

# Effective Decap Insertion in Area-Array SoC Floorplan Design

CHAO-HUNG LU

National Central University, Taoyuan, Taiwan

HUNG-MING CHEN

National Chiao Tung University, Hsinchu, Taiwan

and

CHIEN-NAN JIMMY LIU

National Central University, Taoyuan, Taiwan

---

As VLSI technology enters the nanometer era, supply voltages continue to drop due to the reduction of power dissipation, but it makes power integrity problems even worse. Employing decoupling capacitances (decaps) in floorplan stage is a common approach to alleviating supply noise problems. Previous researches overestimate the decap budget and do not fully utilize the empty space of the floorplan. A floorplan usually has a lot of available space that can be used to insert the decap without increasing the floorplan area. Therefore, the goal of this work is to develop a better model to calculate the required decap to solve the power supply noise problem in area-array based designs, and increase the usage of available space in the floorplan to reduce the area overhead caused by decap insertion. The experimental results of this work are encouraging. Compared with previous approaches, our methodology reduces 38% of the decap budget in average for MCNC benchmarks but can still meet the power supply noise requirements. The final floorplan areas with decap are also smaller than the numbers reported in previous works.

Categories and Subject Descriptors: J.6 [Computer-Aided Engineering]—Computer-aided design (CAD)

General Terms: Algorithm, Reliability

Additional Key Words and Phrases: Power supply noise, floorplan, decap insertion

## ACM Reference Format:

Lu, C.-H., Chen, H.-M., and Liu, C.-N. J. 2008. Effective decap insertion in area-array SoC floorplan design. *ACM Trans. Des. Automat. Elect. Syst.* 13, 4, Article 66 (September 2008), 20 pages, DOI = 10.1145/1391962.1391974 <http://doi.acm.org/10.1145/1391962.1391974>

---

This work was supported in part by the National Science Council under contract NSC 96-2220-E-009-013 and NSC 97-2220-E-009-001.

Authors' addresses: C.-H. Lu and C.-N. J. Liu, Department of Electrical Engineering, National Central University, Taoyuan, 320 Taiwan; email: {chiu, jimmy}@ee.ncu.edu.tw; H.-M. Chen, Department of Electronics Engineering and SoC Research Center, National Chiao Tung University, Hsinchu, 300 Taiwan; email: hmchen@mail.nctu.edu.tw.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or direct commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or [permissions@acm.org](mailto:permissions@acm.org). © 2008 ACM 1084-4309/2008/09-ART66 \$5.00 DOI 10.1145/1391962.1391974 <http://doi.acm.org/10.1145/1391962.1391974>

ACM Transactions on Design Automation of Electronic Systems, Vol. 13, No. 4, Article 66, Pub. date: Sept. 2008.

## 1. INTRODUCTION

As Very Large Scale Integration (VLSI) technology enters the nanometer era, supply voltages continue to drop. This condition helps reduce power dissipation, but also decreases the noise margin of devices. Noise margin interference can sometimes generate erroneous chip functions, seriously reducing chip performance. As a result, the integrity problem has become one of the major factors affecting chip yield. Basically, the integrity issues can be categorized into signal integrity problems and power integrity problems. This article focuses on the power integrity problem caused by power supply noises such as the IR-drop and  $\Delta I$  (delta-I,  $Ldi/dt$ ) noise.

Many researchers have proposed various approaches to solving this problem in every design stage. The power/ground (P/G) network [Mitsuhashi and Kuh 1992; Tan and Shi 2001; Su et al. 2002; Singh and Sapatnekar 2005; Fu et al. 2005; Kahng et al. 2006] is an important factor in the supply noise problem. Power supply noise can be greatly improved by a better P/G network with minimal penalty cost. Besides sizing the power lines [Dutta and Marek-Sadowska 1989; Yan et al. 2004], employing decoupling capacitances (decaps) is a common approach to reducing supply noise. Traditionally, the decap insertion process is performed after routing in the physical design flow. This method would waste many unnecessary area of the decap budget to improve the noise and decrease the efficiency of the decap budget. Therefore, more and more researchers propose to insert the decap before routing. Yeh and Marek-Sadowska [2005] proposed a two-step decap insertion method to improve power supply noise in the placement level. This method includes one prediction method and one correction method. The prediction step estimates the required decap pessimistically. Although the decap size can be adjusted in the correction step, a smaller area overhead can be achieved if decap insertion can be considered at an earlier stage. Yan et al. [2005] and Zhao et al. [2002], proposed decap insertion methods at the floorplan level to reduce supply noise. Unfortunately, these previous researchers often overestimate the decap budget. They assume that the decap is able to fully supply the maximum current of the module, which is too pessimistic in our observation. Besides the decap budget computation, previous works do not fully use the available floorplan space. A floorplan usually has a lot of available spaces that can be used to insert the decap without increasing the floorplan area.

To make a high-performance and high pin-count IC, the area-array architecture is often used. In this architecture, the signal bumps are uniformly distributed over the chip. Therefore, the resistance from the core I/O to the signal bumps can be greatly reduced and larger number of I/Os can be accommodated. Chen et al. [2005] used this architecture in floorplanning to improve the power supply noise. Because the area-array architecture has such advantages, more and more chips adopt this architecture to improve the power supply noise and limited pin number problem. However, without decap insertion, the resulting floorplans and the area-array architecture still suffer from supply noise violations.

