Clock Scheduling and Cell Library Information Utilization for Power Supply Noise Reduction

Yooseong Kim, Sangwoo Han, and Juho Kim

Abstract—**Power supply noise is fundamentally caused by large current peaks. Since large current peaks are induced by simultaneous switching of many circuit elements, power supply noise can be minimized by deliberate clock scheduling which utilizes nonzero clock skew. In this paper, nonzero skew clock scheduling is used to avoid the large peak current and consequently reduce power supply noise. While previous approaches require extra characterization efforts to acquire current waveform of a circuit, we approximate it only with existing cell library information to be easily adapted to conventional design flow. A simulated annealing based algorithm is performed, and the peak current values are estimated for feasible clock schedules found by the algorithm. The clock schedule with the minimum peak current is selected for a solution. Experimental results on ISCAS89 benchmark circuits show that the proposed method can effectively reduce the peak current.**

Index Terms—**Power supply noise, clock scheduling, peak current, current waveform estimation, cell library**

I. INTRODUCTION

As circuits become densely packed and supply voltage scales, power integrity issue has become one of the major problems in chip design. Large current peaks caused by simultaneous switching of circuit elements often lead to IR-drop and L*di/dt drop, known as power supply noise. Power supply noise makes us difficult to ensure that the designated supply voltage is delivered to every circuit element all the time. The fluctuation in supply voltage may cause logic failure or timing violation. What makes the problem more serious is that the noise margin decreases as supply voltage gets lower and lower. This makes power integrity problem to be of particular importance in today's deep submicron environment.

There are several research areas to tackle the power supply noise problem. Optimum wire sizing of P/G lines [8], power supply planning in floorplanning stage [6,9] and decoupling capacitance allocation [10] are the most prevalent techniques to resolve power supply noise. Statistical timing analysis is also starting to consider power supply noise as an important variation component [7].

On the other hand, there is a way to solve the fundamental reason of the problem. Power supply noise is basically caused by a sudden large currents demand, which is due to simultaneous switching of circuit elements. Simultaneous switching occurs severely as a result of zero skew clock arrival time, since clock triggers all sequential elements and their connected combinational logic at the same time. Therefore, it is an obvious way for reducing power supply noise to control clock arrival times so that simultaneous switching events are as much avoided as possible. As shown in Fig. 1, a well-chosen non-zero clock skew can effectively reduce peak current of circuit by avoiding simultaneous switching.

Clock scheduling for minimizing peak current draw is studied by many researchers [1-3]. A Genetic Algorithm (GA) based clock scheduling is proposed in [1], but the authors considered only the first level of combinational logic so that the current draw of combination logic is not fully considered. A heuristic to enhance the quality of GA's solution is presented in [2]. The effect of clock scheduling on current waveform of combinational logic gates is also

Manuscript received Mar. 3, 2009; revised Mar. 12, 2009.
Department of Computer Science and Engineering, Sogang University, Seoul, 121-742, Republic of Korea
E-mail : yooskim@sogang.ac.kr

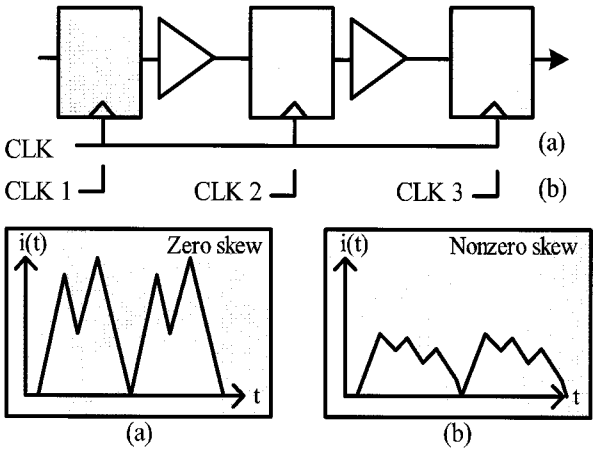


Fig. 1. Peak current reduction by clock scheduling.

considered in [2-4]. The iterative algorithm to find the optimal clock schedule for power supply noise suppression is presented in [3]. Besides the clock scheduling, retiming is also used to minimize the maximum simultaneous switching current in [4].

