Leakage and Dynamic Glitch Power Minimization Using Integer Linear Programming for V_{th} Assignment and Path Balancing

Yuanlin Lu and Vishwani D. Agrawal

Auburn University, Department of ECE, Auburn, AL 36849, USA
luyuanl@auburn.edu, vagrawal@eng.auburn.edu

Abstract. This paper presents a novel technique, which uses integer linear programming (ILP) to minimize the leakage power in a dual-threshold static CMOS circuit by optimally placing high-threshold devices and simultaneously reduces the glitch power using the smallest number of delay elements to balance path delays. The constraint set size for the ILP model is linear in the circuit size. Experimental results show 96%, 40% and 70% reduction of leakage, dynamic and total power, respectively, for the benchmark circuit C7552 implemented in the 70nm BPTM CMOS technology.

1 Introduction

In the past, the dynamic power has dominated total power dissipation of a CMOS device. Since dynamic power is proportional to the square of the power supply voltage, lowering the voltage reduces the power dissipation. However, to maintain or increase the performance of a circuit, its threshold voltage should be decreased by the same factor, which increases the subthreshold (leakage) current of transistors exponentially [1]. Therefore, with the trend of CMOS technology scaling, leakage power is becoming a dominant contributor to the total power consumption. To reduce leakage power, a large number of techniques have been proposed, including transistor sizing, multi-Vth, dual-Vth, optimal standby input vector selection, stacking transistors, etc.

Dual-V_{th} assignment [2-6] is an efficient technique to decrease leakage power. Wei *et al*. [3] describe an algorithm to find the optimal high V_{th} for different circuit structure. However, in reality, the available threshold voltages in a process are predetermined and a designer does not have the choice of arbitrary V_{th}. The back trace algorithm [3] used to determine the dual-V_{th} assignment also has a disadvantage. Because the back trace search direction for non-critical paths is always from primary outputs to primary inputs, the gates close to the primary outputs always have the priority to be assigned high V_{th}, even though their leakage power reduction due to V_{th} increase may be smaller than that of gates close to the primary inputs. This algorithm only gives a possible solution, not an optimal one. On the contrary, using ILP, a global optimization solution can be easily achieved. Nguyen *et al.* [6] use linear programming (LP) to minimize the leakage and dynamic power by gate sizing and dual-threshold voltage devices assignment. However, they have not considered the glitch power, which can account for 20%-40% of the dynamic switching power [7]. To eliminate these unnec-

V. Paliouras, J. Vounckx, and D. Verkest (Eds.): PATMOS 2005, LNCS 3728, pp. 217-226, 2005.

essary transitions, a designer can adopt techniques of hazard filter [7] or path balance [8]. Raja *et al*. [8] have proposed a technique which uses a reduced constraint set linear program (LP) to eliminate dynamic glitch power.

The present work was motivated by the above research. A new ILP 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 ILP method is specifically devised with a set of constraints whose size is linear in the number of gates. Thus, large circuits can be handled. The ILP either holds the critical path delay corresponding to the all-low V_{th} gates, or allows an increase by a user-specified amount. As a result, a tradeoff between power saving and performance degradation can be allowed.

To deal with the complexities of delay models and leakage calculation, two look up tables for the delay and leakage current 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.

2 Leakage and Delay

The leakage current of a transistor is mainly the result of gate leakage, reverse bias PN junction leakage and subthreshold leakage. Compared to the subthreshold leakage, the reverse bias PN junction leakage can be ignored. The subthreshold leakage current is the weak inversion current between source and drain of an MOS transistor when the gate voltage is less than the threshold voltage [1]. Subthreshold current is given by [2]:

$$I_{sub} = u_0 C_{ox} \frac{W_{eff}}{L_{eff}} V_T^{\,2} e^{1.8} \exp\left(\frac{V_{gs} - V_{th}}{n V_T}\right) \cdot \left(1 - \exp\left(\frac{-V_{ds}}{V_T}\right)\right) \tag{1}$$

where u_0 is the zero bias electron mobility, and n is the subthreshold slope coefficient. Due to the exponential relation between V_{th} and I_{sub}, we can increase the V_{th} to reduce the subthreshold current sharply.

Our SPICE simulation results on the leakage current of a two-input NAND gate show that, for 70nm CMOS technology (Vdd=1V, Low V_{th}=0.20V, High V_{th}=0.32V), the leakage current in a high V_{th} gate is only about 2% of the leakage current in 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 98%, which is a significant improvement.

