

# Transistor Reordering Rules for Power Reduction in CMOS Gates

Wen-Zen Shen

Dept. of Electronics Engineering  
& Institute of Electronics  
National Chiao Tung Univ.  
HsinChu, 30050, Taiwan  
e-mail: wzshen@cc.nctu.edu.tw

Jiing-Yuan Lin

Institute of Electronics  
National Chiao Tung Univ.  
HsinChu, 30050, Taiwan  
e-mail: jylin@yankees.ee.  
nctu.edu.tw

Fong-Wen Wang

Telecommunication Lab.  
Ministry of Transportation  
and Communication, Taiwan  
e-mail: wen%tlvaxl@tlrouter.  
motctl.gov.tw

**Abstract** — The goal of transistor reordering for a logic gate is to reduce the propagation delay as well as the charging and discharging of internal capacitances to achieve low power consumption. In this paper, based on the input signal probabilities and transition densities, we propose a set of simple transistor reordering rules for both basic and complex CMOS gates to minimize the transition counts at the internal nodes. The most attractive feature of this approach is that not only the power consumption is reduced efficiently, but also the other performances are not degraded. Experimental results show that this technique typically reduces the power by about 10% in average, but in some cases the improvement is even 35%.

## I. Introduction

The ever-reducing device size gives rise to the high performance of modern VLSIs. Due to their high speed and great transistor count, chips suffer from too much power dissipation. Excessive power dissipation may reduce chip reliability, shorten the lifetime and thus require extra device to remove heat. Moreover, for portable applications low power dissipation is of critical importance. Therefore, how to reduce power dissipation of integrated circuits while retaining their performance has become an important issue in the design of modern VLSIs.

Many techniques have been reported for low power such as technology mapping [1, 2], supply-voltage scaling [3], transistor reordering [4-7], and so on. The main idea of technology mapping for low power is to minimize the total capacitance load and average switching activities via proper mapping. In the voltage scaling technique, because of the quadratic dependence of voltage on power, slight decrease of voltage usually make dramatic power reduction. Both above approaches could reduce power consumption efficiently, but may also slow down the computational throughput or increase layout area. Transistor reordering is another interesting technique for low power. The attractive property of this technique is that it can be used in conjunction with other low power algorithms and reduce the power consumption without any other performance degradation. In [4, 5], the authors combined transistor reordering and transistor sizing, from the signal arrival time point of view, to optimize delay, area and power consumption. The approach reported in [6] is based on the signal's transition probability for reducing the signal frequency of internal nodes; however, complicated evaluation are needed for each reordering process. In [7], the authors have proposed a heuristic

reordering method for low power, but it is not sufficient to support the complex gates.

For a CMOS logic gate, the transition times at internal nodes are strongly dependent on the input signal characteristics and input positions. A naive transistor placement may increase the spurious transitions at internal nodes and result in extra power consumption. In this paper, we propose a set of simple input reordering rules for both basic and complex CMOS gates, from the signal probability and transition density point of view, to rearrange the transistor positions for power reduction. Experimental results shown that for some cases a proper input reordering of a logic gate could save even one third of power consumption. In addition, the orderings predicted by our rules are generally very close to the optimal orderings. In this paper, for convenience, we use the terms *input reordering* and *transistor reordering* almost synonymously.

The rest of this paper is organized as follows. The signal probability and transition density are defined in section II. Input ordering is the subject of section III where we discuss the relationship between input signal characteristics, input position and power consumption, and finally we conclude a set of reordering rules for both basic and complex gates. Experimental results for NAND, NOR, AOI, OAI, and a subset of *cmlex* and *MCNC* benchmarks are demonstrated in section IV. Finally, conclusions and future works are made in section V.

## II. Preliminaries and definitions

In [4], Najm has presented two continuous time probabilistic measures, equilibrium probability and transition density, under the assumption that the logic signal can be modeled as **strict-sense stationary mean-ergodic 0-1 process** [8, 9]. In this section, we redefine these probabilistic measures under the same assumption from discrete time point of view. In the following, we assume the combinational circuit is a synchronous digital system controlled by a global clock with cycle time  $T_{cycle}$ . To capture the glitches exactly in general gate delay circuits, we assume  $T_{sd}$  is the smallest gate delay in the circuit, and divide the clock period into  $S = T_{cycle}/T_{sd}$  slots.

The **signal probability**[12] of an input  $x_i$  being one at a given time, denoted as  $p_i$ , is given by

$$p_i = \lim_{N \rightarrow \infty} \frac{\sum_{k=1}^{N \times S} x_i(k)}{N \times S} \quad (1)$$

where  $N$  is the total number of global clock cycles and  $x_i(k)$  is the value of input  $x_i$  during the interval of time instances  $k$  and  $k+1$ . Similarly, the probability that  $x_i$  is zero at a given time, denoted as  $q_i$  or  $p_i^0$ , is  $q_i = 1 - p_i$ .