All the above researches require extra characterization to acquire current waveform of a circuit. Current profiles are needed to estimate the instantaneous current drawn by each element. Numbers of SPICE simulations has to be done for current profiling of all gates present in the netlist. However, the proposed method approximates the current profiles of each gate using only the existing library information stored in liberty format. Since the proposed work does not require another time consuming characterization step, it can be easily fitted into conventional design flow.

The proposed algorithm takes as input the gate-level netlist of sequential circuit, given clock period and library information in liberty (.LIB) format. A simulated annealing based algorithm is employed to search many feasible clock schedules. For a searched clock schedule, the peak current is estimated and used for cost function value of each immediate solution. To acquire the peak current, the current waveform of the circuit is approximated first by summing up all gates' current profiles. Finally, the solution with the highest cost function value is selected. Clearly, the selected solution is the clock schedule that minimizes the peak current value and reduces power supply noise. It has been proved by experimental results that the proposed algorithm is as effective as the previous work while it does not require significant additional efforts for simulation.

In this paper, only edge triggered flip-flops are considered, since they represent the worst case condition for current peaks [1]. It is because all switching activities are synchronized with clock arrival times. Given timing constraints, the problem to solve is to find the feasible clock schedule with which power supply noise is minimized.

The rest of this paper is organized as follows. For background, problem formulation and basic concepts are described in Section 2. Then the proposed current waveform estimation method using cell library information is presented in Section 3. In Section 4, the algorithm to minimize power supply noise is described. Experimental results are provided in Section 5, and we finally conclude in Section 6.

II. BACKGROUND

1. Problem Formulation

If flip-flop i drives a combinational logic block whose output in turn drives flip-flop j, the timing constraints between them is given as follows:

$$x_i + d(i) + d(i,j) > x_j + Hold(j) \quad (1)$$

$$x_i + d(i) + D(i,j) + Setup(i) < T + x_j \quad (2)$$

,where x_i and x_j are clock arrival times of flip-flop i and j, $d(i)$ is internal delay of flip-flop i, and $Hold(j)$ and $Setup(i)$ is hold time constraint and setup time constraint of corresponding flip-flop. T is the required clock period, and $d(i,j)$ and $D(i,j)$ represent the minimum and maximum delay of combinational logic between flip-flop i and j.

The problem in this paper can be stated as follows: "Given the skew constraints of a system, determine a non-zero clock skew schedule such that the power supply noise is minimized without violating any skew constraints [2-3]."

2. Constraint Graph

To check the feasibility of a given clock schedule, the constraint graph based approach [5] is used. For a pair of flip-flops, a skew constraint is defined as in (1) and (2). If we rewrite the equations, we can use $l_{i,j} < skew_{i,j} (= x_i - x_j) < u_{i,j}$ to represent the skew constraints between two flip-flops.

Then, all skew constraints between every pair of flip-flops can be represented by a constraint graph as shown in Fig. 2 [5]. In a constraint graph G(V,E), each vertex corresponds to a flip-flop. If there is a path of combinational logic between two flip-flops i and j, then a pair of directed edges, e_{ij} and e_{ij}, is added between corresponding vertices.

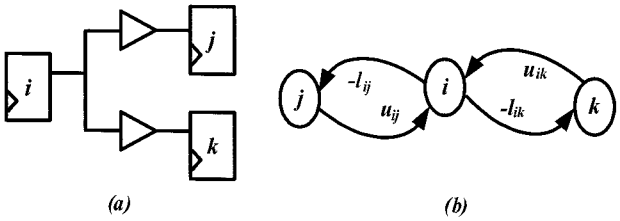


Fig. 2. (a) An example sequential circuit (b) Corresponding constraint graph.

The weight of edge e_{ij} is set to $-l_{i,j}$, and the weight of edge e_{ji} is set to $u_{i,j}$. We denote the weight of edge e_{ij} as w_{ij}. If there is a negative cycle in the graph, then the skew constraints are infeasible so that there is no clock schedule that satisfies all skew constraints. Using the Bellman-Ford algorithm, the presence of negative cycle in G is verified in polynomial time [5].

