Variable Input Delay CMOS Logic for Low Power Design

Tezaswi Raja
Transmeta Corp.
Santa Clara, CA 95054, USA
tezaswir@caip.rutgers.edu

Vishwani D. Agrawal
Auburn University, Dept. of ECE
Auburn, AL 36849, USA
vagrawal@eng.auburn.edu

Michael L. Bushnell
Rutgers University, Dept. of ECE
Piscataway, NJ 08854, USA
bushnell@caip.rutgers.edu

Abstract

Modern digital circuits consist of logic gates implemented in the complementary metal oxide semiconductor (CMOS) technology. The time taken for a logic gate output to change after one or more inputs have changed is called the output delay *of the gate. A conventional CMOS gate is designed to have the same input to output delay irrespective of which input caused the output to change. We propose a new gate design that has different delays along various input to output paths within the gate. This is obtained by differently sizing "permanently on" series transistors at the inputs of the logic gate. We demonstrate the use of the variable input delay CMOS gates for a totally glitch-free minimum dynamic power implementation of a digital circuit. Applying a previously described linear programming method to the c7552 benchmark circuit, we obtained a power saving of 58% over an unoptimized design. This power consumption was 18% lower than another low power design that used the conventional CMOS gates. All circuits had the same overall delay. Since the overall delay was not allowed to increase, the glitch elimination with conventional gates required insertion of delay buffers on non-critical paths. The use of the variable input delay gates drastically reduced the required number of delay buffers.*

1 Introduction

There are many ways of combining the transistors to perform the logic functions such as NOT, NAND and NOR. We will describe the CMOS design style which is the most prominent in current day technologies.

CMOS gates are constructed by a combination of MOSFETs to realize a logic function. But a MOSFET is not an ideal switch. When *open* it provides a large but finite resistance between its source and drain terminals. When *closed* it provides a small non-zero resistance. For a CMOS gate the output signal change follows the input change with a certain delay. First, the closing and opening of MOSFETs in the gate depends upon the slope of input signals. Then, the output signal change requires charging or discharging of the output capacitance through a low resistance path provided by the "on" MOSFETs.

Gate Delay is the time taken for the output signal at the output of the gate to reach 50% of *Vdd* after the signal at the input of the gate has reached 50% of *Vdd*.

Gate delay is a function of the amount of resistance and capacitance in the current path. A MOSFET when closed offers a finite resistance R_{on} that is a function of the width and length of the device. Since gate delay is given by $R_{on} \times C_L$(where C_L is the load capacitance) the gate delay can also be varied by changing the width and length of the transistor [13, 15]. For example, a NAND gate output rises due to current flow in its pFETs. Hence, the delay of the NAND gate for a rising transition, can be altered by changing the sizes of pFETs. To increase the delay, we increase the resistance of the transistor by increasing the length of the transistor. Similarly the output delay for a falling transition can be varied by changing the length of the nFETs. The delay can also be reduced by increasing the width of the transistor. The delay can be effectively changed by manipulating the width and length of the transistors in the gate. Note that it is possible only to manipulate the overall delay of the gate but not the individual delays along the different paths. For instance, the delay of the gate when input 1 transitions cannot be independently controlled without altering the delay when input 2 transitions. The delays are inter-related. This is a drawback in some applications as described in the next section.

1.1 Application to Low Power Circuits

Every signal transition consumes a finite amount of energy. For the correct functioning of the logic circuit, every signal net needs to transition at most one time in one clock cycle. But in reality, the gate outputs transition more than once and these unnecessary transitions are called *glitches*. These transitions consume energy and are quite unnecessary for the correct functioning of the circuit. Glitch power consumption can be up to 40% of the overall power consumption and it is advantageous to eliminate the glitches in the circuit as power consumption is critical in today's chips. Glitches arise due to the differences in the arrival times of signal transitions at the inputs of the gate.