The **transition probabilities**[12] of  $x_i$  for the transition  $0 \rightarrow 0$ ,  $0 \rightarrow 1$ ,  $1 \rightarrow 0$ , and  $1 \rightarrow 1$  can be denoted by  $p_i^{00}$ ,  $p_i^{01}$ ,  $p_i^{10}$ , and  $p_i^{11}$ , respectively, where for example  $p_i^{10}$  is defined by:

$$p_i^{10} = \lim_{N \rightarrow \infty} \frac{\sum_{k=1}^{N \times S} x_i(k) \overline{x_i(k+1)}}{N \times S} \quad (2)$$

The other transition probabilities follow similarly. It is easy to verify that

$$p_i^{00} + p_i^{01} + p_i^{10} + p_i^{11} = 1 \quad (3)$$

$$p_i^{00} + p_i^{01} = q_i \quad (4)$$

$$p_i^{10} + p_i^{11} = p_i \quad (5)$$

There is another switching activity measure called **transition density** [8], denoted by  $D_i$  for signal  $x_i$ , which is defined as follows:

$$D_i = \lim_{N \rightarrow \infty} \frac{\sum_{k=1}^{N \times S} (x_i(k) \overline{x_i(k+1)} + \overline{x_i(k)} x_i(k+1))}{N \times S} = p_i^{01} + p_i^{10} \quad (6)$$

In general, for a digital signal,  $p_i^{01}$  is equal to  $p_i^{10}$  and  $D_i$  is twice that of  $p_i^{01}$  or  $p_i^{10}$ . Thus, it is well to know that only two parameters  $p_i$  and  $D_i$  are needed to determine the characteristics of signal  $x_i$ .

If the leakage current and short-circuit current are ignored, the average power dissipated by a CMOS gate is given by

$$P_{av} = \frac{1}{2} V_{dd}^2 (C_{out} \times D_{out} + \sum_i (C_i \times D_i)) \quad (7)$$

where  $C_i$  is the parasitic capacitance of internal node  $i$  and  $D_i$  is the transition density at internal node  $i$ .

### III. Input reordering

In [7], the authors use exhaustive enumeration method to determine the best input ordering. The approach is suited for the gate with a small number of transistors in series, but it is very inefficient for the gate with parallel-series or series-parallel structures such as AOI and OAI gates. Thus, a heuristic and precise reordering method is indeed necessary.

#### A. Input reordering strategy

In the following, we would find the relationship of input characteristics to power consumption based on Eqn. (7), and try to deduce reordering rules from the relationship to minimize the power consumption. For getting a clarity conclusion, we demonstrate a 2-input NAND gate shown in Fig. 1 as an example. Considering the first order input temporal correlation, the power consumption of NAND2 can be derived as follows:

$$\begin{aligned} P_{av} &= k(p_A^{11} p_B^{10} + p_A^{10} p_B^{11} + p_A^{10} p_B^{10}) \times C_{out} + k(p_A^{11} p_B^{01} + p_A^{01} p_B^{11} + p_A^{01} p_B^{01}) \times C_{out} \\ &\quad + k(p_A^{01} p_B^{10} + p_A^{11} p_B^{10} + p_A^{01} p_B^{00} p_i(int)) \times C_{int} \\ &\quad + k(p_A^{10} p_B^{01} + p_A^{11} p_B^{01} + p_A^{00} p_B^{01} p_h(int) + p_A^{01} p_B^{01} p_h(int)) \times C_{int} \\ &= 2k(p_A p_B^{10} + p_A^{10} p_B - p_A^{10} p_B^{10}) \times C_{out} \\ &\quad + k(2p_A p_B^{10} + p_A^{01} p_B^{00} p_i(int) + p_A^0 p_B^{01} p_h(int)) \times C_{int} \end{aligned} \quad (8)$$

where  $k$  is a constant,  $p_l(int)$  and  $p_h(int)$  are the probabilities of internal node being in low and high state, respectively, and  $p_A^{01} p_B^{00} p_i(int)$  represents that the internal node are in low state at time  $n-2$  and input patterns  $(A, B)$  is  $(0, 0)$  at time  $n-1$  and  $(1, 0)$  at time  $n$ . In Eqn. (8), there are four terms which in regular order are the power consumption equations for charging of output node, discharging of output node, charging of internal node, and discharging of internal node, respectively. Based on the definitions in section II, those equations can be simplified as Eqn. (9). In [13], we have proposed a state transition graph for power estimation (STGPE) to model the power consumption of a logic gate. So we could find the probabilities  $p_h(int)$  and  $p_l(int)$  by calculating the state probabilities of the STGPE of NAND2.

