Transistor Sizing Considering Slew Information to Reduce Glitch Power in CMOS Digital Circuit Design

Hyungwoo Lee and Juho Kim
Department of Computer Science, Sogang University,
C.P.O. Box 1142, Seoul, Korea
Tel. +82-2-706-3997, Fax.: +82-2-704-8273
e-mail : hwlee@sogang.ac.kr, jhkim@ccs.sogang.ac.kr

Abstract: This paper presents the method of low power optimization considering the glitch reduction in CMOS circuits. Our algorithm utilizes the information of MOS size, the load capacitance of fan-out, and input slew to calculate the output waveform by using the linear signal model. Therefore, the accurate waveform of glitch can be obtained for estimation of power dissipation caused by glitches. Our algorithm is applied to ISCAS'85 benchmark circuits and experimental results show 23% glitch reduction and 11% total power reduction.

1. Introduction

Power considerations have become an increasingly dominant factor in the design of both portable and desktop system. An effective way to reduce power consumption is to lower dynamic power dissipation of a CMOS circuit. The average dynamic power consumed by a CMOS gate is given by $P_i = \frac{1}{2} \cdot C_i \cdot V_{dd}^2 \cdot f \cdot D(i)$ where C_i is the load capacitance, V_{dd} is the supply voltage, f is the clock frequency, and $D(i)$ is the transition density. Reducing load capacitance of the output node and transition density [1]-[4] are well-known techniques in low power optimization. The spurious signal transition that does not contribute to the function of a circuit is the glitch [1][5][6].

Our transistor-sizing algorithm considers slew information to reduce glitch power in CMOS digital circuit design. Since the glitch power can occupy 20% ~ 70% of total power dissipation [7], We can expect large amount of power reduction by reducing glitches.

The rest of this paper is organized as follows. Section 2 presents the glitch model used in this paper and sizing algorithm for glitch elimination. Section 3 demonstrates the experimental results from our implementation. Finally, section 4 concludes.

2. Glitch elimination by transistor sizing

To balance path delays in CMOS circuits, waveform of each output node must consider slew information of signal, load capacitance, and the size of PMOS and NMOS. We classify glitches first and eliminate them by controlling the size of PMOS and NMOS.

2.1 Glitch modeling using linear signal model

The data structure of PMOS and NMOS for a gate in the netlist is retained for gate sizing. The linear signal model achieves the accurate output waveform by considering MOS size, fan-out loading capacitance, and input slew. The linear signal model is described in Figure 1.

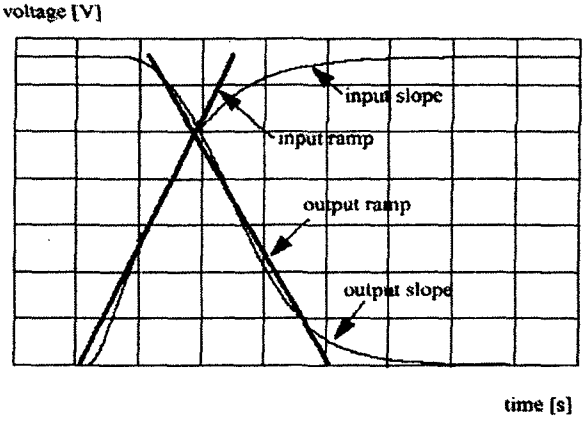


Figure 1. The linear signal model

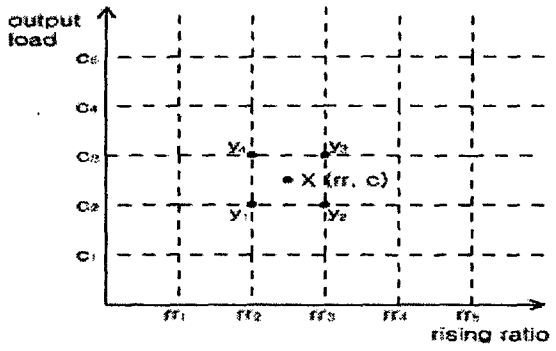
In order to calculate the exact power dissipation and the waveform of glitch, the SPICE simulation is applied to create a table prior to our event-driven simlation approach. If the value of delay and slew for signal is out of range, the delay and slew of an output can be calculated by applying the bilinear interpolation. Figure 2 shows the bilinear interpolation. Let's suppose that the delay or slew of point X is represented as following equations.

$$y_1 = y(rr_2, c_2)$$
$$y_2 = y(rr_3, c_2)$$
$$y_3 = y(rr_3, c_3)$$
$$y_4 = y(rr_2, c_3)$$

By using the bilinear interpolation, $y(rr,c)$ is represented as follows.

$$y(rr,c) = (1-u)(1-v)y_1 + u(1-v)y_2 uvy_3 + (1-u)vy_4$$

where $u = \frac{(rr - rr_2)}{(rr_3 - rr_2)}, v = \frac{(c - c_2)}{(c_3 - c_2)}$.



