

# AN OPTIMIZATION-BASED LOW-POWER VOLTAGE SCALING TECHNIQUE USING MULTIPLE SUPPLY VOLTAGES

Yi-Jong Yeh and Sy-Yen Kuo

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan  
yeh@lion.ee.ntu.edu.tw, sykuo@cc.ee.ntu.edu.tw

## ABSTRACT

In this paper, we proposed a voltage scaling technique with multiple supply voltages for low-power designs. We considered the path sensitization as well as releasing the clustering constraint applied in the CVS (Clustered Voltage Scaling) technique. Our technique operates the gates with the lowest feasible supply voltages and then uses an existing path selection algorithm for optimization. Experiments are conducted on the ISCAS85 benchmarks and the results show that about 20% power on average can be further reduced by our technique in comparison with the CVS technique.

## 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.

The conclusion of [1] provides us a simple rule in power reduction, i.e. operate a circuit as slowly as possible with the lowest possible supply voltage. 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.

In [2]–[4], the 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. In [5] and [6], the power consumption was reduced with two supply voltages at gate level, where level converters were inserted to prevent the static current when the gates with lower supply voltages drive the gates with higher supply voltages.

To reduce the complexity of physical layout with multiple supply voltages, gates of the same supply voltage are

clustered at circuit topology in [5] and [6]. However, gate clustering can be done at the early phase of physical layout. Therefore, we released the clustering constraint, applied in [5] and [6], and proposed a multiple-voltage scaling technique to freely exploit the timing slacks at gate level in this paper.

## 2. DEFINITIONS AND TERMINOLOGIES

We first give some basic terms, which can be found in [11], and use them throughout this paper.

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. Wire  $f_i$ ,  $0 \leq i \leq m - 1$ , is called an *on-input* of  $P$  which connects gate  $G_i$  to gate  $G_{i+1}$ . A wire is called a *side-input* of  $f_i$  if it is connected to  $G_{i+1}$  but is not originated from  $G_i$ .

A *primary input vector* is a vector of logic values at all the primary inputs. 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  $f$ . 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.

A path which can be activated by at least one primary input vector is defined as a *sensitizable path*. On the contrary, a path which will never be activated by any primary input vector is called a *false path*. The *critical paths* are the longest sensitizable paths in a circuit.

The *slack* of a gate  $G$ , denoted by  $s(G)$ , is defined as the maximum increase in delay that  $G$  may have under the timing constraint.

## 3. 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.

This research was supported by National Science Council, Taiwan, under the Grant NSC 87-2213-E259-007.

```

OB_MVS()
1  For (each gate  $G$  of the circuit) Do
2      Set the credit of  $G$  to 0;
3      Set the voltage of  $G$  to the lowest  $V_{ddi}$  such that  $d(G, V_{ddi}) - d(G, V_{dd0}) \leq s(G)$ ;
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 Increase the credit of  $G$ ;
8  Insert the gates with positive credits to a priority queue,  $PQ$ ;
9  Do
10     Retrieve a gate  $G$  from the top of  $PQ$ ;
11     Increase the voltage of  $G$ ;
12     If (the voltage of  $G \neq V_{dd0}$ ) Then Insert  $G$  back to  $PQ$ ;
13     For (each path  $P$  in  $FS$ ) Do
14         If ( $d(P) \leq$  timing constraint) Then
15             Decrease the credit of each gate in  $P$ ; Delete  $P$  from  $FS$ ;
16 While ( $FS \neq \phi$ )

```

Figure 1: The optimization-based algorithm for multiple-voltage scaling.

Several path sensitization criteria have been proposed to estimate the delay of a circuit including the *exact criterion*, the *loose criterion*, the *BMCD criterion* [7], the *DYG criterion* [8], the *PCD criterion* [9], the *viable criterion* [13], the *BI criterion* [10], and the *dynamic criterion* [11]. 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. Generally speaking, 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 [11]–[13], most long paths in a complex circuit are actually false. Furthermore, a significant portion of long false paths do not need to be shortened [14]. 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 in [14] and two selection algorithms, vector-oriented and path-oriented, were proposed. For a circuit with many primary inputs, the vector-oriented algorithm may not be feasible since there are too many input combinations. Consequently, the path-oriented selection algorithm proposed in [14] was adopted in our optimization algorithm.

#### 4. THE PROPOSED ALGORITHM

Now, we can formulate the problem we’d like to solve in this paper 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.*

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 achieves the lower bound of the formulated problem 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 order of the gates. Based on the critical order, 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.