$$p_h(int) = \frac{p_A p_B^0 + p_A^{10} p_B^{00}}{1 - p_A^{00} p_B^{00}} \quad (10)$$

$$p_l(int) = 1 - p_h(int) \quad (11)$$

It is noteworthy that the transition density of the output node is not affected no matter how the inputs reorder. In other words, transistor reordering only affect the charging and discharging of internal nodes. In Eqn. (9), in general,

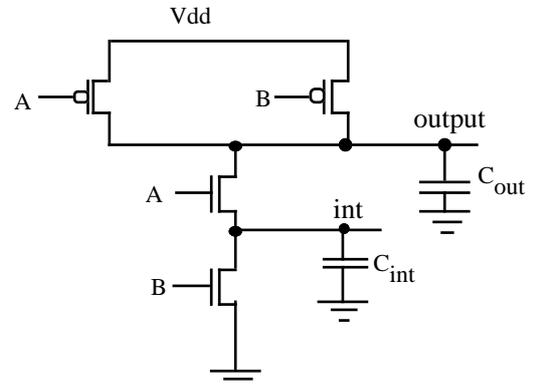


Fig. 1 A 2-input NAND gate (NAND2)

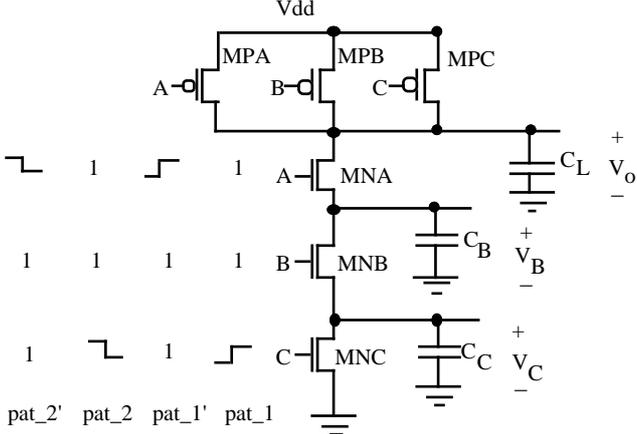


Fig. 2 A three-input NAND gate (NAND3)

$2p_A p_B^{10}$  dominate the power contribution of internal node. Thus, to minimize  $P_{av}$ , it is best to place the input with low signal probability near output node and the input with low transition density near ground end. To verify the usefulness of this rule, we evaluate the charging/discharging equation of internal node, i.e. the last term in Eqn. (9), with different  $P_A$ ,  $D_A$ ,  $P_B$ , and  $D_B$  and different ordering. By exhaustive simulation, we find that only about 7% of the exhaustive simulation cases violate our ordering rules and get lower power, but in most of these cases the power saving is very little.

### B. Critical input point of view

To illustrate the effect of input reordering more clearly, we give a three-input CMOS NAND gate shown in Fig. 2 as an example. If MNA, MNB, and MNC are turned on at time  $n-1$  and one of them would turn off at time  $n$  to make output high, power dissipation would depend on the position of the off transistor. That is, if MNA turns off, only  $C_L$  needs to be charged, while if MNC turns off, all  $C_L$ ,  $C_B$ , and  $C_C$  need to be charged. Thus, to avoid the charging of these internal capacitances, it is better to keep the NMOS near the output node off. In other words, the signal with high probability being one should be placed near ground end to reduce the transition counts of internal nodes.

Before further study, we should define the critical input [6]. The **critical input** of a gate is the input that causes the output node of the gate to switch. In Fig. 2. if the critical input is rising and placed near ground end, e.g.,  $pat_1$ , the charge of  $C_B$  and  $C_C$  must discharge before the charge of  $C_L$  discharge to zero. However, if the critical input is placed near output node, e.g.,  $pat_1'$ , the initial charge of  $C_B$  and  $C_C$  are zero and the delay time of  $C_L$  discharging is less than the case in pattern  $pat_1$ . Similarly, if the critical input is falling and placed near ground end, e.g.,  $pat_2$ ,  $C_B$  and  $C_C$  must be charged to  $V_{DD} - V_{THN}$ , where  $V_{THN}$  is the threshold voltage of NMOS, before  $C_L$  is charged to  $V_{DD}$ . However, for pattern  $pat_2'$ , only  $C_L$  needs to be charged. Thus, the delay time