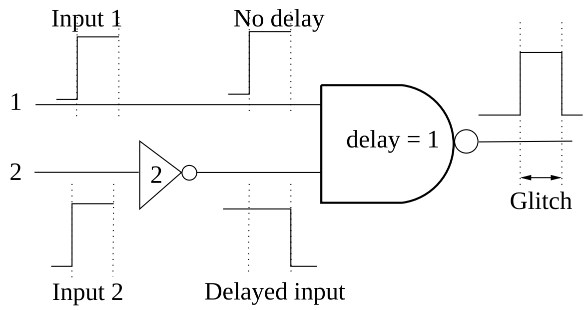

Figure 1: A circuit showing the formation of glitches. The inverter has a delay of 2 and the NAND gate has a delay 1. Due to differing arrival times at the inputs of the NAND gate, the output produces a glitch consisting of two transitions.

Differential Delay is the maximum difference in the signal arrival times at the different inputs of a multi-input gate.

Consider the circuit shown in Figure 1. The signal arrival at the lower input of the NAND gate is always 2 time units later than the signal arrival at the upper input due to the inverter in its path. Thus the differential delay at the NAND gate is 2. This differential delay makes the output of NAND gate transition twice when in reality, it should not have transitioned at all logically. These extra transitions are the glitches and they waste energy.

There have been many techniques proposed to eliminate the glitches. In *delay balancing*, the inputs are made to arrive at the same time by inserting extra delay buffers on selected paths [10, 11, 9, 20, 16]. In *hazard filtering*, the gate delay is made greater than the differential delay at the inputs of the gate to filter the glitch [1]. In *gate sizing*, every gate is assumed to be an equivalent inverter [5, 6, 8, 7, 3, 4]. *Transistor sizing* treats every transistor's size as a variable and tries to find a glitch-free design [14, 22, 12, 24, 23, 21]. However, these techniques are either greedy approaches or have non-linear convergence problems [17].

Some techniques use *linear programming* where the gate delays are treated as variables and the optimum delays are found by solving the *linear program* (LP) [2, 17, 18]. The problem with this technique is that it inserts delay buffers in the circuit. These extra inserted elements also consume power themselves and hence reduce the achievable power savings. In all of the above techniques, the problem of buffer insertion arises due to the conventional gate design. The conventional CMOS gates have a single delay, no matter which input of the gate causes the transition. This restricts the slack allowed to the LP in optimizing the delays. A new technique by Raja *et al.* proposed a LP technique using a new gate delay model, where the input delay of the gate can also be varied [19]. This makes the gate delay different for different input-output paths through the gate. The advantage of this gate model is that the glitches can be completely eliminated in the circuit without the insertion of any delay buffers, thus achieving more power savings. This technique is the focus of this paper.

2 Proposed Gate Design

As described in the earlier section, it is advantageous to design a gate with differing delays along different input-output paths of the gate. We define such a gate as an *variable input-delay gate*. In this section, we propose a transistor level implementation of such a gate and its characteristics. Consider a two-input NAND gate shown in Figure 2. Suppose, the delay required along the path 1-3 is 2 units and 2-3 is 1 unit.

$$\begin{aligned} d_{1\rightarrow 3} &= R_{on} \times C_{in1} + d_3 \\ d_{2\rightarrow 3} &= R_{on} \times C_{in2} + d_3 \end{aligned}$$

where C_{in} is the input gate capacitance seen at the inputs of the gate and R_{on} is the series resistance of the ON transistor in the previous stage. To control the delay along the different paths we propose four different implementations.

- **Input capacitance manipulation** is the technique by which C_{in1} is increased without altering C_{in2}. This is achieved by increasing the sizes of the transistors connected to input 1 such that $C_{in1} > C_{in2}$. Now the delays along the different paths are:

  $$\begin{aligned} d_{1\rightarrow 3} &= R_{on} \times C_{in1} + d_3 \\ d_{2\rightarrow 3} &= R_{on} \times C_{in2} + d_3 \\ d_{1\rightarrow 3} &> d_{2\rightarrow 3} \end{aligned}$$

  The problem with this implementation is that the ON resistances of the transistors in the series path are interrelated and hence the output delay is also altered. The formulation becomes non-linear.