The proposed algorithm, `OB_MVS()`, is shown in Figure 1. The given supply voltages are arranged in descending order and are labeled  $V_{dd0}, V_{dd1}, \dots, V_{ddn}$  if the number of the given supply voltages is  $(n + 1)$ . 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 `POSA_FeasibleSet()`, which can be found in [14], to obtain a set of long paths,  $FS$ , for optimization. Lines 5–7 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 8. The priority queue arranges a data structure such that the

gate with the maximum credit can be easily retrieved. Lines 9–16 optimize the circuit by increasing the supply voltages of the most critical gates until the timing constraint is met.

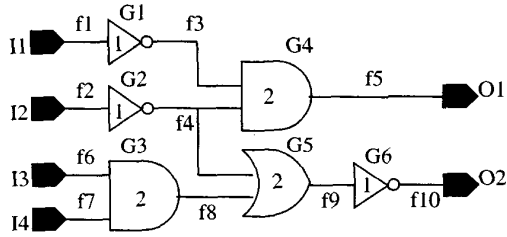


Figure 2: An example for the illustration of the proposed algorithm.

Take Figure 2 as an example. Assume that 3 supply voltages are given, the delays of an AND gate or an OR gate at these 3 voltages are 2, 4 and 6 time units respectively, and those of a NOT gate are 1, 2 and 3 time units. In the beginning, the slacks of  $G1$  and  $G4$  are 2 time units, and that of  $G2$  is 1 time unit. So, the supply voltage of  $G1$  is set to  $V_{dd2}$ , and those of  $G2$  and  $G4$  are set to  $V_{dd1}$ . After such voltage assignment, the delay of the circuit becomes 7 time units while the original delay is 5 time units. Next, `POSA_FeasibleSet()` is applied and 2 paths,  $(I1, f1, G1, f3, G4, f5, O1)$  and  $(I2, f2, G2, f4, G4, f5, O1)$ , are selected for optimization. Based on these 2 paths, the credit of  $G4$  is set to 2 and those of  $G1$  and  $G2$  are set to 1. Therefore, the supply voltage of  $G4$  is set back to  $V_{dd0}$ . Then, the delays of these 2 selected paths are no more than 5 time units and the circuit is optimized.

## 5. EXPERIMENTAL RESULTS

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

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 rising delay  $T_{dLH}$  of gate  $G$  is estimated by

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

where  $C_{out}$  is the sum of the output capacitance of gate  $G$  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

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

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

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

When 2 supply voltages are given, we can compare the results of the OB\_MVS technique with those of the CVS technique. The OB\_MVS technique can identify the false paths as well as releasing the clustering constraint which was applied in the CVS technique. Consequently, a significant improvement in the OB\_MVS technique over the CVS technique is expected. In Table 1, two supply voltages, 5V and 4V, are given for voltage scaling. We can see that the results of OB\_MVS are all much better than those of CVS. On average, the power reduction of OB\_MVS is 22.97%, while that of CVS is 7.17%. In Table 2, where 5V and 3V are applied for voltage scaling, the results of OB\_MVS are even better than those of CVS. The average power reduction of OB\_MVS is 32.28%, while that of CVS is 8.99%.

The results of the OB\_MVS with 3 supply voltages, 5V, 4V and 3V, are shown in Table 3 and are compared to the lower bounds which are obtained from lines 1–3 of OB\_MVS. The third column of Table 3 shows the total negative slacks at the lower bounds, which represent the tightness of the lower bound. We can find that the power reductions of OB\_MVS are close to those of the lower bounds with small total negative slacks.

## 6. CONCLUSIONS

In this paper, we released the clustering constraint applied in the CVS technique and proposed a voltage scaling technique with multiple supply voltages. Our technique operates the gates with the lowest feasible supply voltages and then uses an existing path selection algorithm for optimization.

From the experimental results, we can see that our algorithm adds another 19.55% power reduction on average over that of the CVS technique. Furthermore, the power reductions of our algorithm are close to the lower bounds with small total negative slacks.

## 7. REFERENCES

- [1] Chandrakasan A.P., Sheng S. and Brodersen R.W. "Low-Power CMOS Digital Design". *IEEE Journal of Solid-State Circuits*, pages 473–484, Apr. 1992.

