

## AN OPTIMIZATION-BASED MULTIPLE-VOLTAGE SCALING TECHNIQUE FOR LOW-POWER CMOS DIGITAL DESIGN

YI-JONG YEH and SY-YEN KUO

*Department of Electrical Engineering,  
National Taiwan University, Taipei, Taiwan*

Received 12 April 2002

Accepted 27 June 2002

In this paper, we propose a voltage scaling technique with multiple supply voltages for low-power designs. We adopt the path sensitization technique and release the clustering constraint used by the previous works. Our technique first operates the gates with the lowest feasible supply voltages and then uses an existing path selection algorithm for optimization. Experiments are conducted on all the ISCAS85 benchmarks and the results show that significant power can be further reduced by our technique in comparison with the previous works. Furthermore, the results generated by our technique are close to the optimal values.

*Keywords:* Low-power; voltage scaling; power minimization; optimization-based.

### 1. Introduction

Power consumption is one of the most significant parameters in VLSI designs. In a CMOS digital circuit, power consumption is dominated by dynamic power, which is proportional to the square of the supply voltage. As a result voltage scaling is evidently an effective technique in power reduction and was employed by many researchers.

Chandrakasan *et al.* provided us a simple rule in power reduction, i.e., operate a circuit as slowly as possible with the lowest possible supply voltage.<sup>1</sup> The most popular voltage scaling technique is to operate all the gates in a circuit with a reduced supply voltage that is limited by the critical paths. However, the gates that are not on critical paths could operate slower with lower supply voltages. Consequently two or more supply voltages were employed in previous works.

Power consumption was reduced with multiple supply voltages at function level, where the effect of interconnections between entities with different supply voltages was insignificant and could be ignored.<sup>2–4</sup> At gate level, the power consumption was reduced with two supply voltages, where level converters were inserted to prevent the static current when the gates with lower supply voltages drive the gates with higher supply voltages.<sup>5,6</sup> Furthermore, the power consumption was reduced with multiple supply voltages at gate level and no level converters are necessary.<sup>7</sup>

To reduce the complexity of physical layout with multiple supply voltages, gates of the same supply voltage are clustered at circuit topology. However, gate clustering can be done at the early phase of physical layout. Therefore, in this paper we release the clustering constraint employed by the previous works and propose a multiple-voltage scaling technique to significantly reduce the power consumption at gate level.

The rest of this paper is organized as follows. In Sec. 2, we give some definitions and notations which will be used throughout this paper. Next, we formulate the power minimization problem and propose an algorithm to solve this problem in Sec. 3. Section 4 shows the experiment results of our algorithm on the ISCAS85 benchmarks. Finally, concluding remarks are given in Sec. 5.

## 2. Definitions and Terminologies

**Definition 1 (Path).** A path  $P = (G_0, f_0, G_1, f_1, \dots, f_{m-1}, G_m)$  in a combinational circuit is an alternating sequence of wires and gates.

**Definition 2 (On-Input).** Wire  $f_i$ ,  $0 \leq i \leq m - 1$ , is called an on-input of  $P$  which connects gate  $G_i$  to gate  $G_{i+1}$ .

**Definition 3 (Primary Input Vector).** A primary input vector is a vector of logic values at all the primary inputs.

**Definition 4 (Dominate).** Wire  $f$ , which is connected to gate  $G$ , is considered to dominate  $G$  if the stable value and the stable time of  $G$  are determined by those of  $f$ .

**Definition 5 (Activate).** A path is activated by a primary input vector if each on-input of the path dominates its connected gate when the input vector is applied.

**Definition 6 (Sensitizable Path).** A path which can be activated by at least one primary input vector is defined as a sensitizable path.

**Definition 7 (False Path).** A path which will never be activated by any primary input vector is called a false path.

**Definition 8 (Critical Path).** The critical paths are the longest sensitizable paths in a circuit.

**Definition 9 (Fanin Set).** The fanin set of gate  $G$ , denoted by  $\Gamma^-(G)$ , is a set of gates which connect to the inputs of  $G$ .