The purpose of this work is to develop a better model for calculating the decap required to solve the power supply noise problem and to wisely use the available

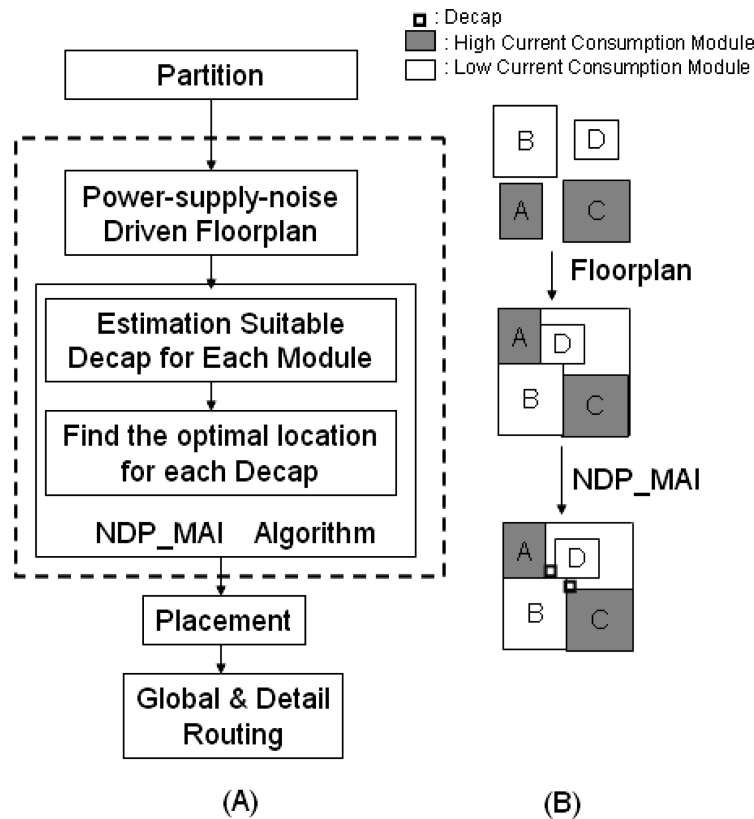


Fig. 1. Using a two-step approach that includes a Power Supply Noise (PSN) driven floorplanning algorithm and a decap insertion algorithm to improve the power supply noise before routing level. (A) The flow chart of our methodology. (B) In the floorplan step, high current modules are not abutted and we use minimal extensive area to insert the decap budget in decap insertion step.

space in the floorplan to reduce the area overhead caused by decap insertion. Based on the area-array architecture, this article proposes a two-step approach that includes a noise-driven floorplanning algorithm and a decap insertion approach to suppressing power supply noise at the floorplan level, as Figure 1 illustrates. First, we use a noise-driven floorplan algorithm to reduce the possible noise. This article adopts a stronger adjacent module relation O-tree representation as the engine for supply noise driven floorplanning, and successfully modifies the primary operations *Delete* and *Insert* in proposed framework. Second, we use a Noise-driven Decap Planning with Minimum Area Insertion (NDP\_MAI) approach to inserting minimal decaps into a noise-guided resultant floorplan, with blocks and decaps legalization. Note that this approach can compute the approximative decap size for a real design, and then provide the optimal location for each decap in floorplanning. After routing, we can use the method in Kahng et al. [2006] to rectify our result, further improving the power supply noise problem.

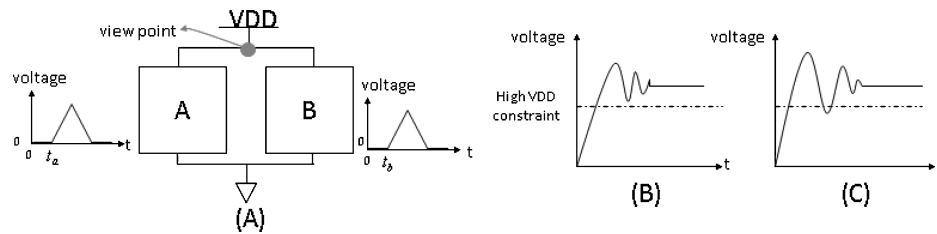


Fig. 2. The example of  $Ldi/dt$  noise effect. A complete circuit is composed of two subcircuits, **A** and **B**, as in (A). If the start time of switching activity for **A** and **B** is not the same, that is,  $t_a \neq t_b$ , the voltage can remain at a high level, as in (B). If not, the unstable voltage increases for a short period and the voltage drops below the high VDD constraint, as in (C).

The rest of this article is organized as follows: Section 2 describes the floorplan design with power supply noise consideration, the new noise estimation method, decap budget computation, and problem formulation. Section 3 presents the floorplanning algorithm and decap insertion approach for power supply noise avoidance. Section 4 shows experimental results, and Section 5 presents conclusions.

## 2. POWER DELIVERY AND SIGNAL INTEGRITY ISSUES IN AREA-ARRAY DESIGNS

As VLSI technology enters the deep submicron (DSM) era, chips contain more functions and are expected to have much better performance. At the same time, reduced supply voltages in modern chip design is tightening the noise margin. IR-drop and  $Ldi/dt$  (delta-I) noise are the main contributors in the noise margin issue, and this study focuses primarily on these sources of noise.

IR-drop is the unavoidable waste of electric charge when the circuit obtains energy from a power. However, as we move into DSM regime, the resistance of the connection wire affects the power consumption of the chip. The  $Ldi/dt$  noise is also called SSN (simultaneous switching noise) or  $\Delta I$  (delta-I) noise. Figure 2 shows that  $Ldi/dt$  noise is a voltage fluctuation phenomenon. When multiple circuits switch simultaneously, the circuit requires a lot of electric charges in one moment. If the VDD pin cannot supply enough energy at that time, the voltage fluctuation margin might exceed the lower boundary constraint. Using VDD pins and the decap to enhance the stabilization of the power supply are popular methods. This section describes our power delivery model, noise estimation model, and decap computation used in this article, and formulate our target problem.

### 2.1 Area-Array Power Delivery Model and Noise Estimation

In this article, the power source distribution is based on the area-array architecture. The area-array architecture is a mesh structure, and the VDD and GND bumps are uniformly distributed across the die with signal bumps in fixed interspersed location, as illustrated in Figure 3(A). The resistance from the I/O to the connection block is substantially decreased, therefore improves the performance of the power delivery. As a result, many high-performance chips adopt the area-array architecture.

