 \text{ nA}$$

If we add a control input to gate 4, the leakage of the gates 1, 2, 3 and 5 will remain the same and leakage of gates 4 and 6 will change. L_{gate4} and L_{gate6} are now 2.63 nA and 37.18 nA. The leakage sensitivity of gate 4, $LS_4 = (4.77 + 4.52 + 4.77 + 20.73 + 62.15 + 22.76) - (4.77 + 4.52 + 4.77 + 20.73 + 62.15 + 37.18) = 3.68 \text{ nA}$.

The proposed algorithm for choosing the candidate gates in a brute force manner is described below. The algorithm is iterative: in each iteration, the gate with highest leakage sensitivity is chosen. The number of iterations depends on the desired leakage energy reduction.

- 1) Compute MLV
- 2) Compute leakage sensitivity for all gates
- 3) Until the required leakage performance is satisfied, do
 - a) Add a control point to the gate which has the highest leakage sensitivity
 - b) Re-compute leakage sensitivity for rest of the gates

This algorithm has been applied to the example circuit with devices having 17\AA gate oxide (shown in Figure 5). In the first iteration, gate 5 is chosen since it has the highest leakage sensitivity. In the second iteration, gate 6 is chosen and so on. The leakage reduction, area and delay penalty as more and more control points are added is shown in Figure 6. All values are normalized with respect to the circuit with MLV in the inputs. We see that as the number of control points increases, the leakage energy reduces while the area and delay increases. The leakage energy drops to 19% after addition of three control points and then reduces mildly as more control points are added. Thus adding more than three control points to this circuit is not advisable.

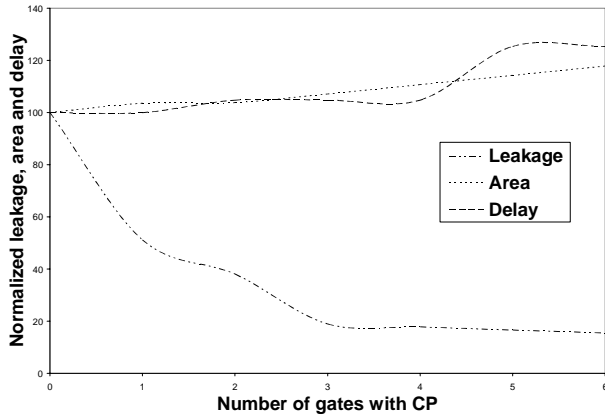


Figure 6: Normalized leakage, area and delay vs number of gates with control point for the example circuit in Figure 5

In order to make the greedy search more efficient, leakage sensitivity is calculated for a select group of gates. These include gates that have all the inputs in logic high state, gates that are at lower level and larger gates that have more number of transistors. This is currently under study.

4.2. Results

The proposed leakage reduction methods were implemented and tested for ISCAS85 benchmark circuits. Each of the benchmark circuits was optimized and synthesized using Berkeley SIS tools and mapped into technology library of feature size $65nm$ and 17\AA gate oxide thickness. The synthesized netlists were simulated for leakage for different input state with and without inserting control points.

Table 2 shows the leakage current after adding control points to the gates using the brute force method described in 4.1.2. All values are normalized with respect to the baseline circuit with MLV applied to the inputs. Addition of control points to 25% of the gates results in an average leakage of 70.3%. If control points are added to all gates of the circuit, the average leakage is only 28.7%. Thus more the number of control points, greater is the leakage reduction. In reality, the number of control points will be determined by the delay and area tolerance level.

Table 2: Leakage reduction for several benchmark circuits

ISCAS85 Bench. Circuits	No. of gate	Leakage (nA) (% of gates with CP)				
		MLV	10%	25%	50%	100%
C432	161	3063	2653	2230	1733	982
C499	518	9617	8039	6530	5451	2569
C880	372	6530	5775	4873	3620	2112
C1355	518	9617	8039	6530	5451	2569
C1908	484	8881	7639	6438	5083	2327
C2670	720	13005	10999	9233	7033	4081
C3540	1005	20258	16082	13314	10186	5210
Average	-	100%	84.5%	70.3%	55.2%	28.7%

Table 3 shows the area and delay penalty due to the addition of control points to the gates. Delay calculation is based

on the propagation delay associated with the longest path and it is measured by extensive HSPICE simulation. Since the standard cell for feature size of $65nm$ gate length is not available yet, area is calculated using the dimensions of PMOS and NMOS devices that are used for the design and analysis. The delay and area after control point insertion is normalized with respect to the baseline circuits. The average delay overhead when control points are inserted in 25% (100%) of the gates is 10.85% (31.9%). The average area overhead when control points are inserted in 25% (100%) of the gates is 6.73% (26.83%). Thus the delay and the area overhead cannot be ignored and the number of control points that could be added would depend on the delay and area tolerance levels.

Table 3: Area and delay penalty for control point insertion

ISCAS85 Bench.	25% gates with CP		100% gates with CP	
	Delay (%)	Area (%)	Delay (%)	Area (%)
C432	105.1	106.3	129.1	125.5
C499	107.6	107.1	138.2	128.5
C880	120.4	106.6	125.5	126.3
C1908	110.3	106.9	135.1	127.0
Average	110.85	106.73	131.9	126.83

5. CONCLUSION

In this paper, we first propose a fast approach of estimating the total leakage current including both gate leakage current and sub-threshold current. This method is fairly accurate compared to SPICE. Next we propose an effective leakage control method based on selective insertion of control points. Results show that this method results in 71.3% leakage reduction with minimal increase in area and delay on circuits with MLV as their inputs.

6. REFERENCES

- [1] S. Mukhopadhyay, A. Raychowdhury, and K. Roy, 'Accurate Estimation of Total Leakage Current in Scaled CMOS Logic Circuits Based on Compact Current Modeling,' *In Proc. DAC*, pp. 169-174, June 2003.
- [2] K. Cao, et.al. 'BSIM4 Gate Leakage Model Including Source Drain Partition,' *IEDM*, pp. 815-818, 2000.
- [3] T. Y. Chen, J. Chen, P. K. Ko, C. Hu, 'The Impact of Gate-induced Drain Leakage Current on MOSFET Scaling,' *Tech. Digest of IEDM*, pp. 718-721, 1987.
- [4] D. Lee, W. Kwong, D. Blaauw, and D. Sylvester, 'Analysis and Minimization Techniques for Total Leakage Considering Gate Oxide Leakage,' *In Proc. DAC*, pp. 175-180, June 2003.
- [5] R.M. Rao, J.L. Burns, A. Devgan and R.B. Brown, 'Efficient Techniques for Gate Leakage Estimation,' *In Proc. ISLPED*, pp. 100-103, August 2003.
- [6] M. Johnson, D. Somasekhar, and K. Roy, 'Models and Algorithms for Bounds on Leakage in CMOS Circuits,' *IEEE Trans. on CAD of Integrated Circuits And Systems*, Vol 18, No. 6, pp. 714-725, June 1999.
- [7] Y. Ye, S. Borker, and V. De, 'A New Technique for Standby Leakage Reduction in High-Performance Circuits,' *Symposium on VLSI Circuits*, pp. 40-41.9, 1998.
- [8] A. Abdollahi, F. Fallah, and M. Pedram, 'Runtime Mechanisms for Leakage Current Reduction in CMOS VLSI Circuits,' *In Proc. ISLPED*, pp. 213-218, Aug. 2002.