- **Resistance with a single *n*MOS pass transistor** can be added in series to the path in which extra delay is desired. This schematic is shown in Figure 2(b). This nMOS transistor

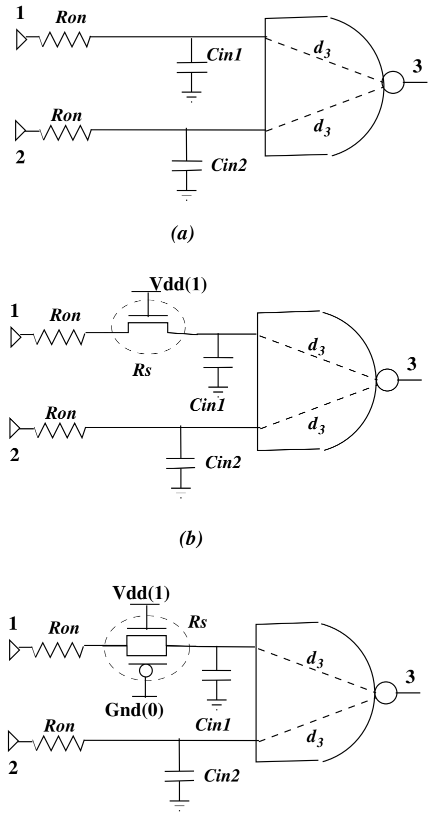

Figure 2: Schematic of the proposed variable input delay gate. (a) A simple 2 input NAND gate with variable delay achieved by input capacitance manipulation, (b) Input delay is manipulated by the introduction of series resistance R_s through a nMOS pass transistor and (c) through a CMOS pass transistor always ON.

is always ON and hence adds a series resistance R_s to the path $1 \rightarrow 3$. Now the delays are:

$$\begin{aligned} d_{1\rightarrow 3} &= R_{on} \times C_{in1} \ + \ d_3 \\ d_{2\rightarrow 3} &= (R_{on} + R_s) \times C_{in2} \ + \ d_3 \\ d_{1\rightarrow 3} &> d_{2\rightarrow 3} \end{aligned}$$

The resistance R_s can be controlled by increasing the size of the nMOS transistor. The delay along the path $1 \rightarrow 3$ can be controlled independent of the delay along path $2 \rightarrow 3$. Hence the gate has different delays along different input-output paths through it, which was the objective. The disadvantage of this design is that the nMOS pass transistor degrades the signal when it passes a logic 1. This causes the transistors in the next stage to have a higher leakage current. This aspect is discussed in more detail in the next section.

- **Resistance with a CMOS pass transistor** can be added to introduce the extra resistance in the path as shown in Figure 2(c). The principle is the same as adding a single nFET, but the CMOS pass transistor contains both nMOS and pMOS transistors that are always ON. This does not degrade the signal but has the disadvantage that it adds two extra transistors.

- **Resistance with a feedthrough resistive cell** is a technique of adding the resistance using a *polysilicon* serpentine resistor overlaid with *silicide block*. This is the standard way of creating a resistance in analog layout design but can be used for this purpose. The advantage of using these cells is the continuous controllability of resistance rather than the discrete control provided by transistors.

2.1 Design Issues

There are certain design issues regarding the new gate design. The delay along the path can be changed by changing the series resistance. R_s is a function of the length of the transistor/transmission gate and hence the delay along the line can be altered by changing the length of the extra transistor/transmission gate.

- This transistor cannot be infinitely large as this would increase the voltage drop across the transistor and cause signal integrity issues at the output of the gate. Hence there is a realistic limit to the length of the transistor added and this determines the maximum differential delay that can be added. Raja *et al.* describe this as the *gate differential delay upper bound* u_b in their theoretical analysis [19]. This parameter u_b is related to the technology in which the gate will be implemented in and hence is called the *feasibility condition*. Our calculations have predicted and measured a u_b of 10 units for the 0.25μm fabrication process.