As long as there is no negative cycle in the graph, the feasibility of a clock schedule can be verified as follows. Let us label each vertex with corresponding flip-flop's clock arrival time. Thus, each vertex has a node value of corresponding flip-flop's clock arrival time. Then the clock schedule is feasible if and only if the following satisfies.

$$x_i < x_j + w_{ji} \text{ for all pairs of vertices } (i, j) \quad (3)$$

3. Current Profiles and Current Waveform

The current waveform of a circuit needs to be constructed to acquire the peak current value of a circuit in a given clock schedule. It is time consuming and sometimes infeasible to simulate the whole circuit to get current waveform. Therefore, current profiles of gates are usually used to acquire the circuit current waveform. Current profiles of all types of gates are prepared by simulation first. Then, current waveform is constructed by combining current profiles as shown in Fig. 3. In previous approaches [1-4], there is a drawback that numerous SPICE simulations are required to obtain current profiles of gates. However, current profiles are obtained directly from existing cell library information in this paper.

III. Current Waveform Estimation Using Cell Library Information

1. Current Profiles Estimation

Cell library information in conventional design flow

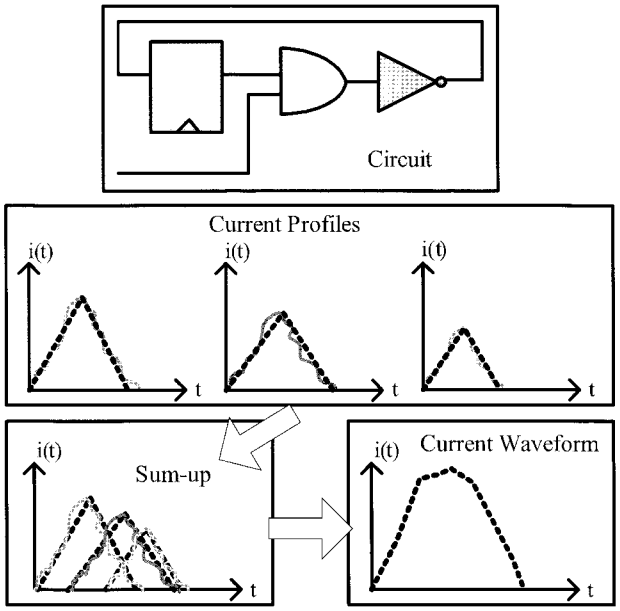


Fig. 3. Current profiles and current waveform.

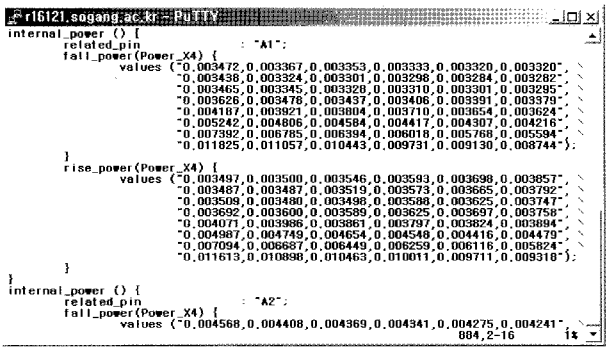


Fig. 4. Internal power table in cell library.

contains timing and power information of all cells in Liberty format [12,13]. The lookup tables for delay, output transition time and internal power are indexed by the input transition time and the output load capacitances. The tables are associated to every input pin transition that causes output transition. Internal power includes short-circuit power and internal switching power dissipated by diffusion capacitance. Thus, internal power tables represent the power consumption according to the any output transition. Fig. 4 shows the internal power table in typical cell library.

It is known that triangular approximation is accurate enough to represent the real current waveform [1-4]. Current profiles consists four parameters as shown in Fig. 5. *Ttrig* and *Tend* represent the time when supply current starts to rise and the time when it ends, respectively. *Tpeak* is the time when supply current value is at the maximum, and *Ipeak* is the maximum current value.

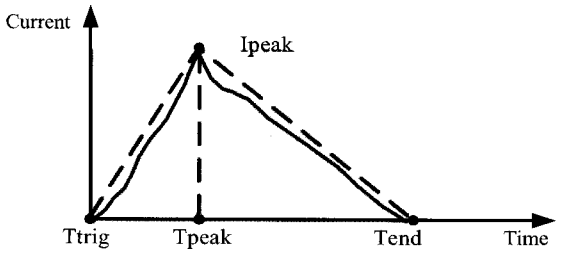


Fig. 5. Four parameters to represent triangular waveform.