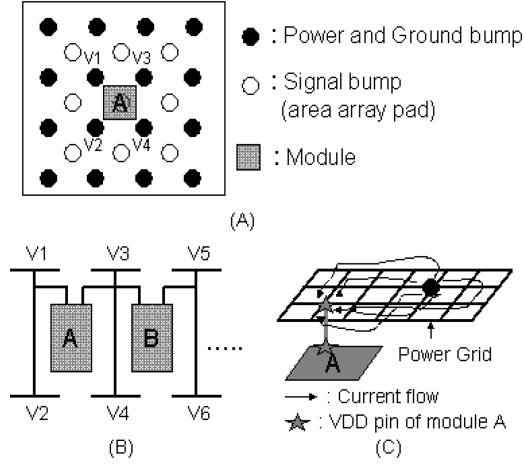


Fig. 3. Area-Array power delivery illustration. (A) Area-array footprint SoC. The VDD and GND bumps are uniformly distributed across the die with signal bumps in fixed interspersed locations. (B) One module gets power from only four VDD pins and others VDD pins are ignored. (C) The current path from one VDD pin to module A.

In the area-array architecture, a VDD bump supplies the current to all modules according to the direct proportion of the distance from the bump to the module. Four neighboring VDD bumps (right-top, right-down, left-top and left-down) of the module supply the main current, as Figure 3(B) shows, so we compute the noise from these VDD bumps only. Since there exist many paths for current delivery to the target module, we only consider the shortest and second shortest paths for noise computation simplification. The main reason is that currents follow the least-impedance paths when flowing from the VDD to the target module. Compared with this method with SPICE simulation, the error is within 10%, which is proved by Zhao et al. [2002]. This computation method is fast and the error can be controlled within tolerable range, so we use this method to compute the power supply noise in the floorplan level. Kirchhoff's voltage law can be used represent the noise calculation of each module:

$$V_{noise}^{(k)} = \sum_{P_j \in T^k} i_j R_{P_{jk}} + L_{P_{jk}} \frac{di_j}{dt}, \quad (1)$$

where  $V_{noise}^{(k)}$  denotes the power supply noise at module  $k$ ,  $P_j$  denotes the path from the power bump to node  $j$ ,  $P_{jk}$  denotes the path from node  $j$  to node  $k$ ,  $T^k$  denotes the union of shortest paths and the second shortest paths,  $R_{P_{jk}}$  denotes the resistance of  $P_{jk}$ ,  $L_{P_{jk}}$  denotes the inductance of  $P_{jk}$  and  $i_j$  is the current flowing along path  $P_j$ .

## 2.2 Decap Budgeting in Area-Array Architecture

Zhao et al. [2002] and Yan et al. [2005] assumed that the decap should fully supply the maximum current of the module, as shown in the white region in

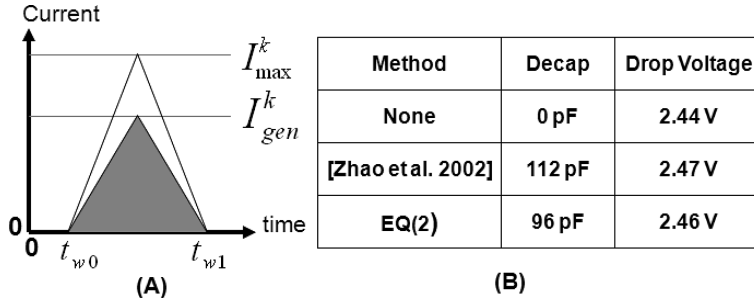


Fig. 4. (A) The current consumption profile of module  $k$ .  $I_{gen}^k$  is supplied by VDD pins only, and  $I_{max}^k$  is supplied by VDD pins and the decap budget. (B) We use HSPICE to verify our method and the approach used in Zhao et al. [2002], the simulation circuit is shown in Figure 2(A). We use inverters in Module A and Module B. In the simulation result, our method yields less required decap than Zhao et al. [2002].

Figure 4(A). In this environment, the decap budget is possibly over-estimated. Actually, the VDD pin continuously provides current when the chip is operating, as the grey region in Figure 4(A). Therefore, the required decap size can be significantly reduced.

The required decap size can be obtained by the difference between the maximum current ( $I_{max}$ ) and the target current limit ( $I_{gen}$ ) for each module. Assume the target current limit of module  $k$  is defined as  $I_{gen}^k$ ,  $k = 1, 2, \dots, M$ , and the maximum switching current of module  $k$  is  $I_{max}^k$ . Let  $C^k$  be the required decap for circuit  $k$  and  $Q^k$  be the amount of electric charge for the  $C^k$ .  $Q^k$  can then be obtained by the following equation based on the triangle model shown in Figure 4.

$$Q^k = \int_{t_{w0}}^{t_{w1}} I_{max}^k(t) dt - \int_{t_{w0}}^{t_{w1}} I_{gen}^k(t) dt, \quad (2)$$

where  $t_{w0}$  is the start time and  $t_{w1}$  is the finish time when the target module is in operational mode. The charge can be converted to the silicon area of the capacitance fabrication as follows:

$$C^k = \frac{Q^k}{V_{con}} \quad (3)$$

$$S_{decap}^k = \frac{C^k}{C_{ox}} \quad (4)$$

where  $V_{con}$  is the noise constraint of the voltage,  $C^k$  is the decap budget and  $S_{decap}^k$  is the silicon area of  $C^k$ .  $C_{ox}$  is the unit area capacitance of a MOS capacitor and  $C_{ox} = \epsilon_{ox}/t_{ox}$ , where  $\epsilon_{ox}$  is the permittivity of SiO<sub>2</sub> and  $t_{ox}$  is the oxide thickness. We use SPICE to verify the accuracy of (2), and compare the result with Zhao et al. [2002]. In the experiment, we use 0.25  $\mu m$  technology to do this simulation. The supply voltage is set at 2.5V and the power supply noise tolerance level is set at 0.04V.  $t_{w0}$  is set at 0 ns and  $t_{w1}$  is set at 24 ns. Adopting

the Zhao et al. [2002] method to compute the required decap produces the result of 112 pF. The decap budget is 96 pF when using (2). Figure 4(B) shows the simulation result. The proposed method yields less required decap than Zhao et al. [2002].