- If the voltage drop across the transistor is too large, it does not drive the gate transistors in the fanout into cut-off. This increases the leakage from the supply to the ground through the gate transistors as both of them are not completely off. This problem can be alleviated by using a CMOS transmission gate instead of a single transistor. The effect of increased leakage is shown in the results section.

- The placement of the series transistor with respect to the routing capacitance also needs to be considered. If the routing capacitance is small it does not matter where the transistor is

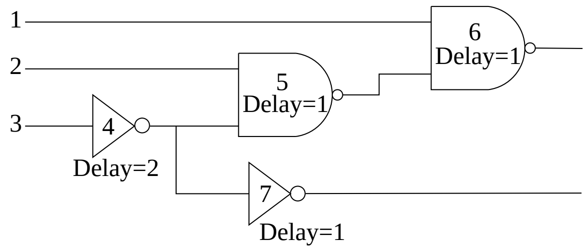

Figure 3: An example circuit.

placed in the path. But if the routing capacitance is large, then the delay at the input of the gate changes as the transistor is moved along the path as it sees a different capacitance at every stage. We have inserted the transistor at the end of the routing path in our designs.

3 Results

In this section we present an application of the new gate design in implementing custom circuits for minimum dynamic power.

An Unoptimized Example Circuit. Consider the simple example circuit of Figure 3. Assume that the delays along every gate are the minimum allowed by the technology. We observe that the differential delay, at gates 5 and 6, exceeds the inertial delay and we expect these gates to glitch. The circuit was simulated for rising signals at all three inputs. The simulation was done using the *Spectre* analog simulator from *Cadence*. As expected, gates 5 and 6 transition 2 and 3 times, respectively. These are the glitches we wish to eliminate in the following design.

Buffer Optimized Circuit. The buffer optimization using conventional gate design requires the use of one buffer for the circuit to operate at the same speed. The optimized circuit with the buffer is shown in Figure 4. The simulation was done using the *Spectre* analog simulator from *Cadence*. The buffer optimized circuit was simulated for the same vector-pair as the unoptimized circuit. As expected, the buffer optimization has eliminated the glitches.

Low-Power Design with Proposed Gate. Instead of the buffer, the variable input-delay gate is designed and the optimized circuit is shown in Figure 5. We have used the single nMOS transistor implementation in this but any of the proposed designs could have been used. The simulation was done using the *Spectre* analog simulator from *Cadence*. This circuit was simulated for the same vector-pair as the previous circuits. The optimized circuit with the proposed gate design also eliminates the glitches effectively.

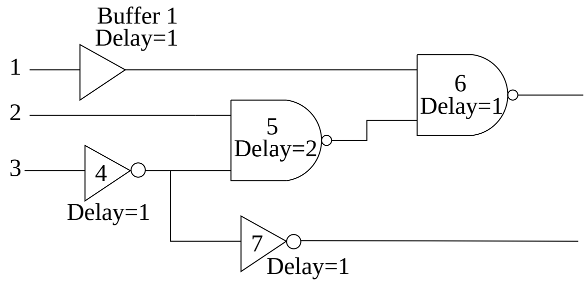

Figure 4: Optimized example circuit with buffer insertion.

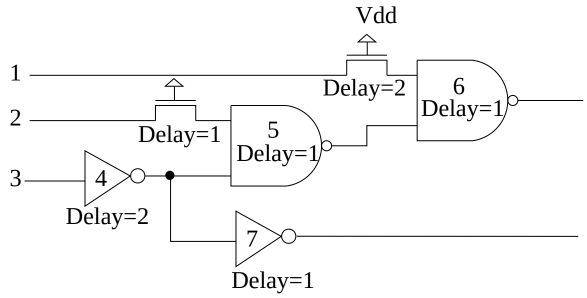

Figure 5: Optimized example circuit with the proposed gate.