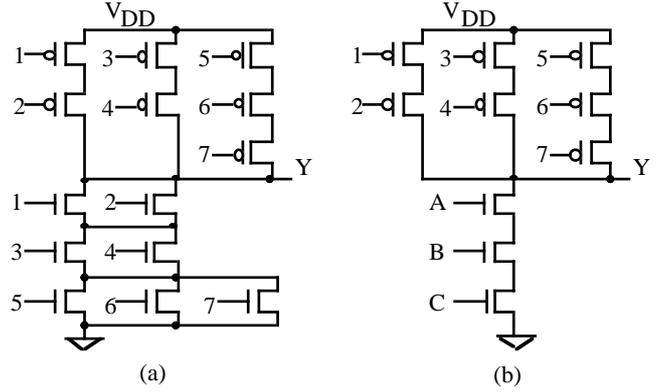


Fig. 3 An OAI223 gate and its simplified structure

of pattern  $pat_2'$  is less than pattern  $pat_2$ . In a word, if the critical input is placed more closer ground end, the propagation delay of the gate is more larger and more power is dissipated. In general, the input with high transition density is often the critical input. Therefore, to reduce the power dissipation we should place the input with low transition density near the ground end.

### C. Complex gates

The configuration of AOI and OAI gates are more complex than NAND and NOR gates; however, the reordering rules of NAND and NOR gates still satisfy the reordering of complex gates after simplifying the complex gates. Given a complex gate, we separate the circuit into two parts, n-block and p-block. n-block and p-block are the collection of NMOS and PMOS transistors, respectively. To simplify the reordering rules, the transistors in n-block and p-block is reordered separately. For an AOI gate, the reordering method of n-block is identical to the NAND gate case. Similarly, the reordering of p-block in OAI is identical to the NOR gate case. In the following, we present a method to simplify AOI and OAI gates for reordering the p-block in AOI gate and n-block in OAI gate.

Fig. 3(a) illustrated an OAI223 gate and 3(b) is the simplified circuit where we simplify the parallel-series NMOS transistors to a NMOS train. Given the signal probabilities of  $x_1$  to  $x_7$ , it is easy to evaluate the signal probabilities of  $A$ ,  $B$ , and  $C$  [10]. For transition density calculation, our approach is built on the notion of Boolean difference presented in [8], but consider the case where the input signals change state simultaneously. In Fig. 3(b), the transition probability of  $A$  from 0 to 1 is

$$P_A^{01} = p_{x_1}^{00} p_{x_2}^{01} + p_{x_1}^{01} p_{x_2}^{00} + p_{x_1}^{01} p_{x_2}^{01} \quad (12)$$

The last term in the right-hand side of Eqn. (12) represents the probability that both  $x_1$  and  $x_2$  are changing from 0 to 1 simultaneously, which is ignored in [8]. Based on Eqn. (12) and [10], it is easy to derive the signal probability and transition density for signal  $A$  as follows:

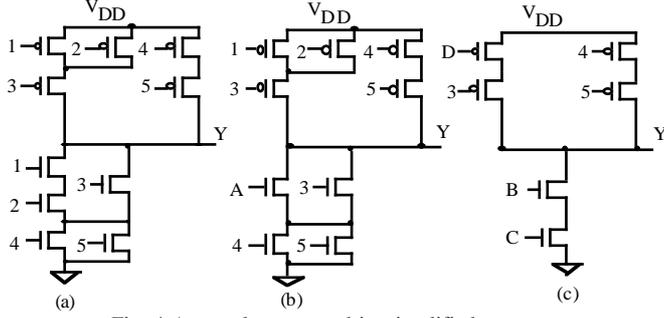


Fig. 4 A complex gate and its simplified structure

$$P_A = p_{x_1} + p_{x_2} - p_{x_1}p_{x_2} \quad (13)$$

$$D_A = D_{x_1}q_{x_2} + D_{x_2}q_{x_1} - \frac{1}{2}D_{x_1}D_{x_2} \quad (14)$$

According to these data, the reordering of n-block can be achieved by applying the reordering rule of NAND gate. Similarly, for an AOI gate, the input signal probabilities and transition densities of the simplified structure can be got by replacing  $p_i$  with  $q_i$  and  $q_i$  with  $p_i$  in Eqns. (13) and (14). So, the p\_block of AOI gate can be reordered with the same manner of NOR gates.

For the reduction of transistors in series, for example transistors 1 and 2 in Fig. 4(a), the signal probability and transition density can be got as follows

$$P_A = p_{x_1}p_{x_2} \quad (15)$$

$$D_A = D_{x_1}p_{x_2} + D_{x_2}p_{x_1} - \frac{1}{2}D_{x_1}D_{x_2} \quad (16)$$

The simplification and reordering of  $B$  and  $C$  are identical to the case of OAI223. So, reordering of a complex gate can be easily achieved after these simplification.