Table 1. Formulas to derive the parameters.

Ttrig		*0*
Tpeak		T_{input_slew}
Ipeak	**hl transition**	$2 * Q_{cell} / Tend$
	lh transition	$2 * (Q_{cell} + Q_{load}) / Tend$
Tend	***Simple cells***	$T_{input\_slew} + T_{output\_slew}$
	Combination cells	$T_{input\_slew} + T_{output\_slew} + (T_{delay} / 2)$

Since current is the amount of total charge movement in a given time, we need to estimate the amount of charge movement using the information in library to estimate the current waveform. Internal power values in the lookup tables are in unit of energy per transition. Therefore, we calculate the amount of charge by the simple physics rule as shown in Equation (4).

$$Q_{cell} = E_{cell} / Vdd \tag{4}$$

Table 1 shows the formulas to derive the parameters using the given information in cell library. We categorized cells into two sets: simple cells and combination cells. Simple cells are gates in single stage structure which are, for example, INV, NAND, or NOR gates. Combination cells are in multiple stage structure in which two or more simple cells are combined. AND, OR gates, or flip-flops and latches are combination cells.

Since *Ttrig* is the time when input signal starts to change, it is simply set to zero. By numerous simulation, it is found that *Tpeak* is almost the same as the time when input signal finishes to change. Therefore, we approximate *Tpeak* as input transition time, T_{slew}. *Ipeak* is calculated by the above formula because the area of the triangle is the same as the amount of total charge movement. When output transition is low to high, the charge in output loading should also be included using the $Q = CV$ rule. Thus, Q_{load} can be calculated by using the equation below.

$$Q_{load} = C_{load} * Vdd \tag{5}$$

,where C_{load} is the output load capacitance. As shown in Table 1, *Tend* is assumed to be input transition time plus output transition time, for simple cells. However, for combination cells, a half of the cell delay is added to approximate the transition time at internal stages.

Note that the above parameters are relative to T_{trig}. To estimate current waveform of a circuit, we need to combine the current profiles of gates. It is assumed that the output pin starts to transition after delay of the gate. Thus, T_{trig} of the fanout gate is calculated to be T_{trig} of the fanin gate plus T_{delay} of the fanin gate. When calculating the T_{trig} of the gate whose input pin is connected to primary input, T_{trig} is simply set to zero.

2. Current Waveform Estimation

All the information needed to derive the parameters mentioned in the previous section can be found in lookup tables in library. Then we can make triangular current profiles of all gates for all input vectors with corresponding output loading and input slew. The output loading and input slew can also be calculated from cell library information by input pin capacitance of fanout gate and output transition time of fanin gate.

For more efficiency, the proposed method clusters each flip-flop and its corresponding combinational logic. As shown in Fig. 6, each cluster consists of a flip-flop and combinational gates which are subject to the flip-flop.

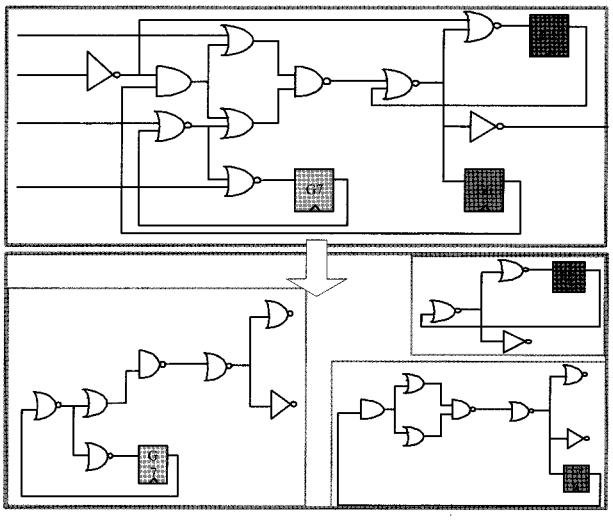


Fig. 6. Clustering of ISCAS89 s27.