However, the gate delay increases with the increase of V_{th}. From SPICE simulation result for a NAND gate delay when the output fans out to a specified number of inverters, we observe that the gate delay increases 30%-40% by increasing V_{th} form 0.20V to 0.32V.

Thus, we can make tradeoffs between leakage power and performance, leading to a significant reduction in the leakage power while sacrificing only some or no circuit performance. Such a tradeoff is made in the ILP. Results in Section 4.1 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%.

3 Integer Linear Programming

To minimize the leakage power, we use an ILP model to determine the optimal assignment of V_{th} while controlling any sacrifice in performance. Due to the constraints on the maximum path delay, all the gates on the critical path are assigned low V_{th}. The V_{th} assignments of gates on the non-critical path are determined jointly by their delay increases and leakage reductions if high V_{th} were assigned to them. To eliminate the glitch power, additional ILP constraints determine the positions and values of the delay elements to be inserted to balance path delays. Unlike the heuristic algorithms [2-5], this ILP gives us a globally optimal solution.

We can easily make a tradeoff between power reduction and performance degradation by changing the constraint for the maximum path delay in the ILP model.

3.1 ILP for Leakage Power reduction

Raja *et al.* [8] proposed a LP formulation to reduce dynamic glitch power by a reduced constraint set linear program whose number of constraints is proportional to the total number of gates. We first modify their formulation into an integer linear program (ILP) to reduce subthreshold leakage power as described below.

3.1.1 Variables

Each gate has two variables.

T_i: the 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.

X_i: the assignment of low or high V_{th} to gate i; X_i is an integer 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}.

3.1.2 Objective function

In a CMOS static circuit, the leakage power is

$$P_{leak} = V_{dd} \sum_i I_{leaki} \tag{2}$$

If we know the leakage currents of all gates, the leakage power can be easily obtained. Therefore, the objective function for this ILP is to minimize the sum of all gate leakage currents, which is given by

$$Min \sum_i \left(X_i \cdot I_{Li} + (1 - X_i) \cdot I_{Hi} \right) \tag{3}$$

I_{Li} is the leakage current of gate i with low $V_{th;}$

I_{Hi} is the leakage current of gate i with high $V_{th;}$

The leakage current of a gate depends on the input vector. Therefore, we make a leakage current look up table, which is indexed by the gate type and the input vector. I_{Li} and I_{Hi} can both be searched from this look-up table. The values in this lookup table, as found by Smart-SPICE simulation, are the total leakage currents including sub-threshold and gate leakages of a cell under specific input vector conditions.

3.1.3 Constraints

- **Constraints for each gate:**

$$T_i \geq T_j + X_i \cdot D_{Li} + (1 - X_i) \cdot D_{Hi} \tag{4}$$

$$0 \leq X_i \leq 1 \tag{5}$$

Constraint (5) assigns either low V_{th} (X_i =1) or high V_{th} (X_i =0) to gate i ;
D_{Hi} is the delay of gate i with high V_{th};
D_{Li} is the delay of gate i with low V_{th}.

With the increase of the fanout, the delay of the gate also increases proportionately. Therefore, a second look-up table is constructed and specifies the delay for given gate type and fanout number. D_{Hi} and D_{Hi} can be searched from this look-up table indexed by the gate type and the number of fanout of gate i.

We explain constraint (4) using the circuit of Figure 1. 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 inequality (4) are given by

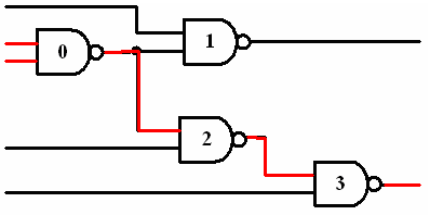

Fig. 1. Circuit for explaining ILP constraints.

$$T_2 \geq T_0 + X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \tag{6}$$

$$T_2 \geq 0 + X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \tag{7}$$

T_2 that satisfies both inequalities is the latest time at which an event (signal change) may occur at the output of gate 2.

- **Max delay constraints for primary outputs (PO):**

$$T_i \leq T_{\max} \tag{8}$$

T_{max} can be the critical path delay when all the gates are assigned low V_{th} or the maximum delay specified by the designer. We use a simplified ILP model, whose description is omitted here, to find the delay T_c of the critical path. If T_{max} equals T_c the objective function of our ILP 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.