#### D. Summary of reordering rules

Based on the discussions above, the reordering rules for logic gates are summarized as follows:

For *NAND* gate

**Rule 1:** Place the input signal with high signal probability  $p_i$  near ground.

**Rule 2:** Place the input signal with low transition density  $D_i$  near ground.

For *NOR* gate

**Rule 1:** Place the input signal with low signal probability  $p_i$  near power supply.

**Rule 2:** Place the input signal with low transition density  $D_i$  near power supply.

For *complex* gate

**Rule 1:** Simplify the complex gates to *NMOS* trains and *PMOS* trains.

**Rule 2:** Obey the rules of *NAND* and *NOR* gates for *NMOS* and *PMOS* trains, respectively.

Sometimes rule 1 and rule 2 for both *NAND* gate and *NOR* gate may conflict. Then the reordering is determined by the ratio of  $p_i$  to  $D_i$  i.e.,  $p_i/D_i$  ( $q_i/D_i$  for *PMOS* trains). For *NMOS* (*PMOS*) trains, input signals with higher ratio

are placed near ground ( $V_{DD}$ ) end.

## IV. Experimental results

In this section, we will present the preliminary experimental results using the input reordering rules described in the last section. To show the efficiency of our rules, we carried out many experiments on various basic and complex gates as well as a subset of *cmlex-91* and *MCNC-91* benchmarks on a SUN SPARCstation 10 machine. In the experimental results, power dissipation is in micro-watts and CPU time is in seconds. All of power dissipations are estimated by the exact *SPICE* simulation. The transistor models used are the level 3 model of  $0.8\mu\text{m}$  SPDM CMOS technology obtained from *CIC* (*Chip Implementation Center*). During *SPICE* simulation, 1000 input patterns with  $T_{cycle} = T_{sd} = 50\text{ns}$  and 1ns rise/fall time are generated randomly according to the input signal probabilities and transition densities, and then imposed on the inputs of the basic and complex gates.

For convenience, the *NMOS* train of *NAND* gate with input signals  $A$ ,  $B$ , and  $C$  is denoted by *NAND\_ABC* where the label is ordered from the output end to the ground end. Similarly, the *PMOS* train of *NOR* gate is denoted by *NOR\_ABC* if  $A$ ,  $B$ ,  $C$  are ordered from the power supply end to the output end. Table 1 illustrates the *SPICE* simulation results for two inputs *NAND* and *NOR* gates with different input ordering, and Table 2 shows the results for three inputs *NAND* and *NOR* gates. In Table 1, both signal probability and transition density of signal  $B$  are 0.5. In Table 2, all input signal probabilities are assumed to be 0.5 in the second column, and all the input transition densities are assumed to be 0.5 in the third column. The symbol "*EFF.*" in the table means the power efficiency of input reordering and is defined as follows:

$$EFF. = \frac{Pow_{worst} - Pow_{best}}{Pow_{worst}} \times 100\% \quad (17)$$

where  $pow_{worst}$  and  $pow_{best}$  are the power dissipations under the worst ordering case and best ordering case, respectively. For *NAND* (*NOR*) gate in Table 1 and Table 2, it is worthy to note that the input with high probability being one (zero) and low transition density should be placed near the ground ( $V_{DD}$ ) end to achieve low power dissipation.

For complex gates, *OAI223*, *AOI223* and a complex gate shown in Fig. 4 are used to check the validity of our reordering rules. Table 3 shows the experimental results of *AOI223* and *OAI223* by reordering the inputs of both n-block and p-block. For *AOI223*, the simulation results of the best and worst ordering predicted by our rules are listed in the second and third columns respectively, while the results of the best ordering and worst ordering examined by exhaustive simulation are listed in the fourth and fifth columns respectively. Table 5 demonstrates the simulation results of a complex gate shown in Fig. 4(a) by reordering the inputs of both n-block and p-block. The input characteristics of the patterns used in Table 3 and

Table 5 are tabulated in Table 4 and Table 6, respectively. From Tables 3 and 5, it is seen that the best and worst input ordering predicted by our method are very close to the optimal best and worst orderings. In addition, from Table 3, it can be found that a good input ordering may save even one third of power consumption with respect to the worst ordering.