**Definition 10 (Fanout Set).** The fanout set of gate  $G$ , denoted by  $\Gamma^+(G)$ , is a set of gates which the output of  $G$  connects to.

**Definition 11 (Stable Time).** The time when the output of gate  $G$  becomes stable is called the stable time of gate  $G$ , denoted as  $T_s(G)$ . Given the stable time at each primary input, the stable time of gate  $G$  can be obtained by  $T_s(G) = \text{delay}(G) + \max_{v \in \Gamma^-(G)} T_s(v)$ .

**Definition 12 (Required Time).** The required time of gate  $G$ , denoted as  $T_r(G)$ , is the latest time when the output of gate  $G$  has to be stable to meet the timing constraint of the circuit. Given the required time at each primary output, the required time of gate  $G$  can be obtained by  $T_r(G) = \min_{v \in \Gamma^+(G)} (T_r(v) - \text{delay}(v))$ .

**Definition 13 (Slack).** The slack of gate  $G$ , denoted as  $\text{slack}(G)$ , is the maximum delay increase which gate  $G$  may have under the timing constraint. When the stable time and the required time of gate  $G$  are known, the slack of gate  $G$  can be obtained by  $\text{slack}(G) = T_r(G) - T_s(G)$ .

### 3. Optimization-Based Multiple-Voltage Scaling

After releasing the clustering constraint used by the previous works, we can formulate the power minimization problem as:

*Given a combinational circuit with a timing constraint and a set of supply voltages, assign the supply voltages to the gates in the circuit to minimize the total power consumption of the circuit.*

In the following subsections, we first describe the basic idea of the optimization-based multiple-voltage scaling (OBMVS) technique. Next, we give a brief introduction to the path selection algorithms which play an important role in the OBMVS technique. Then, we propose the OBMVS algorithm followed by an illustration example.

#### 3.1. Basic idea

The basic idea of previous works<sup>5-7</sup> applies a depth-first search algorithm and then tries to scale down the supply voltage of every visited gates. Such method obviously lacks a global view and could obtain only a local optimal solution.

The basic idea of our algorithm is to operate the gates with the lowest feasible supply voltages according to their slacks. Such voltage assignment evidently obtains the lower bound of the power consumption and the delay of the circuit may be more than the given timing constraint. Therefore, a path selection algorithm is applied to select a set of long paths for performance optimization. According to the selected long paths, we can determine the critical ordering of the gates. Based on the critical ordering, we increase the supply voltages of the gates in order until the delays of all selected long paths are no more than the given timing constraint.

### 3.2. Path selection algorithms

The actual delay of a combinational circuit is defined as the delay of its longest sensitizable paths instead of that of its longest paths. Therefore, it is pessimistic to reduce the delays of all long paths in a circuit for performance optimization without taking path sensitization into account. Here a long path means that its delay is larger than the timing constraint of the circuit.

Several path sensitization criteria have been proposed to estimate the delay of a circuit including the *exact criterion*, the *loose criterion*, the *BMCD criterion*,<sup>8</sup> the *DYG criterion*,<sup>9</sup> the *PCD criterion*,<sup>10</sup> the *viable criterion*,<sup>11</sup> the *BI criterion*,<sup>12</sup> and the *dynamic criterion*.<sup>13</sup> From the timing verification point of view, a path sensitization criterion is considered to be “correct” if the estimated circuit delay is never shorter than the actual delay of the circuit. Certainly, a criterion is more accurate if the estimated delay is closer to the actual delay of the circuit.

The objective of path selection algorithms is to select a set of paths for performance optimization techniques. The cost of performance optimization usually depends on the number of long paths selected to be shortened. In general, the more long paths need to be shortened, the more expensive the optimization will be. As a consequence the number of selected paths should be as small as possible.