3.1 Energy Consumption Results

The energy consumption of a circuit is a function of the current drawn by it from the supply. During our simulations for the three circuits described above, we measured the input current for the given input vector and computed the energy. The energy consumption results are shown in Table 3.1. The simulations were done with *Spectre* analog simulator from *Cadence*. As seen in the figure, the unoptimized circuit consumes 800fJ, the buffer optimized circuit consumes 550fJ and the new gate optimized circuit consumes 300fJ. Thus the energy savings of the new design are 62.5% with respect to the unoptimized circuit. The new gate design achieves 36.8% more savings than the buffer optimized design with respect to the unoptimized circuit.

3.2 Leakage current analysis

The introduction of a single nMOS transistor degrades the signal at the gates of the transistors. This increases the leakage current of the circuit and may even drive the transistors out of cut-off. The current flowing in steady state is called the *quiescent current* and is due to the leakage through the different transistors. The quiescent current is a function of the input vectors at the PIs of the circuit. To analyze the relative effect, we have simulated the above circuit with two different input

Table 1: Power comparison of the three designs of the example circuit.

vectors and let the circuit settle for a long time. Three circuits were considered in the simulation: an unoptimized circuit, an optimized circuit with single nMOS transistors used to implement the variable input delay gates and the third with CMOS transmission gates used to implement the variable input delay gates. The leakage currents for the vector 000 show no change for the three circuits as in this state the nMOS transistors are passing logic 0 which is not degraded. For vector 111, however, there is an increase of 0.45% in leakage current due to the nMOS transistor addition. The circuit with the CMOS transmission gates had an increase of only 0.2%. This increase is not due to the degradation in value but due to the leakage path added from Vdd to Gnd through the sidewall capacitance. This is a very minor increase for the current 0.25μm fabrication technology but further analysis needs to be done for more recent technologies.

3.3 A Larger Circuit

We have tried the optimization procedure on an ISCAS'85 benchmark circuit, c7552. The circuit contained 3,827 gates and approximately 40,000 transistors. The unoptimized circuit was designed at the physical layout level using the *Cadence layout editor*. Using the buffer insertion technique, the circuit optimization needed 366 buffers to maintain the same speed of operation [18]. We redesigned the circuit using the proposed variable input delay gates and the design contained needed no extra buffers. The new design functions at the same speed of operation as the unoptimized version. The results of instantaneous and average power measurements are shown in Figures 6 and 7. The circuits were designed at the physical level and the load capacitances extracted. An event-driven delay simulator was used for the energy estimation, with the loading capacitances taken from the physical design. The circuits were simulated on the same set of 156 compacted ATPG test vectors.

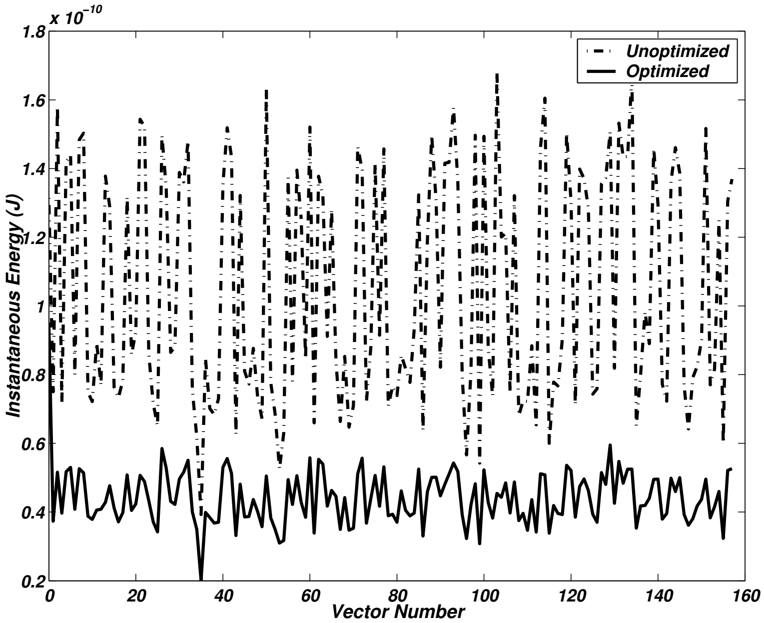

Figure 6: Instantaneous energy consumption in benchmark circuit c7552 for 156 vectors. Results show a peak power savings of 68% over the unoptimized circuit.