### 2.3 Problem Formulation

The goal of this article is to use minimum decap to solve the power supply noise problem in the floorplan level. In other words, we suppress power noise by each module in different locations and empty minimal decap area to avoid possible power noise during floorplanning. The problem can be formulated as follows:

*Given a set of modules,  $B_1, B_2, \dots, B_m$ , current consumption,  $I_{gen}^k$  and  $I_{max}^k$ , of each block  $B_k, 1 \leq k \leq m$ , a set of power bumps,  $P_1, P_2, \dots, P_m$ , and the noise constraint for each module  $V_{con}^k$ , find a feasible solution such that each module  $B_k$  obtains an appropriate and minimal decap budget size  $DBS_k$ , and minimum penalty area when  $DBS_k$  is inserted. At the same time, the voltage noise of module  $V_{noise}^{(k)}$  must be smaller than the noise constraint  $V_{con}^{(k)}$ .*

## 3. MINIMAL DECAP ALLOCATION IN POWER SUPPLY NOISE AWARE FLOORPLANNING

To solve the power supply noise problem, this article develops a two-step methodology to suppress and reduce noise at the floorplan level, as Figure 1 illustrates. Since placing two high current consumption modules close together seriously increases noise, we first propose a noise-driven floorplan algorithm to improve this issue; the idea is to place the high-current consumption modules intelligently. The goal of our floorplan averages the high power consumption block at one chip. This method can bring two benefits: (1) the peak noise can be improved; (2) the decap budget can be averagely planned at one chip. The empty room after floorplanning is small and dispersive. If many decaps are inserted into one particular region, the area of the floorplan may increase because the empty room does not have enough space for the decap. We then propose a Noise-driven Decap Planning with Minimum Area Insertion (NDP\_MAI) approach to reducing the power supply noise and area overhead after floorplanning. We briefly introduce the representation of O-tree and new needed operations, *Delete* and *Insert*, and then discuss the feasible region of the decap budget. Finally, we discuss the compensation of the decap budget due to the resistance of the power line.

### 3.1 O-Tree Based Power Supply Noise Aware Floorplanning

To obtain a better result for noise-driven floorplan, a suitable and controllable floorplan representation is needed. Table I compares six floorplan representations. We choose O-tree to be our representation, the main reason is that the adjacent relations can be directly obtained. High current consumption modules can be placed at a distance from each other.

The O-tree is composed of a horizontal tree and a vertical tree, as shown in Figures 5(B) and 5(D). The horizontal (vertical) tree uses  $\mathcal{T}(\alpha, \beta)$  to represent the data structure, as shown in Figures 5(C) and 5(E), where  $\mathcal{T}$  denotes the tree



Table I. Six Floorplan Representations Comparison

Floorplan Representation	Adjacent Relation	Solution Space [Wen and Yan 2002]	Operation	
			Delete	Insert
SP [Murata et al. 1995]	Not Good	$O((m!)^2)$	-	-
B*-tree [Chang et al. 2000]	Good	$O(m!2^{2m-2}/m^{1.5})$	$O(1)$	$O(m)$
O-tree [Guo et al. 1999]	Best	$O(m!2^{2m-2}/m^{1.5})$	$O(m)$	$O(m)$
TCG [Lin and Chang 2001]	Good	$O((m!)^2)$	$O(m^2)$	$O(m^2)$
CBL [Hong et al. 2001]	Good	$O(m!2^{3-3}/m^{1.5})$	$O(m)$	$O(m)$
DBL [Yan et al. 2005]	Good	$O(m!2^{m-1})$	$O(m)$	$O(m)$

The O-tree structure records stronger adjacent relations for each module, while the memory cost of DBL is minimal.

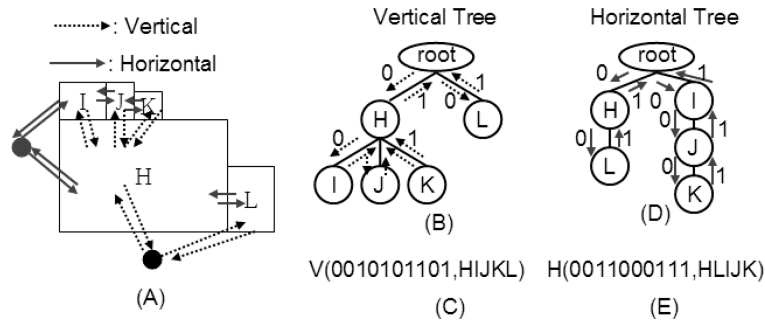


Fig. 5. An O-tree example. (A) One floorplan result. (B) Vertical tree of (A). (C) Vertical tree representation. (D) Horizontal tree of (A). (E) Horizontal tree representation.

type,  $\alpha$  denotes the paternity of the tree structure, and  $\beta$  denotes the permutation of modules. If the module touches another module horizontally (vertically), such as modules H and L in Figure 5, it could be easily observed in the horizontal (vertical) representation. If we use other representations, the adjacent relation of each module is more difficult to be found.

Figure 6(A) shows the original O-tree operations. If module J is deleted, the original *Delete* operation generates a LD-packing floorplan. The result is that two high-current modules (I and K) are placed at an adjacent location. In some special regions, they consume more power than other regions, and must use more decap to reduce power supply noise. A similar situation occurs for the *Insert* operation because it only considers the area and the wire length in original operation conditions. In accordance with the previous description, the original O-tree operations can not control the neighboring blocks. Therefore, new transformation operations are necessary. These operations help avoid placing the high current consumption modules at adjacent locations. We propose two new transformation operations:

- Delete*: The original operation deletes the selected module only. The new operation can delete the selected module and top-right modules of the selected module.
- Insert*: The original operation considers the area factor only. The module can be inserted into a low noise location and an extensive area can be minimized in our new operation.