In Table 7, we show several reordering results for *cmlex* and *MCNC* benchmark circuits. In our procedure, *misII* [11], with library *mcnc.genlib*, are used for logic optimization and technology mapping. Table 8 tabulates the signal probabilities and transition densities, which are generated randomly, for benchmark *cm150a* since this circuit has the maximal number of input in Table 7. In Table 7, if the benchmark circuit evaluated has  $m$  inputs ( $m \leq 21$ ), we take the first  $m$  inputs of *cm150a* listed in Table 8 as the input signal characteristics of the given benchmark. Because of the lack of efficient transition density simulator considering general gate delay, we use *VERILOG* simulator to estimate the signal probabilities and transition densities for each node in the circuit network. All input signals for *SPICE* and *VERILOG* simulations are generated similar to the case of basic and complex gates except that  $T_{sd}$  is 1ns (the smallest gate delay in benchmark circuits). According to the signal characteristics of each node in the circuit network, we reorder the transistors for each gate individually to minimize the power consumption. "CPU Time" in Table 7 is the running time of *VERILOG* simulation and transistor reordering. *SPICE* simulation time are not included in "CPU Time". In fact, *VERILOG* simulation time take about 95-percent of the "CPU Time". From Table 7, although the power reduction of input reordering for some benchmarks are small, however, input reordering is still an effective approach for low power design. The most advantage of this method is that it is orthogonal to other low power algorithms.

## V. Conclusions

The major contribution of transistor reordering is that the power dissipation could be reduced dramatically without extra layout area or decrease in speed. In addition, it can be used in conjunction with other low power algorithm to achieve further improvement. In this paper, we have presented a set of reordering rules for NAND, NOR, AOI, OAI, and other complex gates. Experimental results show that the input order rearranged by our rules is very close to the optimal realization.

The method presented in this paper is currently limited to combinational static CMOS circuits; however, it can be extended to other MOS circuit structures such as pass transistor circuits and dynamic CMOS circuits. In our procedure signal probabilities and transition densities are estimated by *VERILOG* simulation. *VERILOG* simulation takes much time in our reordering procedure; therefore, an efficient transition density simulator considering variable gate delay is necessary. In our experiments, if the reordering rule 1 and rule 2 for NAND(NOR) gate are conflicting, then the ordering is determined by comparing

the ratios of  $p_i(q_i)$  to  $D_i$ . However, this comparison is only a simple judgment, which may result in a poor reordering. In fact, the parasitic capacitances of each internal node are an important factor for determining the reordering when the conflict occurs. In the future work, this problem will be further studied. In addition, from probabilistic point of view, combining transistor reordering, transistor sizing, and the loading of interconnection line with the optimization of power, delay, and area will be an interesting problem.

## REFERENCES

- [1] V. Tiwari, P. Ashar, S. Malik, "Technology Mapping for Low Power," In ACM/IEEE 30th Design Automation Conference, pages 74-79, 1993.
- [2] C. Y. Tsui, M. Pedram, A. M. Despain, "Technology Decomposition and Mapping Targeting Low Power Dissipation," In ACM/IEEE 30th Design Automation Conference, pages 68-73, 1993.
- [3] A. Chandrakasan and R. W. Brodersen, "Low-power CMOS digital design," IEEE Journal of Solid-State, pp. 473-483, April 1992.
- [4] B. S. Carlson and C. Y. R. Chen, "Performance enhancement of CMOS VLSI circuits by transistor reordering," IEEE Design Automation Conference, pp. 361-366, 1993.
- [5] C. H. Tan and J. Allen, "Minimization of Power in VLSI Circuits Using Transistor Sizing, Input Ordering, and Statistical Power Estimation," In Proc. International Workshop on Lower Power Design, pp. 75-80, 1994.
- [6] J. Akita and K. Asada, "A Method for Reducing Power Consumption of CMOS Logic Based on Signal Transition Probability," In Proc. The European Design and Test Conference, pp. 420-424, 1994.
- [7] S. C. Prasad and K. Roy, "Circuit Optimization for Minimization of Power Consumption under Delay Constraint," In Proc. International Workshop on Lower Power Design, pp. 15-20, 1994.
- [8] F. N. Najm, "Transition Density : A New Measure of Activity in Digital Circuits," IEEE Trans. Computer-Aided Design, Vol. 12, No. 2, pp. 310-323, Feb. 1993.
- [9] Paul G. Hoel, Sidney C. Port, and Charles J. Stone, Introduction to Stochastic Process, Mei Ya Publications, INC. 1990.
- [10] K. P. Parker and E. J. McCluskey, "probabilistic treatment of general combinational networks," IEEE Trans. Computers, vol. 24, pp. 668-670, June, 1975.
- [11] R. Brayton, R. Rudell, A. Sangiovanni-Vincentelli, and A. Wang, "MIS: A multiple-level logic optimization system," IEEE Trans. Computer-Aided Design, Vol. CAD-6, pp. 1062-1081, Nov. 1987.
- [12] A. Ghosh, S. Devadas, K. Keutzer, J. White, "Estimation of Average Switching Activity in Combinational and Sequential Circuits," In ACM/IEEE 29th Design Automation Conference, pp. 253-259, 1992.
- [13] J. Y. Lin, T. C. Liu, W. Z. Shen, "A Cell-Based Power Estimation in CMOS Combinational Circuits," In Proc. IEEE International Conference on Computer-Aided Design, pp. 304 - 309, 1994