Leakage currents were measured using *Spectre* analog simulator and added to the final results, to get an accurate estimate of the energy consumption.

The results show a peak power savings of up-to 68% and average power savings of up to 58%. The proposed technique using variable input delay gates consumed 18% less power compared to the buffer optimized circuit. The technique is thus scalable to larger circuits as well.

4 Conclusion

We have proposed a new variable input delay gate design, which has different delays along different input-output paths through the gate. This new design has applications to low power design of digital CMOS circuits. Using the new gate design, energy savings of up to 62% have been achieved. The delays of the gates tend to change due to process variations such as temperature, fabrication impurities, etc. These variations make the delay vary over a range rather than being a single static number. This can be accounted for in our technique during the LP stage, where the constraints can be slightly modified to incorporate the maximum gate delay value in the latest time of arrival constraints and the minimum gate delay value in the earliest time arrival constraints.

The future work includes designing larger circuits using this technique and also applying the

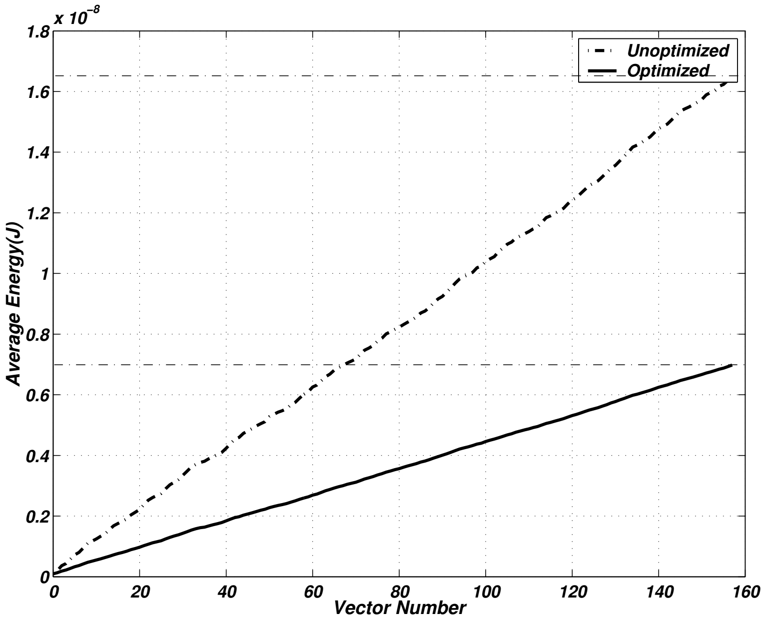

Figure 7: Average energy consumption results for benchmark circuit c7552 for 156 vectors. Results show an average energy savings of 58%

technique to newer fabrication technologies.

References

[1] V. D. Agrawal, "Low Power Design by Hazard Filtering," in *Proc. of the International Conference on VLSI Design*, Jan. 1997, pp. 193–197.

[2] V. D. Agrawal, M. L. Bushnell, G. Parthasarathy, and R. Ramadoss, "Digital Circuit Design for Mimimum Transient Energy and a Linear Programming Method," in *Proc. of the International Conference on VLSI Design*, Jan. 1999, pp. 434–439.

[3] M. Berkelaar, "Statistical Delay Calculation," in *Workshop notes of the International Workshop on Logic Synthesis*, (Lake Tahoe, CA), May 1997, pp. 2.1.1–2.1.4.

[4] M. Berkelaar, "Statistical Delay Calculation: A Linear Time Method," in *Proc. of the IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, Dec. 1997, pp. 15–24.

[5] M. Berkelaar, P. Buurman, and J. Jess, "Computing Entire Area/Power Consumption versus Delay Trade-off Curve for Gate Sizing Using a Piecewise Linear Simulator," *IEEE Transactions on Circuits and Systems*, vol. 15, no. 11, pp. 1424–1434, Nov. 1996.