3.2 ILP for Leakage Power and Dynamic Glitch Power reduction

Glitches can account for 20%-40% dynamic power [7]. To eliminate these unnecessary transitions, a designer can adopt techniques of hazard filter [7] or path balance [8]. Combined with the method of path balance, the technique of Section 3.1 can be extended to reduce leakage power and dynamic glitch power simultaneously. Such an extended ILP model is developed below.

3.2.1 Variables

Each gate has four variables:

X_i: the assignment of low or high V_{th} to gate i; X_i is an integer 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}.

T_i: the 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: the 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.

$\Delta d_{i,j}$: the delay of the inserted buffer at the j_{th} input path of gate i.

3.2.2 Objective Function

The objective function for this ILP is to minimize the sum of all gate leakage currents and the sum of all inserted delays:

$$Min\left(\sum_i I_{leaki} + \sum_i \sum_j \Delta d_{i,j}\right) = Min\left(\sum_i \left(X_i I_{Li} + (1 - X_i) I_{Hi}\right) + \sum_i \sum_j \Delta d_{i,j}\right) \tag{9}$$

Besides the objective to minimize the leakage power consumption which is the same as Equation (3), an additional objective function is to minimize the glitch power. We insert minimal delays to balance path delays and eliminate glitches. This leads to another objective function:

$$Min \sum_i \sum_j \Delta d_{i,j} \tag{10}$$

Our objective function (9) combines objectives (3) and (10).

When implementing these delay elements, we use transmission gates with only the gate leakage, which is much smaller than the subthreshold leakage and can be ignored.

3.2.3 Constraints

- **Constraints for each gate:**

$$0 \le X_i \le 1 \tag{11}$$

$$T_i >= T_j + \Delta d_{i,j} + \left(X_i \cdot D_{Li} + (1 - X_i) \cdot D_{Hi}\right) \tag{12}$$

$$t_i <= t_j + \Delta d_{i,j} + \left(X_i \cdot D_{Li} + (1 - X_i) \cdot D_{Hi}\right) \tag{13}$$

$$X_i \cdot D_{Li} + (1 - X_i) \cdot D_{Hi} >= T_i - t_i \tag{14}$$

where, *i* is the the gate on which constraints are set, and *j* is the the gate whose output is gate *i*'s fanin. Constraints (12-14) ensure that gate *i*'s inertial delay is always larger than the delay difference of its input paths by inserting some delays on its faster input paths. Therefore, glitches can be eliminated.

We explain constraints (12-14) using the circuit shown in Figure 1. 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 inequality (12-14) are:

$$T_2 \geq T_0 + X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \tag{15}$$

$$T_2 \geq 0 + X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \tag{16}$$

$$t_2 \leq t_0 + X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \tag{17}$$

$$t_2 \leq 0 + X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \tag{18}$$

$$X_2 \cdot D_{L2} + (1 - X_2) \cdot D_{H2} \geq T_2 - t_2 \tag{19}$$

Time T_2 that satisfies both inequalities (15) and (16) is the latest time at which an event (signal change) may occur at the output of gate 2.

Time t_2 is the earliest time at which an event may occur at the output of gate 2, if it satisfies both inequalities (17) and (18).

Constraint (19) means that the difference of T_2 and t_2, which equals the delay difference between two input paths, is smaller than gate 2's inertial delay, which may be either low V_{th} gate delay, D_{L2}, or high V_{th} gate delay, D_{H2}.

- **Max delay constraints for primary outputs (PO):**

$$T_i \leq T_{\max} \tag{20}$$

As in Section 3.1, T_{max} can be the maximum delay specified by the circuit designer or the critical path delay.

When we use the ILP 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 in Figure 2 is shown in Figure 3. The label in or near a gate is its inertial delay.

Three black shaded gates are assigned high V_{th} since they are not on the critical path and their delay increases do not affect the critical path delay. Two delay elements (grey shaded) are inserted to eliminate glitches. Although delay elements, if implemented as buffer gates, may consume additional leakage power, we may assign high V_{th} to them. Therefore, the three low V_{th} gates (without shading) on the critical path still dominate the total leakage power. Actually, in our design, delay elements are implemented by CMOS transmission gates that have no subthreshold leakage. Transmission gates also consume very little dynamic power since they are not driven by any supply rails [9].