2.2 Sizing algorithm for glitch elimination

The method of gate sizing for glitch elimination is to control the size of PMOS and NMOS such that rising and falling time become the same value before reaching the logic threshold. First, only one of the rising and falling time at a gate is decreased if the slack of a selected gate is greater than 0. Then, a simple comparison is made for reduction of the power dissipation by glitch removal. Otherwise, the size of PMOS and NMOS are adjusted to retain the same area and at the same time the peak value of glitch does not reach logic threshold voltage. The Figure 2, Figure 3, and Figure 4 illustrate generated glitch condition and elimination method of generated glitch in the NAND gate, NOR gate, and INVERTER gate respectively. The PMOS and NMOS control rising and falling time of gate respectively. In the NAND gate, rising time is longer and falling time is shorter than original time. Through upsizing PMOS and downsizing NMOS, the peak value of glitch becomes smaller than the logic threshold. After all, glitch is eliminated by transistor sizing.

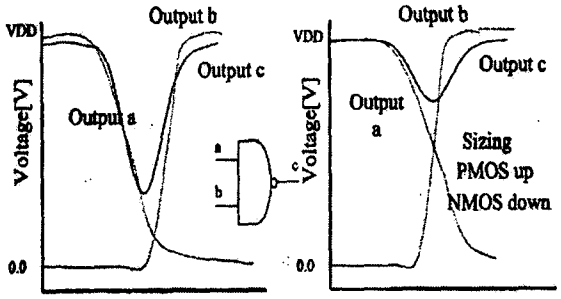


Figure 2. Generated glitch condition and elimination method in NAND gate

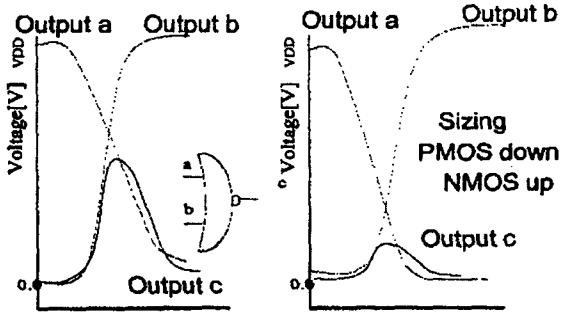


Figure 3. Generated glitch condition and elimination method in NOR gate

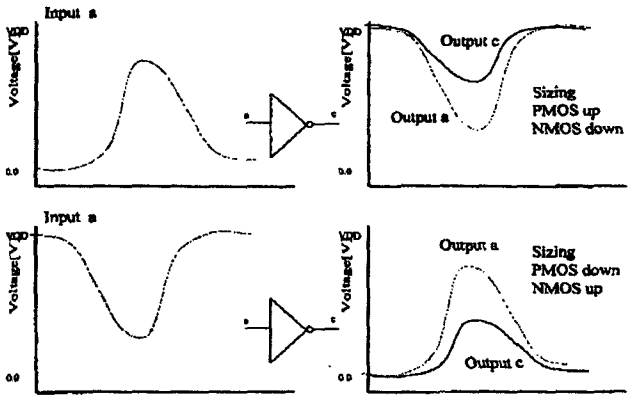


Figure 4. Generated glitch condition and elimination method in INVERTER gate

A straightforward method to estimate transition density is to simulate a circuit with arbitrary input vectors. In this case, the problem becomes how many input vectors have to be applied to achieve given accuracy level. Given a user-specified allowable percentage error ε and confidence level α, the approach described in [8], computes the number of input vectors with which to simulate the circuit. With $\alpha \times 100\%$ confidence, the number of input vectors L is represented as follows:

$$L \geq \left(\frac{erf^{-1}\left(\frac{\alpha}{2}\right) \times s}{\varepsilon \times \overline{p}} \right)^2$$

where $\overline{p}$ and s are the measured average and standard deviation of the power, and $erf^{-1}(\alpha/2)$ is the inverse error function obtained from the normal distribution. For a typical logic circuit and reasonable error and confidence level, the numbers of vectors is small, making this approach very efficient. Although this technique has some limitation that it only guarantees accuracy for the average switching activity over all gates, we employed it because of its fast run time and accurate glitch estimation.

Without slew information, glitch modeling at a gate-level is not accurate. In our algorithm, load capacitance and waveform of an output node is represented by signal delay information and slew. Our event-driven simulation algorithm can compute glitch sensitivity and selects candidate gate for resizing. During the sizing iteration, un-sizing function call is made if timing violations gets occurred.