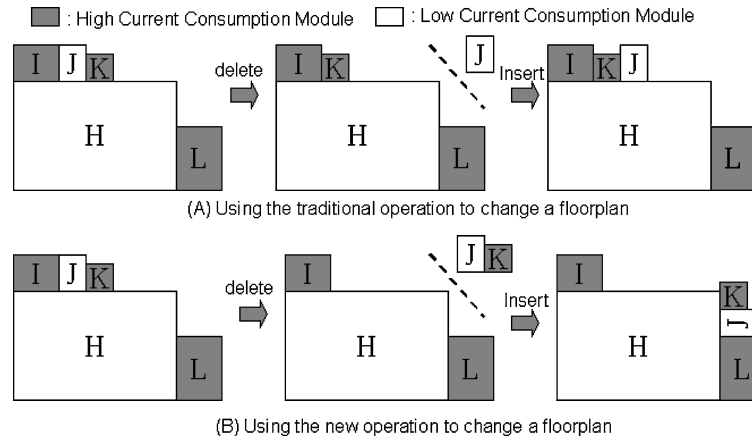


Fig. 6. The difference between operations in the original O-tree and our approach. The planned operation is to delete and to insert module J. If we use the original operations to change the floorplan in (A), high current consumption modules are possibly placed in the adjacent location. New operations can improve this problem, as (B), where high current consumption modules are not placed together.

We use an example to explain new transformation operations. To delete module J in Figure 6(B), the selected module is module J and the top-right modules of the selected module is module K only, so module J and module K must be deleted together. The reason that top-right modules must be deleted is because the floorplan must maintain a LD-packing result. If the right-top modules are not deleted at the same time, the high-current consumption module may be placed at a neighboring location.

The new *Delete* operation consists of several steps. We first choose a to-be-deleted module  $v$  from  $\beta$  of the horizontal and vertical O-trees, and then all modules after  $v$  in  $\beta$  are chosen. We could obtain two block sets  $\phi$  and  $\varphi$ . We then find the intersection of  $\phi$  and  $\varphi$  and obtain the candidate list of deletion modules. The final step in the *Delete* operation is to delete the modules at the intersection of the vertical and horizontal O-tree. To clarify, we use an example to explain our *Delete* operation. Figure 7 shows the horizontal and the vertical representations of Figure 6. The horizontal representation is set as (0011000111,HLIJK) and the vertical representation is set as (0010101101,HIJKL). In this case, module J must be deleted. Therefore, the block set of  $\phi$  includes modules J and K, and the block set of  $\varphi$  includes modules J, K and L. The deletion candidate list, JK, can be obtained after the intersection:  $JK \cap JKL$ . Finally, modules J and K in the representation are deleted. The horizontal representation changes to (001101,HLI), and the vertical representation is modified as (001101,HIL). The time complexity for the new *Delete* operation is  $O(m)$ , where  $m$  denotes the number of modules.

The new *Insert* operation consists of three parts: (1) find all possible locations; (2) compute costs; (3) choose the optimal location. If one module is inserted in a floorplan, there are many locations to choose from, and the first step is to discover these candidate locations in a LD-packing floorplan. The possible

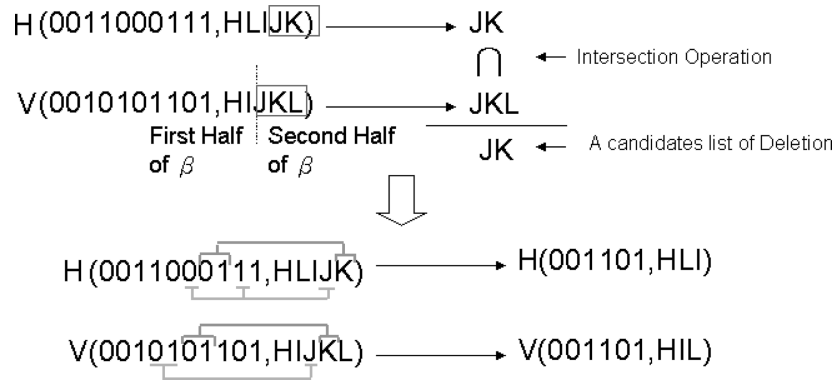


Fig. 7. Using the new *Delete* operation to delete module *J*. First, the candidate list can be obtained by applying intersection operation. Second, the original O-tree representation is modified until the corresponding O-trees do not show the candidate list.

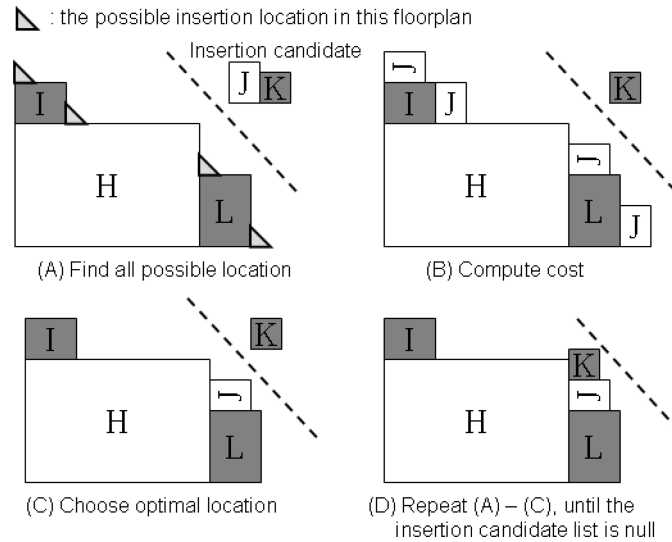


Fig. 8. Using the new *Insert* operation to insert all candidate modules. First, all possible insertion locations in one floorplan are found in (A). Second, the cost of the location is computed when the module is inserted in one of the possible locations. In (B), all location costs are computed when module *J* is inserted in the candidate location. The minimum cost location is chosen in (C). If the candidate list is not null, the *Insert* step must be repeated, as illustrated in (D).