Table 1 Simulation results of NAND2 and NOR2 gates with different input ordering

(p(A), D(A))	NAND_AB	NAND_BA	EFF.	NOR_AB	NOR_BA	EFF.
(0.3, 0.1)	30.27	28.01	7.47%	56.75	71.92	21.79%
(0.3, 0.3)	41.55	43.62	4.75%	70.49	81.01	12.99%
(0.3, 0.5)	55.79	61.47	9.24%	89.74	96.77	7.26%
(0.5, 0.1)	47.49	40.24	15.27%	41.79	52.06	19.73%
(0.5, 0.3)	63.38	60.54	4.48%	59.73	63.31	5.65%
(0.5, 0.7)	89.78	79.44	11.52%	92.59	90.64	2.11%
(0.5, 0.9)	103.53	105.67	2.03%	112.87	109.97	2.57%
(0.7, 0.1)	67.81	56.03	17.37%	28.05	33.82	17.06%
(0.7, 0.3)	79.44	70.94	10.70%	50.16	49.44	1.44%
(0.7, 0.5)	94.84	87.21	8.05%	66.75	60.54	9.30%

Table 2 Simulation results of NAND3 and NOR3 gates with different input ordering

	D(A) = 0.7 D(B) = 0.5 D(C) = 0.3 p = 0.5	p(A) = 0.7 p(B) = 0.5 p(C) = 0.3 D = 0.5		D(A) = 0.7 D(B) = 0.5 D(C) = 0.3 p = 0.5	p(A) = 0.7 p(B) = 0.5 p(C) = 0.3 D = 0.5
Type and Order	Power (uW)	Power (uW)	Type and Order	Power (uW)	Power (uW)
NAND_CBA	45.33	39.21	NOR_CBA	41.41	37.24
NAND_CAB	45.31	39.65	NOR_CAB	42.42	37.79
NAND_BCA	44.67	40.94	NOR_BCA	41.87	38.16
NAND_BAC	44.76	41.73	NOR_BAC	43.55	38.38
NAND_ACB	44.25	41.77	NOR_ACB	43.32	40.47
NAND_ABC	44.22	42.02	NOR_ABC	44.09	40.09
EFF.	2.45%	6.69%	EFF.	6.08%	7.11%

Table 3 Simulation results by reordering the inputs of both n-block and p-block of AOI\_223 and OAI\_223

	AOI_223				OAI_223					
	Our method		Exhaustive		Our method		Exhaustive			
	best(A)	worst(B)	best	worst	best(C)	worst(D)	best	worst		
pat. 1	37.11	53.44	37.08	53.56	30.56%	71.11	86.56	69.81	87.51	17.85%
pat. 2	51.74	69.02	51.12	69.35	25.04%	55.46	64.59	55.01	64.96	14.14%
pat. 3	42.16	62.10	41.78	62.27	32.11%	48.55	58.09	48.55	58.11	16.42%
pat. 4	57.35	70.54	57.29	70.67	18.70%	73.51	112.35	73.51	112.35	34.57%
pat. 5	36.50	47.33	36.50	48.49	22.88%	44.89	59.44	44.89	59.44	24.48%
pat. 6	52.06	72.48	52.06	72.48	28.17%	63.09	77.74	63.09	78.38	18.84%

Table 4 Input characteristics of the input patterns used in Table 3 (p<sub>1</sub>, D<sub>1</sub>; p<sub>2</sub>, D<sub>2</sub>; p<sub>3</sub>, D<sub>3</sub>; .....)

pat. 1 (0.9, 0.1; 0.7, 0.2; 0.6, 0.2; 0.5, 0.4; 0.3, 0.4; 0.2, 0.3; 0.2, 0.2)  
pat. 2 (0.5, 0.1; 0.5, 0.08; 0.5, 0.6; 0.5, 0.4; 0.5, 0.2; 0.5, 0.1; 0.5, 0.1)  
pat. 3 (0.3, 0.2; 0.4, 0.2; 0.7, 0.2; 0.8, 0.2; 0.4, 0.2; 0.3, 0.2; 0.4, 0.2)  
pat. 4 (0.3, 0.4; 0.4, 0.6; 0.8, 0.2; 0.2, 0.2; 0.7, 0.2; 0.9, 0.1; 0.3, 0.1)  
pat. 5 (0.9, 0.15; 0.2, 0.3; 0.8, 0.1; 0.7, 0.1; 0.2, 0.2; 0.5, 0.3; 0.7, 0.3)  
pat. 6 (0.7, 0.4; 0.5, 0.6; 0.8, 0.1; 0.6, 0.15; 0.7, 0.3; 0.3, 0.4; 0.4, 0.2)