As illustrated in previous works, most long paths in a complex circuit are actually false. Furthermore, a significant portion of long false paths do not need to be shortened.<sup>14</sup> We may need only to shorten long sensitizable paths in order to meet the timing constraint. However, when all the long sensitizable paths are shortened, a long false path may become sensitizable. On the other hand, some long sensitizable paths may not need to be selected for optimization. These problems were tackled and two selection algorithms, vector-oriented and path-oriented, were proposed.<sup>14</sup> For a circuit with many primary inputs, the vector-oriented algorithm may not be feasible since there are too many input combinations. Consequently, the proposed path-oriented selection algorithm is adopted in the OBMVS algorithm.

### 3.3. The OBMVS algorithm

According to the basic idea, we propose the OBMVS algorithm in Fig. 1. The given supply voltages are arranged in descending order and are labeled  $V_{dd0}$ ,  $V_{dd1}, \dots, V_{dd(n-1)}$  if the number of given supply voltages is  $n$ . Lines 1–3 reset the credits of all gates and operate the gates with their lowest feasible supply voltages. *Credit* is used to represent the critical order of gates. Line 4 calls the path selection algorithm, `POSA_FeasibleSet`,<sup>14</sup> to obtain a set of long paths,  $FS$ , for optimization. Lines 5–8 set the credits of the gates based on the selected paths in  $FS$ . Next, the gates with positive credits are inserted into a priority queue,  $PQ$ , in line 9. The priority queue arranges a data structure such that the gate with the maximum credit can be easily retrieved. Lines 10–18 optimize the circuit by increasing the supply voltages of the most critical gates until the timing constraint is met.

---

```

OBMVS()
1  For (each gate  $G$  of the circuit) Do
2    credit( $G$ ) = 0;
3    Set the voltage of  $G$  to the lowest  $V_{ddi}$  such that
    delay( $G, V_{ddi}$ ) - delay( $G, V_{dd0}$ ) <= slack( $G$ );
4   $FS = \text{POSA\_FeasibleSet}()$ ;
5  For (each path  $P$  in  $FS$ ) Do
6    For (each gate  $G$  in  $P$ ) Do
7      If (voltage of  $G \neq V_{dd0}$ ) Then
8        credit( $G$ ) ++;
9  Insert the gates with positive credits to a priority queue,  $PQ$ ;
10 While ( $FS \neq \phi$ ) Do
11   Retrieve a gate  $G$  from the top of  $PQ$ ;
12   Increase the voltage of  $G$ ;
13   If (the voltage of  $G \neq V_{dd0}$ ) Then
14     Insert  $G$  back to  $PQ$ ;
15   For (each path  $P$  in  $FS$ ) Do
16     If (delay( $P$ ) <= timing constraint) Then
17       Decrease the credit of each gate in  $P$ ;
18       Delete  $P$  from  $FS$ ;

```

---

Fig. 1. The optimization-based algorithm for multiple-voltage scaling.

### 3.4. Example

Take Fig. 2 as an example and assume that

- the given supply voltages are 5 V, 4 V and 3 V;
- the delays of each gate at 5 V, 4 V, and 3 V are 2 ns, 4 ns, and 6 ns, respectively;
- the power consumptions of each gate at 5 V, 4 V, and 3 V are 25  $\mu\text{W}$ , 16  $\mu\text{W}$ , and 9  $\mu\text{W}$ , respectively;
- the delays of level converters are lumped into the delay increase of voltage scaling.