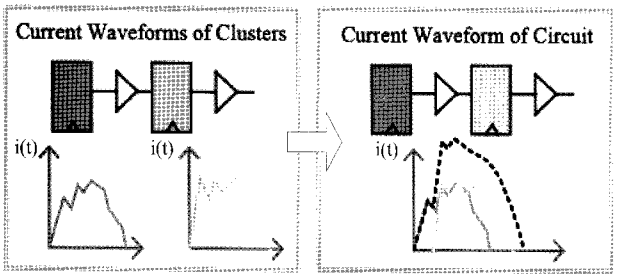


Fig. 7. Combining cluster waveforms to construct circuit waveform.

The circuit current waveform can be constructed summing up all current profiles of gates considering the given input vector and corresponding delay, output loading and input transition time. Once the current waveform is constructed for each cluster, it does not need to be calculated again because the circuit structure is not modified in this algorithm. Since only clock arrival time is modified, we can horizontally shift the waveform of the corresponding flip-flop. The total current waveform can be constructed by summing up current waveforms of clusters considering its clock arrival time as shown in Fig. 7.

IV. Clock Scheduling Algorithm for Power Supply Noise Reduction

The proposed algorithm requires a technology mapped gate-level netlist, a given clock period, and library information in liberty format. The constraint graph explained in Section 2 is constructed to check the feasibility of constraints and clock schedules. We first start with the zero-skew schedule as the initial solution. Since the clock period is already determined, to find zero-skew schedule is trivial; thus, all flip-flops have the same clock arrival time.

Some small period of time, e.g. a few pico-seconds, is set to a unit of skew change. Then, the algorithm perturbs each flip-flop's clock arrival time by the unit. Thus, the clock arrival time is increased, decreased by the unit skew value or unchanged. It makes the algorithm not generate too small skew that does not make any noticeable difference in the current waveform.

The above perturbation process establishes the concept of neighboring solutions in the simulated annealing algorithm. If a clock schedule can be generated by one perturbation of another clock schedule, two clock schedules are neighboring solutions. For example, let us set a unit of skew change to 2. There are two flip-flops, and the current solution is zero-skew clock schedule (0, 0). Then these are all neighboring solutions so that they can be generated from each other after one execution of the perturbation function: (0, -2), (0, 0), (0, 2), (2, -2), (2, 0), (2, 2).

To make the algorithm more efficient, we can set a limit of a clock skew $skew_{i,j}$ as the below when the algorithm perturbs a clock schedule.

$$| x_i - x_j | < T / 2 \text{ for all pairs of flip-flops } (i, j) \quad (6)$$

,where T is the clock period. This ensures that the algorithm does not generate too large skew that is hard to be realized in real circuit. This reduces the search space of the algorithm so as to increase the efficiency.

For each generated solution, the current waveform is estimated as explained in the previous section. Since only the clock arrival time is changed but the circuit structure is not modified, the current waveform of a flip-flop and its corresponding combinational logic does not change. Thus, once we construct the waveform of clusters, we do not need to do it again. Per iteration, we only horizontally shift a waveform of each cluster of a flip-flop and the combinational logic connected to the output of the flip-flop. Then the total circuit's waveform can be constructed summing up all the individual cluster waveforms.

The cost function f is simply defined as follows:

$$f(clock\_schedule) = 1 / Ipeak\_max \quad (7)$$

where $Ipeak\_max$ represents the peak current value over the whole clock period. A clock schedule with lower peak current has a higher cost function value than those with higher peak current.

The proposed algorithm is based on simulated annealing [14]. A clock schedule with higher cost function value than the previous one is always accepted for the next solution. Since the algorithm has a nonzero probability for accepting inferior solution, the clock schedule that generates higher peak current is sometimes accepted to avoid being stuck in local minima. The probability is $min\{1, \exp(-\Delta E / T)\}$, where ΔE is the absolute difference of cost function values between the current solution and new solution, and T is the current temperature. The temperature is gradually decreased so that the probability for accepting inferior solution tends to be high at first but decreases as the algorithm proceeds.