[6] M. Berkelaar and E. Jacobs, "Using Gate Sizing to Reduce Glitch Power," in *Proc. of the ProRISC Workshop on Circuits, Systems and Signal Processing*, (Mierlo, The Netherlands), Nov. 1996, pp. 183–188.

[7] M. Berkelaar and E. T. A. F. Jacobs, "Gate Sizing Using a Statistical Delay Model," in *Proc. of the Design Automation and Test in Europe Conference*, (Paris, France), Mar. 2000, pp. 283–290.

[8] M. Berkelaar and J. A. G. Jess, "Transistor Sizing in MOS Digital Circuits with Linear Programming," in *Proc. of the European Design Automation Conference*, (Mierlo, The Netherlands), Mar. 1990, pp. 217–221.

[9] A. Chandrakasan and R. Brodersen, editors, *Low-Power CMOS Design*. New York: IEEE Press, 1998.

[10] A. P. Chandrakasan and R. W. Brodersen, *Low Power Digital CMOS Design*. Boston: Kluwer Academic Publishers, 1995.

[11] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low Power CMOS Digital Design," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 4, pp. 473–484, Apr. 1992.

[12] S. Datta, S. Nag, and K. Roy, "ASAP: A Transistor Sizing Tool for Area, Delay and Power Optimization of CMOS Circuits," in *Proc. of the IEEE International Symposium on Circuits and Systems*, May 1994, pp. 61–64.

[13] W. C. Elmore, "The Transient Response of Damped Linear Networks with Particular Regard to Wideband Amplifiers," *J. of Applied Physics*, vol. 19, no. 1, pp. 55–63, Jan. 1948.

[14] J. P. Fishburn and A. E. Dunlop, "TILOS: A Posynomial Approach to Transistor Sizing," in *Proc. of the International Conference on Computer-Aided Design*, Nov. 1985, pp. 326–328.

[15] J. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*. Upper Saddle River, NJ: Prentice Hall, 2003.

[16] J. M. Rabaey and M. Pedram, *Low Power Design Methodologies*. Boston: Kluwer Academic Publishers, 1995.

[17] T. Raja, "A Reduced Constraint Set Linear Program for Low Power Design of Digital Circuits," Master's thesis, Dept. of ECE, Rutgers Univ., New Jersey, USA, 2002.

[18] T. Raja, V. D. Agrawal, and M. L. Bushnell, "Minimum Dynamic Power CMOS Circuit Design by a Reduced Constraint Set Linear Program," in *Proc. of the International Conference on VLSI Design*, Jan. 2003, pp. 527–532.

[19] T. Raja, V. D. Agrawal, and M. L. Bushnell, "CMOS Circuit Design for Minimum Dynamic Power and Highest Speed," in *Proc. of the International Conference on VLSI Design*, Jan. 2004, pp. 1035–1040.

[20] K. Roy and S. C. Prasad, *Low-Power CMOS VLSI Circuit Design*. New York: Wiley Interscience Publication, 2000.

[21] C. V. Schimpfle, A. Wroblewski, and J. A. Nassek, "Transistor Sizing for Switching Activity Reduction in Digital Circuits," in *Proc. of the European Conference on Circuit Theory and Design*, volume 1, Aug. 1999, pp. 114–117.

[22] V. Sundararajan, S. Sapatnekar, and K. Parhi, "Fast and Exact Transistor Sizing Based on Iterative Relaxation," *IEEE Transactions on Computer Aided Design of Circuits and Systems*, vol. 21, no. 5, pp. 568–581, May 2002.

[23] A. Wroblewski, C. V. Schimpfle, and J. A. Nassek, "Automated Transistor Sizing Algorithm for Minimizing Spurious Switching Activities in CMOS Circuits," in *Proc. of the IEEE International Symposium on Circuits and Systems*, May 2000, pp. 291–294.

[24] A. Wroblewski, O. Schumacher, C. V. Schimpfle, and J. A. Nassek, "Minimizing Gate Capacitances with Transistor Sizing," in *Proc. of the IEEE International Symposium on Circuits and Systems*, May 2001, pp. 186–189.