The original circuit is shown in Fig. 2(a) and the critical paths are emphasized by bold lines. Obviously, the path delay of the critical path is 8 ns, and the slacks of  $G1$ ,  $G2$ , and  $G3$  are 4 ns, 4 ns, and 2 ns, respectively. So, the supply voltages of  $G1$  and  $G2$  are scaled to 3 V, and that of  $G3$  is scaled to 4 V in Fig. 2(b). After such voltage scaling, the critical path delay of the circuit becomes 12 ns, which is 4 ns longer than the original one. When a path selection algorithm is applied, four long paths, which are represented by bold lines in Fig. 2(b), are selected for optimization. Based on these four paths, the credit of  $G2$  is set to 4, and those of  $G1$  and  $G3$  are set to 2. Hence, the supply voltage of  $G2$  is scaled to 4 V as shown

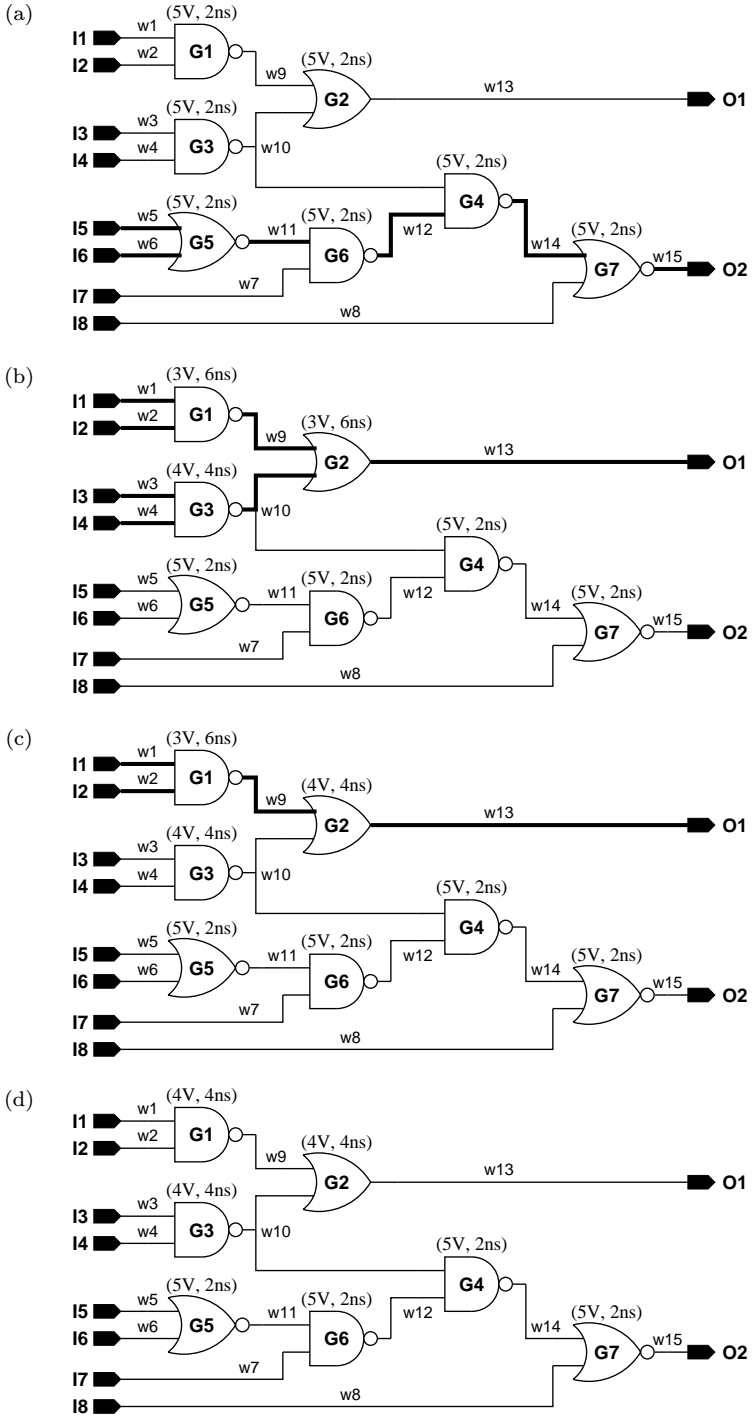


Fig. 2. An illustration example for the proposed algorithm.

in Fig. 2(c). There remain two long paths, and the credits of  $G1$  and  $G2$  are 2. Since the power increase of  $G1$  from 3 V to 4 V ( $7 \mu\text{W}$ ) is less than that of  $G2$  from 4 V to 5 V ( $9 \mu\text{W}$ ),  $G1$  is scaled to 4 V as shown in Fig. 2(d). Then, the circuit is optimized and the power reduction is  $27 \mu\text{W}$ .