insertion location is at the lower-left corner of the floorplan result, as shown in Figure 8. Every possible insertion location has a different cost, and next step is to compute the cost for each candidate location. The cost function can be represented as follows:

$$C_a = D_1(A_{new} - A_{original}) + D_2(I_a + I_b + I_c - I_{th}), \quad (5)$$

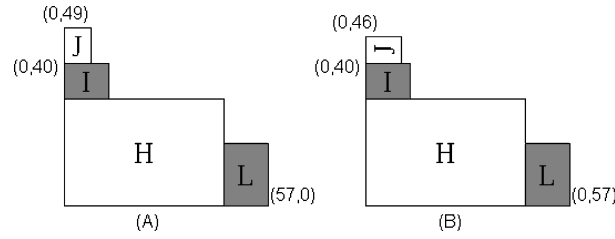


Fig. 9. The cost must compute twice for each corner because they are different during rotation.

where  $C_a$  denotes the cost when module A is inserted in this location,  $D_1$  and  $D_2$  are the weights,  $A_{new}$  is the area of the floorplan after the module is inserted,  $A_{original}$  is the original area,  $I_{a(b,c)}$  denotes the current consumption of the module  $a(b,c)$ , and  $I_{th}$  denotes the threshold value for local current consumption.  $D_2$  is set at a large value for penalizing high local current consumption. Every candidate location cost must be computed twice since costs are different when the insert module is directly inserted or rotated, as Figure 9 shows. Note that EQ(5) considers the area and the power consumption only, this cost computation can be extended by considering other objectives, such as wire length, etc. Finally, the module is inserted in the minimal cost location. The following illustration explains the new *Insert* operation. In Figure 8(A), the initial floorplan result was made up of modules I, H and L and modules J and K are the insert module candidates. Four triangles denote the candidate location in the floorplan. In Figure 8(B), we compute the cost after module J is inserted in all the candidate locations. Finally, module J is inserted in the minimal cost location. In this case, the minimal cost location is at the corner between module H and module L, as Figure 8(C) shows. Because the candidate list is not null, the *Insert* operation must be repeatedly applied, as shown in Figure 8(D). Note that EQ(5) considers left and down adjacent modules only when calculating the cost function. The main reasons for this are (1) Because the *Insert* operation must maintain a LD-packing floorplan, it only considers the left and down module of A; (2) The *Delete* operation deletes all the top-right modules of A. If all modules are inserted at the down-left corner only, there are no modules on the top and right side. The time complexity of the new *Insert* operation is  $O(m^2)$ , where  $m$  denotes the number of modules.

According to previous operations and the SA (Simulated Annealing) [Kirkpatrick et al. 1983] algorithm, we propose a power-supply noise-driven floorplan algorithm, as illustrated in Figure 10. We first provide a floorplan result and set the initial value for two annealing temperature parameters (line 2). The new *Delete* operation is used to delete modules (lines 4–6). Then, the new *Insert* operation and EQ(5) are used to insert modules (lines 7–11). After the *Insert* operation, the new LD-packing floorplan can be obtained. The difference ( $\Delta C$ ) between the new floorplan and the original floorplan is computed (line 12). If  $\Delta C$  is smaller than zero, it means that a better floorplan is obtained and we would adopt this result to be our new solution (line 14). If not, we randomly decide that the original floorplan must be replaced by the new floorplan (lines 15–17). Finally, the temperature is cooled (lines 18).

**Power-supply Noise-driven Floorplan Algorithm**

Input : A compacted floorplan,  $F$ , with the current consumption for each block

Output : A compacted floorplan,  $F$ , that two continuous high current consumption blocks could not placed to together

```

1. Begin
2.   Initial floorplan; Temperature; Final_Temperature;
3.   while Temperature > Final_Temperature
4.     Random choose one block  $B_x$  from  $F$ ;
5.     Using Delete operation to delete  $B_x$ ;
6.     All deletion blocks add into a candidate list;
7.     while candidate list is not NULL
8.       choose one block by from candidate list;
9.       Using EQ(6) and Insert operation to insert  $B_y$ ;
10.      Delete  $B_y$  candidate information;
11.    end_while
12.     $\Delta C = \text{Cost}(\text{New\_Floorplan}) - \text{Cost}(\text{Floorplan})$ ;
13.    if  $\Delta C < 0$ 
14.      Floorplan = New Floorplan;
15.    else if  $\text{Random}(0,1) > \exp(\frac{-\Delta C}{\text{Temperature}})$ 
16.      Floorplan = New Floorplan;
17.    end_if
18.    Cooling(Temperature);
19.  end_while
20. end

```

Fig. 10. The power-supply noise-driven floorplan algorithm.

### 3.2 Feasible Region for Decap Allocation

After obtaining one floorplan, we calculate the required decap size for each module. According to Murata et al. [1995], Chang et al. [2000], Guo et al. [1999], Lin and Chang [2001], Hong et al. [2001], and Yan et al. [2005], the empty room after floorplanning is small and dispersive. If a bigger decap is inserted into one floorplan, the area of the floorplan may increase because the unitary empty room does not have enough space for the decap. Besides the area factor, the charge/discharge time of the capacitance must be considered. The charge time is substantially reduced when several smaller decaps form the required decap. Our method cuts required decap into four smaller decaps to minimally increase the floorplan area and reduce the charge/discharge time of the decap. Note that the sizes of each smaller decap are not the same. The distribution is based on the Manhattan distance from the VDD source to the power bump and  $(P_x, D_x)$  denotes the connection relation.  $P_x$  denotes the Manhattan distance from the module's VDD location to the power bump  $x$  and  $D_x$  denotes the obtainable current contribution ratio from the power bump  $x$ . The computational equation of the current contribution ratio can be written as follows:

$$D_x = \frac{\frac{1}{P_{a,(a \neq x)}} + \frac{1}{P_{b,(b \neq x,a)}} + \frac{1}{P_{c,(c \neq x,a,b)}}}{P_x + \frac{1}{P_{a,(a \neq x)}} + \frac{1}{P_{b,(b \neq x,a)}} + \frac{1}{P_{c,(c \neq x,a,b)}}}, \quad (6)$$

where  $(a \neq x)$  denotes  $a$  and  $x$  are the different power bump.  $P_{a,(a \neq x)}$  denotes the power bump source of  $P_a$  and  $P_x$  are different. In accordance with Eq. (6),