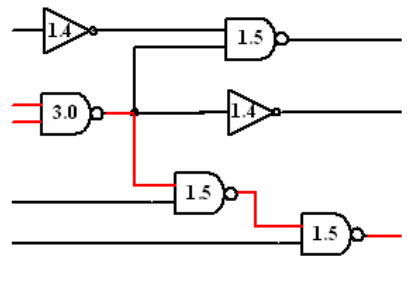

Fig. 2. Unoptimized circuit with potential glitches

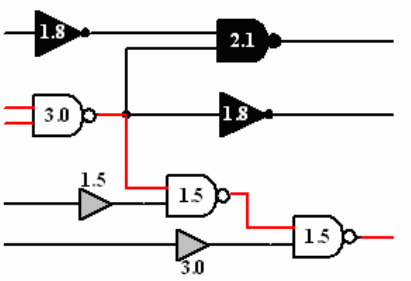

Fig. 3. Optimized circuit

4 Results

To study the increasingly dominant effect of leakage power dissipation, we use the BPTM 70nm CMOS technology. Low V_{th} for NMOS and PMOS are 0.20V and -0.22V. High V_{th} for NMOS and PMOS are 0.32V and -0.34V, respectively.

We regenerate the netlists of all 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 are 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 [10]. CPLEX then give the optimal V_{th} assignment as well as the value and position of every delay element.

4.1 Leakage Power Reduction

The results of the leakage power reduction for ISCAS'85 benchmark circuits are shown in Table 1. The numbers of gates in column 2 are for the gate library used and differ from those for original benchmark netlists. T_c in column 3 is the minimum delay of the critical path when all gates have low V_{th}. Column 4 shows the leakage reduction (%) for optimization without sacrificing any performance. Column 6 shows the leakage reduction with 25% performance sacrifice. The CPU times shown are for the ILP runs and are, as expected, linear in circuit size since both number of variables and number of constraints are linear in circuit size. From Table 1, 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 4). For several large benchmarks leakage is reduced by 90% due to a smaller percentage of gates being on the critical path. However, for some highly symmetrical circuits, which have many critical paths, such as C499 and C1355, the leakage reduction is less. Column 6 shows that the leakage reduction reaches the highest level, around 98%, with some performance sacrifice.

The curves in Fig. 4. 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 in a slower reduction trend. When the delay increase is more than 30%, the leakage reduction saturates at about 98%. Therefore, Figure 4 provides a guide for making a tradeoff between leakage and power.

Table 1. Leakage reduction due to dual-V_{th} reassignment (@ 27℃).

Ckt.	Gates #	T_c (ns)	Leakage Red. (%) (T_{max}=T_c)	Sun OS 5.7 CPU secs.	Leakage Red. (%) (T_{max}=1.25T_c)	Sun OS 5.7 CPU secs.
C432	160	0.75	61.0	0.25	95.0	0.25
C499	182	0.39	19.3	0.31	94.8	0.30
C880	328	0.67	88.1	0.54	96.5	0.53
C1355	214	0.40	25.0	0.33	93.3	0.36
C1908	319	0.57	66.4	0.57	96.6	0.56
C2670	362	1.26	90.4	0.68	97.9	0.53
C3540	1097	1.75	93.8	1.71	98.0	1.70
C5315	1165	1.59	87.1	1.82	98.0	1.83
C6288	1177	2.18	73.8	2.07	97.1	2.00
C7552	1046	1.92	96.0	1.59	98.0	1.68

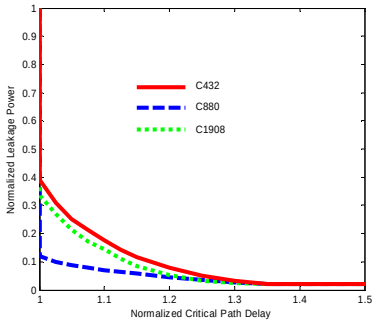

Fig. 4. Tradeoff between leakage power and performance.

4.2 Leakage, Dynamic Glitch and Total Power Reduction

The leakage current strongly depends on the temperature. Our SPICE simulation shows that for a 2-input NAND gate with low V_{th}, when temperature increases from 27℃ to 90℃, the leakage current increases by a factor 10X. For a 2-input NAND gate with high V_{th}, this factor is 20X. The leakage in the look-up table is from simulation for a 27℃ operation. To manifest the dominant effect of leakage power, we estimate the leakage currents at 90℃ by multiplying the total leakage current obtained from

CPLEX [10] by a factor between 10X and 20X as determined by the proportion of low to high threshold transistors.