#### 4. Experimental Results

We have implemented our algorithm in C on a Pentium-II 450 PC running Linux (RedHat 6.0) with 128 MB memory, and performed experiments on all the ISCAS85 benchmarks. In addition, we implemented the CVS technique and the CFMV technique for comparison.

The experimental environment is shown in Fig. 3. The control file provides the supply voltages. In our experimental cell library, the length of each MOS is  $0.8 \mu\text{m}$ , the width of each PMOS is  $16.8 \mu\text{m}$  and the width of each NMOS is  $8 \mu\text{m}$ . Using HSPICE to simulate each gate in the cell library, we obtained the parameters for timing and power analysis. The voltage scaling program then takes the netlist, the control file, and the parameter file as inputs to reduce the power consumption with the aid of the power estimator.

For the timing analysis, the rising delay  $T_{dLH}$  of a gate is estimated by:

$$T_{dLH} = (\text{rise a0}) + (\text{rise a1}) \times C_{\text{out}}, \tag{1}$$

where  $C_{\text{out}}$  is the sum of the output capacitance of the gate and the input capacitances of its fanouts. The falling delay is estimated similarly. If the supply voltage of a gate is scaled to  $V'_{dd}$ , its rising delay is estimated by:

$$T'_{dLH} = T_{dLH} \times \frac{V'_{dd}}{V_{dd}} \times \frac{(V_{dd} - V_{thp})^2}{(V'_{dd} - V_{thp})^2}, \tag{2}$$

where  $V_{thp}$  is the threshold voltage of PMOS.

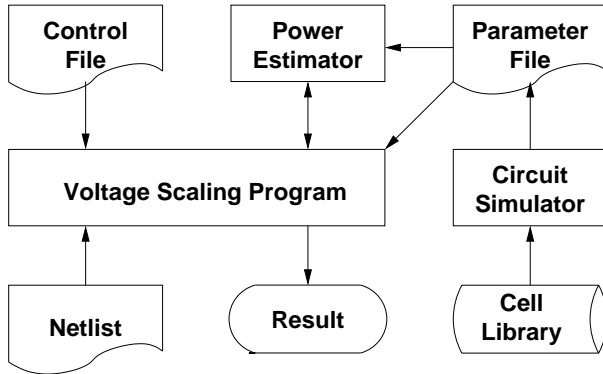


Fig. 3. Experiment environment.

For the power analysis, the activity factor of each primary input is assumed to be 0.5, and the activity factors of other gates are computed accordingly. Then, the power consumption  $P_d$  of a gate with supply voltages  $V'_{dd}$ , can be estimated by:

$$P_d = \frac{1}{2} \times f \times \alpha \times (V'_{dd})^2. \quad (3)$$

To make a fair comparison between the proposed algorithm and the previous works, we first show the characteristics of the OBMVS, the CVS and the CFMV techniques in Table 1.

Table 1. Characteristics of OBMVS, CVS, and CFMV.

Features	OBMVS	CVS	CFMV
No. supply voltages	2 or more	2	2 or more
Clustered	No	Yes	Yes
Level converters	Yes	Yes	No
Voltage difference	Unlimited	Unlimited	Limited
Path sensitized	Yes	No	No

Table 2. Results of OBMVS, CVS and CFMV with 5 V and 4 V.