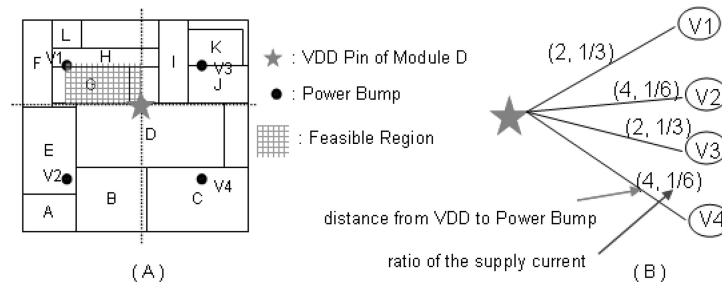


Fig. 11. The planned decap budget is partitioned into four smaller decaps. Each smaller decap is given a feasible region that ranges from the location of the power bump to the VDD source, as in (A). For each power bump, we set the supply current according to the distance from the power bump to the VDD source, as in (B). Based on these constraints, a smaller decap can be inserted into the chip and the charge time of the capacitance can be substantially decreased.

$D_x$  is inversely proportional to  $P_x$ , it is to follow the current divided theorem. A simple example helps to explain Eq. (6). Figure 11(A) shows a result of the floorplan. Block D needs decap to supply the current consumption. We first use Eqs. (2)–(4) to compute the optimal decap sizing. We then use Eq. (6) to compute the current ratio for each power bump, as Figure 11(B) shows.

In modern chip design, the decap is inserted in the empty space after detailed routing. In reality, the decap can not improve the power noise for the high-current consumption module if the distance from the decap to the module is far. To effectively utilize the energy for each decap, the rectangle scope from the power bump to the VDD pin is the feasible region for each small decap, as shown in Figure 11(A).

### 3.3 Identification of Space Priority for Decap Insertion

After floorplanning, the chip has some exploitable space for decap insertion. If these spaces can be fully utilized, the cost of the chip might not increase even when the decap is inserted into the chip. This section discusses the effect of placing the decap in each different space. Furthermore, we propose a Noise-driven Decap Planning with Minimum Area Insertion approach that simultaneously considers the area cost and the noise effect.

In a floorplan result, it certainly has one or more horizontal (vertical) longest paths. The path notes the maximum width (height) of the floorplan. As shown in Figure 12, module H and L compose a horizontal longest path. Varying these modules directly modifies the area. Therefore, if one decap is inserted in the channel space between these modules, the area would increase significantly. This channel space is called the extensive space. A channel space that overlaps edge of the empty room is called the empty space that is not held by any module. If one decap is inserted in this space, the location of each module does not change. The remaining channel spaces are called the available space except for the channels of the extensive and empty space. If one decap is inserted in this space, the area of the floorplan is fixed and the location of some modules shifts only slightly. In Figure 12, the horizontal longest path is  $H \rightarrow L$  and the vertical longest path is  $H \rightarrow I$ . If one decap is inserted into the extensive

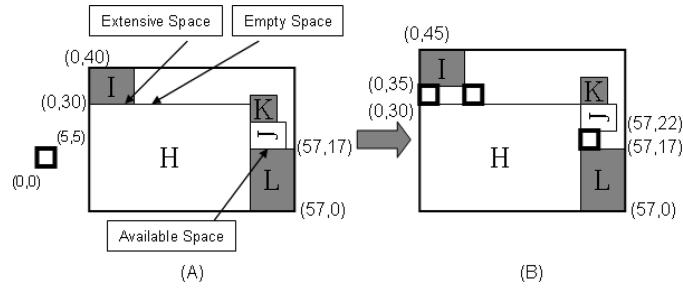


Fig. 12. When the decap is inserted into different spaces in a floorplan, the area of the floorplan might increase. (A) The area of the floorplan is  $2280 \mu m^2$  before the decap insertion. (B) The area of the floorplan is  $2565 \mu m^2$  after the decap insertion. The area does not increase when the decap is inserted into the empty space. If the decap is inserted into the available space, the area increases sometimes.

space between  $H$  and  $I$ , the area would increase  $285 \mu m^2$ . If one decap is inserted into the empty space corner between  $H$  and  $I$ , the area and the topology are not affected. Hence, the cost is lowest when the decap is inserted in the empty space. If the decap is inserted into the available space between  $L$  and  $J$ , the area is not increased but the topology changes. Therefore three types of spaces are the candidates location for the decap—Available Space, Extensive Space and Empty Space. The priorities of these spaces are defined as follows: *Empty Space* > *Available Space* > *Extensive Space*. The minimum cost space will be selected for the decap insertion. Figure 13 illustrates the NDP\_MAI approach.