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

$$P_{dyn} = \frac{E_{dyn}}{T} = \frac{0.5 \cdot C_{inv} \cdot V_{dd}^{\ 2} \cdot \sum_i T_i FO_i}{1000(1.2 \cdot T_c)} \tag{21}$$

where C_{inv} is the gate capacitance of one inverter, T_i is the number of transitions at gate i's output when 1,000 random test vectors are applied at PIs, and FO_i is the number of fanouts. Vector period is assumed to be 20% greater than the critical path delay, T_c. By simulating each gate's transition number, we can estimate the glitch power reduction.

Table 2. Leakage, dynamic and total power reduction comparison for unoptimized and optimized circuits (@ 90℃).

Ckt.	$P_{leak}1$ (uW)	$P_{leak}2$ (uW)	**Leak. Red. (%)**	$P_{dyn}1$ (uW)	$P_{dyn}2$ (uW)	**Dyn. Red. (%)**	$P_{total}1$ (uW)	$P_{total}2$ (uW)	**Total Red. (%)**
C432	35.8	11.9	**66.8**	101	73	**27.4**	137	85	**37.7**
C499	50.4	39.9	**20.7**	226	160	**29.0**	276	200	**27.5**
C880	85.2	11.1	**87.0**	177	128	**27.8**	263	139	**47.0**
C1355	54.1	40.0	**26.3**	293	166	**43.5**	347	206	**40.8**
C1908	92.2	29.7	**67.8**	255	198	**22.4**	347	227	**34.5**
C2670	116	11.3	**90.2**	129	101	**21.6**	244	112	**54.1**
C3540	303	18.0	**94.1**	333	228	**31.5**	636	246	**61.3**
C5315	421	9.80	**88.2**	466	304	**34.6**	887	354	**60.1**
C6288	389	97.2	**75.0**	1691	406	**76.0**	2080	503	**75.8**
C7552	444	18.8	**95.8**	381	228	**40.2**	825	247	**70.1**

We compare the leakage power and dynamic power at 90℃ in Table 2. The suffix-1 means the unoptimized circuit which has all the low threshold gates and the largest glitch power, and suffix-2 means the optimized circuit whose V_{th} has already been optimimally assigned and most of the glitches have been eliminated. We observe that for 70nm BPTM CMOS technology at 90℃, unoptimized leakage power (column 2) of some large ISCAS'85 benchmark circuits can account for about one half or more of the total power consumption (column 8). 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 most circuits are more than 50%. Some have a total reduction of up to 70%.

5 Conclusion

A new technique to reduce the leakage and glitch dynamic power simultaneously in a dual-V_{th} process is proposed in this paper. An integer linear programming (ILP) model

is generated from the circuit netlist and the AMPL CPLEX [10] solver determines the optimal V_{th} assignments for leakage power minimization and the delays and positions of inserted delay elements for glitch power reduction. The experimental results for ISCAS'85 benchmark 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 ILP formulation.

References

[1] 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.

[2] M. Ketkar and S. S. Sapatnekar, "Standby Power Optimization via Transistor Sizing and Dual Threshold Voltage Assignment," *Proc. ICCAD*, 2002, pp. 375-378.

[3] L. Wei, Z. Chen, M. Johnson and K. Roy, "Design and Optimization of Low Voltage High Performance Dual Threshold CMOS Circuits," *Proc. DAC*, 1998, pp. 489-494.

[4] L. Wei, Z. Chen, K. Roy, Y. Ye and V. De, "Mixed-Vth (MVT) CMOS Circuit Design Methodology for Low Power Applications," *Proc. DAC*, 1999, pp.430-435.

[5] Q. Wang, and S. B. K. Vrudhula, "Static Power Optimization of Deep Submicron CMOS Circuits for Dual V_T Technology," *Proc, ICCAD*, 1998, pp490-496.

[6] 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. ISLPED*, 2003, pp. 158-163.

[7] V. D. Agrawal, "Low Power Design by Hazard Filtering," *Proc. 10tth International Conference on VLSI Design*, 1997, pp. 193-197.

[8] T. Raja, V. D. Agrawal and M. L. Bushnell, "Minimum Dynamic Power CMOS Circuit Design by a Reduced Constraint Set Linear Program," *Proc. 16tth International Conference on VLSI Design*, 2003, pp. 527-532.

[9] N. R. Mahapatra, S. V. Garimella. 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.

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