Circuit name	OBMVS		CVS		CFMV	
	Pwr. Red.	Time	Pwr. Red.	Time	Pwr. Red.	Time
c432	12.65%	0.02	0.00%	0.010	4.18%	0.02
c499	10.63%	0.02	0.00%	0.010	8.97%	0.08
c880	31.31%	0.44	16.25%	0.100	14.25%	0.10
c1355	6.51%	0.02	0.00%	0.020	5.14%	0.10
c1908	26.97%	15.67	7.15%	0.300	17.36%	0.41
c2670	31.00%	14.95	9.14%	0.900	21.36%	1.59
c3540	28.53%	229.73	3.54%	0.490	16.23%	1.96
c5315	32.92%	311.26	19.78%	5.660	21.72%	5.62
c6288	18.28%	5391.84	0.62%	0.460	8.63%	1.97
c7552	30.92%	3141.26	15.21%	11.540	15.59%	10.45

When 5 V and 4 V are given as supply voltages, we can compare the results of the OBMVS technique with those of the CVS and the CFMV techniques in Table 2. Obviously, the results of the OBMVS technique are all better than those of the previous works. On average, the percentages of power reductions by the OBMVS, the CVS and the CFMV techniques are 22.97%, 7.17% and 13.34%, respectively.

When 5 V and 3 V are given as supply voltages, we cannot make a comparison between the OBMVS technique and the CFMV technique as the voltage difference of the CFMV technique is limited. Therefore, we compare the results of the OBMVS technique and the CVS technique with the loose lower bounds which are obtained in the first phase of the OBMVS algorithm. In Table 3, we can see that the results of the OBMVS technique are all much better than those of the CVS technique and

Table 3. Results of OBMVS, CVS and the lower bound with 5 V and 3 V.

Circuit name	OBMVS		CVS		Lower bound	
	Pwr. Red.	Time	Pwr. Red.	Time	Pwr. Red.	Slack
c432	18.88%	0.03	0.00%	0.010	20.46%	-47
c499	17.30%	0.22	0.00%	0.010	18.90%	-84
c880	43.05%	4.01	17.08%	0.070	54.06%	-420
c1355	10.76%	0.38	0.00%	0.030	11.58%	-76
c1908	36.72%	62.50	6.53%	0.160	48.89%	-2366
c2670	43.06%	107.55	18.58%	0.880	54.95%	-1762
c3540	39.72%	581.18	5.67%	0.460	49.97%	-6871
c5315	52.07%	1125.43	29.66%	4.820	60.33%	-5224
c6288	18.84%	8186.79	1.69%	0.440	54.17%	-48268
c7552	42.40%	12456.35	10.57%	5.860	58.48%	-13417

are close to the lower bounds in most benchmarks. On average, the percentages of power reductions by the OBMVS and the CVS techniques as well as the lower bound are 32.28%, 8.99% and 43.18%, respectively. In fact, the negative slacks at the seventh column of Table 3 could represent the tightness of the lower bounds. The larger the negative slack, the looser the lower bound.

When 5 V, 4 V and 3 V are given as supply voltages, the CVS technique is not available for comparison since only two supply voltages are allowed in the CVS technique. Consequently, the results of the OBMVS technique are compared with those of the CFMV technique and the lower bounds in Table 4. As expected, the results of the OBMVS technique are even better than those of the CFMV technique and are closer to the lower bounds. On average, the percentages of power reductions by the OBMVS, the CFMV techniques and the lower bound are 34.72%, 17.60% and 44.97%, respectively.

Table 4. Results of OBMVS, CFMV and the lower bound with 5 V, 4 V and 3 V.