Table 1: Comparisons between OB\_MVS and CVS with 2 supply voltages, 5V and 4V.

circuit name	OB_MVS		CVS	
	Pwr. Red.	Time	Pwr. Red.	Time
c432	12.65%	0.02	0%	0.010
c499	10.63%	0.02	0%	0.010
c880	31.31%	0.44	16.25%	0.100
c1355	6.51%	0.02	0%	0.020
c1908	26.97%	15.67	7.15%	0.300
c2670	31.00%	14.95	9.14%	0.900
c3540	28.53%	229.73	3.54%	0.490
c5315	32.92%	311.26	19.78%	5.660
c6288	18.28%	5391.84	0.62%	0.460
c7552	30.92%	3141.26	15.21%	11.54

Table 2: Comparisons between OB\_MVS and CVS with 2 supply voltages, 5V and 3V.

circuit name	OB_MVS		CVS	
	Pwr. Red.	Time	Pwr. Red.	Time
c432	18.88%	0.03	0.11%	0.010
c499	17.30%	0.22	0%	0.010
c880	43.05%	4.01	17.08%	0.070
c1355	10.76%	0.38	0%	0.030
c1908	36.72%	62.50	6.53%	0.160
c2670	43.06%	107.55	18.58%	0.880
c3540	39.72%	581.18	5.67%	0.460
c5315	52.07%	1125.43	29.66%	4.820
c6288	18.84%	8186.79	1.69%	0.440
c7552	42.40%	12456.35	10.57%	5.860

Table 3: Comparisons between OB\_MVS and the lower bound with 3 supply voltages, 5V, 4V and 3V.

circuit name	LB		OB_MVS	
	Pwr. Red.	Slack	Pwr. Red.	Time
c432	21.86%	-62	20.26%	0.16
c499	18.90%	-84	17.49%	0.88
c880	56.46%	-441	48.27%	14.57
c1355	11.58%	-76	10.76%	1.73
c1908	51.22%	-2452	39.42%	268.60
c2670	57.05%	-1840	46.69%	379.83
c3540	54.53%	-7689	42.59%	2435.34
c5315	61.58%	-5904	53.81%	3849.79
c6288	55.87%	-48989	22.40%	31062.83
c7552	60.61%	-14251	45.51%	40073.60

- [2] Raje S. and Sarrafzadeh M. "Variable Voltage Scheduling". *Proceedings ISLPD*, Apr. 1995, pages 9–14.
- [3] Chang J.M. and Pedram M. "Energy Minimization Using Multiple Supply Voltages". *IEEE Transactions on VLSI Systems*, pages 436–443, Dec. 1997.
- [4] Manzak A. and Chakrabarti C. "A Low Power Scheduling Scheme with Resources Operating at Multiple Voltages". *IEEE International Symposium on Circuits and Systems*, Jun. 1999, pages 354–357.
- [5] Usami K. and Horowitz M. "Clustered Voltage Scaling Technique for Low-Power Design". *Proceedings ISLPD*, Apr. 1995, pages 3–8.
- [6] Yeh C.W., Chang M.C., Chang S.C. and Jone W.B. "Gate-Level Design Exploiting Dual Supply Voltages for Power-Driven Applications". *Proceedings Design Automation Conference*, Jun. 1999, pages 68–71.
- [7] Benkoski J., Vanden Meersch, E., Claesen L.J.M. and De Man, H. "Timing Verification Using Statically Sensitizable Paths". *IEEE Transactions on Computer-Aided Design*, pages 1073–1084, Oct. 1990.
- [8] Du D., Yen H. and Ghanta S. "On the General False Path Problem in Timing Analysis". *Proceedings Design Automation Conference*, Jun. 1989, pages 555–560.
- [9] Perremans S., Claesen L.J.M. and De Man H. "Static Timing Analysis of Dynamically Sensitizable Paths". *Proceedings Design Automation Conference*, Jun. 1989, pages 568–573.
- [10] Brand D. and Iyengar Y. "Timing analysis using functional analysis". IBM Thomas J. Watson Research Center, Technical Report, 1986.
- [11] Chen H.C. and Du D.H.C. "Path Sensitization in Critical Path Problem". *IEEE Transactions on Computer-Aided Design*, pages 196–207, Feb. 1993.
- [12] Devadas S., Keutzer K. and Malik S. "Delay computation in combinational logic circuits: Theory and algorithms". *International Conference on Computer-Aided Design*, 1991, pages 176–179.
- [13] McGeer P. and Brayton R. "Efficient algorithms for computing the longest viable path in a combinational network". *Proceedings Design Automation Conference*, Jun. 1989, pages 561–567.
- [14] Chen H.C., Du D.H.C. and Liu L.R. "Critical Path Selection for Performance Optimization". *IEEE Transactions on Computer-Aided Design*, pages 185–195, Feb. 1993.