The gate-sizing algorithm for glitch reduction is described in Figure 5. The overall gate-sizing algorithm for glitch reduction is described in Figure 6. Our algorithm simulates the circuit until sets of input vectors are applied.

```
1:   ResizeCandidate_Transistor( )
2:   {
3:      single_sizing_Transistor( );  /* PMOS and NMOS
                                        sizing of gate */
4:      small_Simulate( );  /* 3-level simulation of input
                               node */
5:      if ( nogain )
6:     double_sizing_Transistor( );
/* sizing of gate input node considering path balancing */
```

Figure 5. The gate sizing algorithm for glitch reduction

```
1:   Overall_ReduceGlitch( )
2:   {
3:     for each input_vector {
4:       repeat {
5:                do simulate_circuit( );
6:                do power_estimation( );
7:                resize candidate_Transistor( );
8:                do timing_analysis( );
9:       if(no_timing_violation){
10:                compute power( );
11:                compute glitch_reduction( );
12:                compute power_reduction( );
13:                 }else if (timing_violation)
14:                      unSizng( );
15:     }until (glitch_reduction and power_reduction are
                 positive);
16:    }
17:  }
```

Figure 6. The overall gate sizing algorithm for glitch reduction

3. Experimental results

Our algorithm is applied to ISCAS'85 benchmark circuit and the result shows the 23% glitch power reduction and 11% total power reduction. The experimental results are presented in Table 1 and Table 2. The glitch power section of Table 1 shows the glitch power results when optimization process was not performed. After sizing, the glitch power section illustrates optimization result. Table 2 shows the effect of glitch power reduction in total power dissipation of the circuit.

Table 1. Glitch reduction results

Circuit name	Glitch Power	After sizing Glitch Power	Glitch reduction
C432	0.75	0.44	42%
C2670	2.39	1.9	20%
C7552	8.77	7.41	15%
i9	3.29	2.95	10%
rot	1.63	1.01	37%
dalu	3.59	3.02	15%
Average			23.17%

Table 2. Total power reduction results

Circuit name	Total Power	After sizing Glitch Power	Glitch reduction
C432	1.48	1.18	20%
C2670	6.28	4.95	21%
C7552	21.73	20.24	7%
i9	5.07	4.75	6%
rot	5.02	4.68	7%
dalu	13.12	12.38	6%
Average			11.17%

4. Conclusions

In this paper, a glitch elimination algorithm by transistor sizing has been proposed. Our method of gate sizing for glitch reduction is to control the size of PMOS and NMOS in order to remove glitches at gates. With our algorithm, an average of 23% glitch reduction and 11% power reduction were achieved.

Acknowledgement

This work was supported by a grant No.(20014014) from Korea Science & Engineering Foundation.
This research was sponsored in part by Institute for Applied Science and Technology of Sogang University.

References

[1] Manjit Borah, Robert Michael Owens and Mary Jane Irwin, "Transistor sizing for minimizing power consumption of CMOS circuits under delay constraint," in Proceedings of the International Symposium on Low Power Design, pp. 167-172, April 1995.

[2] M. Sarrafzad and D. Chen, "An Exact Algorithm for low power library-specific gate re-sizing," in Proc. 33rd Design Automation Conference, pp. 783-788, June 1996.

[3] F. Najm, "Transition Density, A New Measure of Activity in Digital Circuits", IEEE Transaction on Computer-Aided Design, Vol. 10, no. 4, pp. 310-323, February 1993.

[4] H. Mehta, M. Borah, R. M. Owens and M. J. Irwin, "Accurate Estimation of Combinational Circuit Activity," in Proceedings of the 32nd IEEE/ACM Design Automation Conference, pp. 618-622, June 1995.

[5] M. Hashimoto, H. Onodera, and K. Tamaru, "A Power Optimization Method considering Glitch Reduction by Gate Sizing," in Proceedings of the International Symposium on Low Power Design, pp. 221-226, August 1998.

[6] Sachin S.Sapatnekar and Weitong Chuang, "Power vs Delay in Gate Sizing: conflicting Objectives?," in Proceedings of the 1995 IEEE/ACM International conference of Computer-Aided Design, pp. 463-466, 1995.

[7] A. Shen, A. Ghosh, S. Devadas and K. Kreutzer, "On Average Power Dissipation and Random Pattern Testability of CMOS Combinational Logic Networks," in Proc. Int'l Conference on Computer Aided Design, pp. 402-407, November 1992

[8] R. Burch, F. Najm, and T. Trick, "McPOWER: A Monte Carlo approach to power estimation, "in Proceedings of the 1992 IEEE/ACM international Conference on Computer Aided Design, pp. 90-97, November 1992.