Circuit name	OBMVS		CFMV		Lower bound	
	Pwr. Red.	Time	Pwr. Red.	Time	Pwr. Red.	Slack
c432	20.26%	0.16	5.07%	0.02	21.86%	-62
c499	17.49%	0.88	13.15%	0.15	18.90%	-84
c880	48.27%	14.57	21.39%	0.37	56.46%	-441
c1355	10.76%	1.73	8.58%	0.29	11.58%	-76
c1908	39.42%	268.60	20.56%	0.73	51.22%	-2452
c2670	46.69%	379.83	27.56%	5.29	57.05%	-1840
c3540	42.59%	2435.34	23.09%	6.48	54.53%	-7689
c5315	53.81%	3849.79	30.91%	21.35	61.58%	-5904
c6288	22.40%	31062.83	9.11%	4.44	55.87%	-48989
c7552	45.51%	40073.60	16.53%	19.13	60.61%	-14251

## 5. Conclusions

In this paper, we released the clustering constraint used by previous works and proposed a voltage scaling technique with multiple supply voltages to significantly reduce the power consumption of combinational circuits. Our technique first operates the gates with their lowest feasible supply voltages and then uses an existing path selection algorithm for optimization.

The main ideas of the proposed OBMVS technique are the identification of the false paths as well as the release of the clustering constraint. Consequently, a significant improvement by the OBMVS technique over previous works is achieved. From the experimental results, we can see that on average the amount of power reduction by the OBMVS technique is about 3.4 times of that by the CVS technique and is about 1.9 times of that by the CFMV technique. Furthermore, the percentages of power reductions by the OBMVS technique are close to the lower bounds.

## Acknowledgment

This work was supported by the National Science Council, Taiwan, R.O.C., under Grant NSC 90-2213-E-002-113.

## References

1. A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design", *IEEE J. Solid-State Circuits* **27** (1992) 473–484.
2. S. Rajee and M. Sarrafzadeh, "Variable voltage scheduling", in *Proc. ISLPD*, Apr. 1995, pp. 9–14.
3. J. M. Chang and M. Pedram, "Energy minimization using multiple supply voltages", *IEEE Trans. VLSI Systems* **5** (1997) 436–443.
4. A. Manzak and C. Chakrabarti, "A low power scheduling scheme with resources operating at multiple voltages", in *Proc. Int. Symposium on Circuits and Systems*, June 1999, pp. 354–357.
5. K. Usami and M. Horowitz, "Clustered voltage scaling technique for low-power design", in *Proc. ISLPD*, Apr. 1995, pp. 3–8.
6. C. W. Yeh, M. C. Chang, S. C. Chang, and W. B. Jone, "Gate-level design exploiting dual supply voltages for power-driven applications", in *Proc. 36th Design Automation Conf.*, June 1999, pp. 68–71.
7. Y. J. Yeh, S. Y. Kuo, and J. Y. Jou, "Converter-free multiple-voltage scaling techniques for low-power CMOS digital design", *IEEE Trans. Computer-Aided Design of Integrated Circuits* **20** (2001) 172–176.
8. J. Benkoski, E. V. Meersch, L. J. M. Claesen, and H. De Man, "Timing verification using statically sensitizable paths", *IEEE Trans. Computer-Aided Design of Integrated Circuits* **9** (1990) 1073–1084.
9. D. Du, H. Yen, and S. Ghanta, "On the general false path problem in timing analysis", in *Proc. 26th Design Automation Conf.*, June 1989, pp. 555–560.
10. S. Perremans, L. J. M. Claesen, and H. De Man, "Static timing analysis of dynamically sensitizable paths", in *Proc. 26th Design Automation Conf.*, June 1989, pp. 568–573.
11. P. McGeer and R. Brayton, "Efficient algorithms for computing the longest viable path in a combinational network", in *Proc. 26th Design Automation Conf.*, June 1989, pp. 561–567.

12. D. Brand and Y. Iyengar, "Timing analysis using functional analysis", Tech. Rep., IBM Thomas J. Watson Research Center, 1986.
13. H. C. Chen and D. H. C. Du, "Path sensitization in critical path problem", *IEEE Trans. Computer-Aided Design of Integrated Circuits* **12** (1993) 196–207.
14. H. C. Chen, D. H. C. Du, and L. R. Liu, "Critical path selection for performance optimization", *IEEE Trans. Computer-Aided Design of Integrated Circuits* **12** (1993) 185–195.