Table 5 Simulation results of Fig. 4 with different input ordering

	Our method		Exhaustive		EFF.
	best (D)	worst (E)	best	worst	(E-D)/E
pat. 1	34.29	43.68	34.29	43.68	21.50%
pat. 2	42.14	51.75	42.14	51.75	18.57%
pat. 3	28.20	37.36	27.23	37.93	24.52%
pat. 4	41.52	56.56	41.52	56.56	26.59%
pat. 5	55.28	60.60	55.28	60.79	8.78%
pat. 6	22.36	31.45	22.36	31.90	28.90%
pat. 7	34.36	40.45	32.18	41.45	15.06%
pat. 8	46.17	60.11	44.62	61.44	23.19%

Table 6 Input characteristics of the input patterns used in Table 5

pat. 1 (0.2, 0.2; 0.7, 0.2; 0.4, 0.2; 0.3, 0.2; 0.5, 0.2)  
pat. 2 (0.5, 0.1; 0.5, 0.2; 0.5, 0.2; 0.5, 0.4; 0.5, 0.3)  
pat. 3 (0.3, 0.1; 0.4, 0.3; 0.8, 0.2; 0.7, 0.1; 0.6, 0.15)  
pat. 4 (0.4, 0.4; 0.3, 0.3; 0.8, 0.2; 0.5, 0.2; 0.3, 0.3)  
pat. 5 (0.8, 0.2; 0.7, 0.3; 0.3, 0.4; 0.5, 0.2; 0.7, 0.2)  
pat. 6 (0.4, 0.2; 0.3, 0.3; 0.7, 0.1; 0.3, 0.2; 0.8, 0.1)  
pat. 7 (0.8, 0.2; 0.2, 0.3; 0.4, 0.1; 0.8, 0.2; 0.5, 0.1)  
pat. 8 (0.5, 0.1; 0.4, 0.4; 0.7, 0.4; 0.8, 0.1; 0.6, 0.2)

Table 7 Experimental results of cmlex and MCNC benchmarks

Circuit	Inputs	Outputs	NO. of Transistors	NO. of Gates	Avg. Power (A) best reordering	Avg. Power (B) worst reordering	EFF. (B-A)/B	CPU Time
C17*	5	3	24	6	77.12	79.73	3.26%	4.2
cm138a*	6	8	74	17	89.85	95.13	5.55%	4.9
cm150a*	21	1	216	47	688.74	785.1	12.27%	25.2
cm151a*	12	2	104	23	296.66	338.03	12.24%	12.6
cm152a*	11	1	64	11	199.47	228.01	12.52%	10
cm162a*	14	5	172	40	500	547.65	8.70%	18
cm163a*	16	5	164	38	503.04	553.95	9.19%	17.7
cm42a*	4	10	94	25	128.36	138.94	7.61%	7.7
cm82a*	5	3	72	16	272.02	303.58	10.40%	8
cm85a*	11	3	142	31	392.15	429.25	8.64%	12.8
cmb*	16	4	166	36	305.58	315.33	3.11%	15.2
con1^	7	2	64	15	198.5	216.89	8.48%	7.5
f2^	4	4	64	14	175.62	202.36	13.21%	5.7
f51m^	8	8	488	104	1367.3	1505	9.15%	36.3
mixex1^	8	7	178	40	418.97	466.9	10.27%	14.1
rd53^	5	3	200	46	576.03	625.22	7.87%	17
rd73^	7	3	578	124	1350	1475.3	8.49%	49.8
sao2^	10	4	588	116	941.77	993.87	5.24%	35.4

\*: cmlex benchmark

^: MCNC benchmark

Table 8 The input signal probabilities and transition densities of cm150a (p<sub>1</sub>, D<sub>1</sub>; p<sub>2</sub>, D<sub>2</sub>; p<sub>3</sub>, D<sub>3</sub>; .....)

(0.3, 0.002; 0.6, 0.008; 0.5, 0.004; 0.8, 0.004; 0.4, 0.008; 0.7, 0.0048; 0.5, 0.008; 0.2, 0.006; 0.4, 0.004; 0.8, 0.0032; 0.4, 0.006; 0.7, 0.002; 0.6, 0.008; 0.5, 0.006; 0.8, 0.004; 0.5, 0.0068; 0.2, 0.0048; 0.6, 0.0032; 0.5, 0.0052; 0.4, 0.0072; 0.7, 0.0064)