Since the temperature decreases one by one per iteration, the temperature is the number of iteration. The

pseudocode of the proposed algorithm is shown in Fig. 8. Note that perturb() in line 8 is the perturbation function explained above. The function rand(0,1) in line 14 returns a random floating number uniformly distributed between 0 to 1. The final solution *sb* shown in line 17 is the clock schedule with the best cost function value.

```
T:  Temperature (pre-determined number of iteration)

1.  Construct a constraint graph;
2.  if ( there is a negative cycle )
3.          return error;
4.  Construct the current waveforms of clusters;
5.  end = the highest peak current value among
    clusters;
6.  sb = s = initial clock schedule with zero-skew;
7.  while ( T > 0   &&   f(s) > end )
8.          sn = perturb(s) ;
9.          if   ( f(s) < f(sn) )
10.                 s = sn;
11.                 if    ( f(s) < f(sb) )
12.                         sb = sn;
13.         else
14.                 if   ( exp(-(f(s) - f(sn)) / T)   >
        rand(0,1) )
15.                         s = sn;
16.         T--;
17. return sb;
```

Fig. 8. Pseudocode of the proposed algorithm.

V. EXPERIMENTAL RESULTS

The performance of the proposed power noise reduction algorithm is evaluated using ISCAS89 benchmark circuits. The proposed algorithm has been implemented in C++ and tested on Linux machine with P4 2.4 Ghz processor. Nangate 45 nm open cell library [11] was used. The .lib cell library file was read and parsed to obtain delay, output transition time, internal power, and output load capacitance of all gates in benchmark circuits. Clock skews were perturbed in a unit of 30 ps to avoid too small skews that might rarely affect the current waveform. The temperature *T*, which is the number of iteration of the proposed algorithm, was set to 5000. The runtime for s5378, the largest circuit in this experiment, took less than 5 minutes for any input vector.

Given gate-level netlists of benchmark circuits and RC-extracted SPICE netlists for all types of gates, transistor level netlists were generated. For the maximum 100 randomly chosen input vectors, the proposed algorithm was applied. Using the same 100 input vectors and the corresponding optimal clock schedule, HSPICE simulation was performed to measure the peak current.

In Table 2, the proposed algorithm's performance on peak current reduction is shown. The peak current is reduced by 41.82% on average using the optimal clock schedule. Clock periods set to circuits are shown on the table. It is shown that the peak current can be reduced effectively by using the proposed algorithm.

Table 3 shows the accuracy comparison data with the previous simulation-based method [3]. Current profiles were obtained by HSPICE simulation for the simulation-based method. Genetic algorithm based clock scheduling method with the gene therapy strategy A was applied as in [3]. As shown in the table, the proposed method is comparably effective to the previous method.

While the accuracy is rarely impaired in the proposed method, the efficiency is greatly improved comparing to the previous method. In the previous method, to obtain current profiles for a cell, numerous SPICE simulations

Table 2. Experimental Results of Benchmark Circuits.

Circuit	# of FFs	Clock Period (ns)	Peak current (mA)		
			Before opt.	After opt.	% Reduction
s349	15	1.1	3.117	1.917	**38.5**
s382	21	1.1	4.327	2.649	**38.78**
s953	29	1.1	5.978	3.107	**48.03**
s838	32	1.9	6.594	3.791	**42.51**
s5378	179	1.4	36.90	21.67	**41.27**
Avg.					**41.82**

Table 3. Experimental Results of Benchmark Circuits.

Circuit	Peak current (mA)					
	Before optimization	Proposed	%	Previous	% Reduction	Difference
s349	3.117	1.917	**38.5**	1.958	**37.18**	**+1.32**
s382	4.327	2.649	**38.78**	2.601	**39.89**	**-1.11**
s953	5.978	3.107	**48.03**	3.286	**45.03**	**+3.0**
s838	6.594	3.791	**42.51**	3.618	**45.13**	**-2.62**
s5378	36.90	21.67	**41.27**	22.03	**43.04**	**-1.77**
Avg.			**41.82**		**42.05**	**-0.23**

were required for all possible input vectors, input slew values, and output load capacitance values combinations. As a result, it required over 30,000 simulations for 127 cells in library.

VI. Conclusions and Future Work

We have presented an approach to reduce power supply noise by clock scheduling. Efficient current waveform estimation method has been introduced to avoid the extra characterization effort. Without any need to pre-simulate all cells in library to acquire current profiles, the proposed algorithm finds the optimum non-zero skew clock schedule which minimizes the maximum simultaneous switching current. It has been shown that the proposed method can effectively reduce the peak current and therefore power supply noise.

As product life cycle becomes shorter and shorter, there are many benefits in reducing design and sign-off time. The proposed method may be used to hasten design time while maintaining design quality which is low power supply noise in this subject.

For future work, it may be possible to adapt the proposed current waveform estimation method to technology mapping. Using existing library information, the cells with lower *Ipeak* value can be selected as far as it satisfies other design constraints. However, the use of cells with lower *Ipeak* does not guarantee the lower total circuit's peak current. Therefore, an integrated solution will be needed. Also, due to the randomized nature of simulated annealing, it is hard to apply the algorithm to large circuits. A more efficient heuristic is needed. Also, to increase the accuracy of the algorithm, piecewise linear approximation of current profiles might be needed rather than the triangular approximation.

References

[1] P. Vuillod, L. Benini, A.Bogliolo, and G. De Micheli, "Clock-skew optimization for peak current reduction," *in Proceedings of ISLPED*, pp. 265-270, 1996.

[2] W.-C. D. Lam, C.-K. Koh, and C.-W. A. Tsao, "Power supply noise suppression via clock skew scheduling", in *Proceedings of ISQED*, pp. 355-360, 2002.

[3] W.-C. D. Lam, C.-K. Koh, and C.-W. A. Tsao, "Clock scheduling for power supply noise suppression using genetic algorithm with selective gene therapy," *in Proceedings of ISQED*, pp. 327-332, 2003.

[4] A. Mukherjee and R. Sankaranarayan, "Retiming and clock scheduling to minimize simultaneous switching," *in Proceedings of IEEE SOCC*, pp. 259-262, 2004.

[5] N. Shenoy, R. K. Brayton, and A. L. Sangiovanni-Vincentelli, "Graph algorithms for clock schedule optimization," *in Proceedings of ICCAD*, pp.132-136, 1992.

[6] X. Wang, J. Shi, Y. Cai. and X. Hong, "Heuristic power/ground network and floorplan codesign method," *in Proceedings of ASPDAC*, pp. 617-622, 2008.

[7] T. Enami, S. Ninomiya, and M. Hashimoto, "Statistical timing analysis considering spatially and temporally correlated dynamic power supply noise," *in Proceedings of ISPD*, pp.160-167, 2008.

[8] D.E. Khalil, and Y. Ismail, "Optimum sizing of power grids for IR drop," *in Proceedings of ISCAS*, pp. 481-484, 2006.

[9] H.-M. Chen, L.-D. Huang, and M. D. F. Wong, "Simultaneous power supply planning and noise avoidance in floorplan design," *IEEE Transactions on CAD*, vol. 24, no. 4, pp. 578-587, Apr. 2005.

[10] S. Zhao, K. Roy, and C.-K. Koh, "Decoupling capacitance allocation and its application to power-supply noise-aware floorplanning," *IEEE Transactions on CAD*, vol. 21, no. 1, pp. 81-92, Jan. 2002.

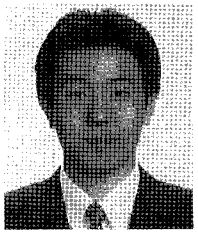
[11] Nangate 45nm Open Cell Library: http://www.si2.org/openeda.si2.org/projects/nangatelib/

[12] "Liberty Reference Manual (Version 2006.06)," Synopsys, 2006.

[13] "Library Compiler Reference Manual, Volume 1," *Synopsys*, 1999.

[14] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, May 1983.

Yooseong Kim received the BE degree in Computer Science and MS degree in Computer Science and Engineering from Sogang University in 2006 and 2009, respectively. He earned the Best Paper award at ISOCC 2008. He is currently a part-time research assistant in the CAD & Security Lab at Sogang University. His research interests are mainly on design and optimization of embedded systems and VLSI in gate or higher level.



Sangwoo Han received the BS degree in Physics and Computer Science from Sogang University in 2005. He also received the MS degree in Computer Science and Engineering from Sogang University in 2008. He is currently pursuing a Ph.D. degree in computer engineering at Sogang University. His research interests are SSTA, robust design for process variation, and design for reliability enhancement.

Juho Kim received B.S degree and Ph.D degree in Computer and Information Science from University of Minnesota in 1987 and 1995, respectively. After getting Ph.D degree, he worked as a senior member of technical staff at Cadence Design System until 1997. Professor Kim joined the department of computer science and engineering in Sogang University, Seoul, Korea in 1997, and he was a department chair from 2005 to 2008.