### 3.4 Decap Compensation for Voltage Drop in Power Network

When using Eqs. (2)–(4) to calculate the required decap for each module, the decap must be placed around the target. The NDP\_MAI approach does not consider this factor when placing the decap. If the location of the decap is not near the target, the power supply noise violates the given constraint because a part of the supply power from the decap would be consumed by wire resistance. To improve this problem, we use a simple compensative computation as follows:

$$Q_{com}^k = Q^k + V(l \times c) \quad (7)$$

where  $Q_{com}^k$  is the required decap of module  $k$  after the compensation,  $V$  is the supply voltage,  $l$  is the distance from the space to the connection point, and  $c$  is the wire capacitance per unit length. Although we could compensate the power consumption of the wire by Eq. (7), the power network is another important issue that affects the supply power of the decap. Figure 14(A) shows the power network after the decap insertion. We could utilize the superposition theorem to analyze the circuit, as Figures 14(B) and 14(C) shows. The discharge current from the decap disperses to different modules because they both depend on the same power network. If decap budget computations do not consider this factor, the power supply noise constraint may be violated.

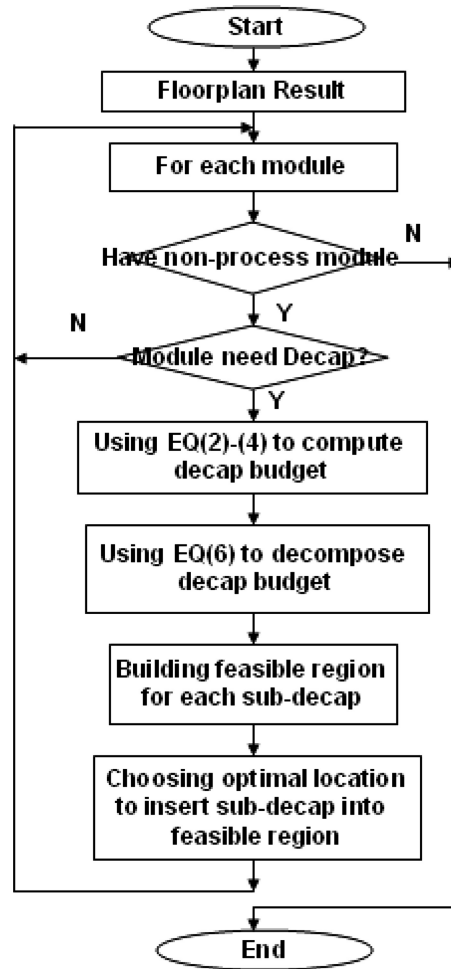


Fig. 13. The NDP\_MAI flow chart. Using this approach, the module obtains sufficient energy and uses minimal extended area to finish the decap insertion.

To solve the above problem, we need a more accurate compensation equation for Eq. (7). In accordance with the current divided theorem,

$$Q_{com}^k = Q^k \times \frac{R_{sk} + R_{\overline{sk}}}{R_{\overline{sk}}} + V(l \times c), \quad (8)$$

where  $R_{sk}$  is the total resistance on the side of the module  $k$  and  $R_{\overline{sk}}$  is the total resistance of other sides (Figure 14(B)). Equation (8) can accurately compensate the required decap. We use SPICE to verify the accuracy of these two compensative equations. In our experiment (the circuit in Figure 14), these modules are of the same resistance. We expect module B to obtain  $2.75 \text{ mA}$  from decap. If we use Eq. (7) to correct the decap, module B obtains only  $2.66 \text{ mA}$ , which is insufficient for module B. If we use Eq. (8) to adjust the decap, it obtains  $2.78 \text{ mA}$  from decap. This experiment shows that the module can obtain sufficient current



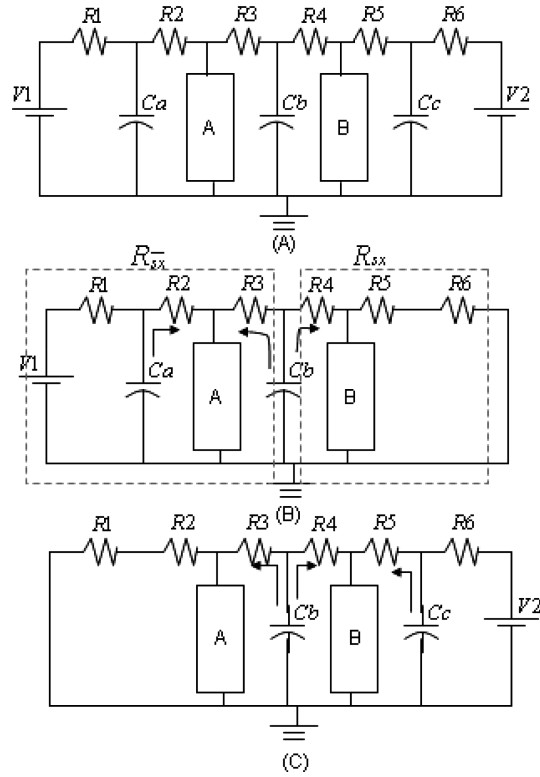


Fig. 14. Circuit analysis for the power network. (A) denotes the power network after the decap insertion and it has two VDD sources. Using superposition theorem in (A), the decomposed results appear in (B) and (C).

from the decap when we use Eqs. (2)–(4) and Eq. (8) to compute the required decap.

#### 4. EXPERIMENTAL RESULTS

This article implemented the Power-supply Noise-driven floorplan algorithm, the NDP\_MAI approach, and the approach in [Zhao et al. 2002] using C++ language on an AMD 3200 computer with 1G memory. Tables II and IV compare the run-time, peak noise, and decap budget with Zhao et al. [2002]. To obtain an equivalent comparison, the original cost function of Zhao et al. [2002] is “ $(A + \delta A) + \lambda_1(W + \delta W)$ ”, where “ $\lambda_1(W + \delta W)$ ” denotes the wire length cost. We set  $\lambda_1$  as zero because the wire length is not considered in our cost function. Five MCNC benchmark circuits, apte, hp, xerox, ami33 and ami49 are used to test the performance of proposed methodology. Since the MCNC benchmark includes no noise constraint, the noise constraint is set at 0.13V and 0.25V. In our experiments, the operation times  $t_{w0}$  and  $t_{w1}$  of the switching current waveform are set to be 0.3 and 0.8. The power supply voltage is 1.2V and the distance between two continuous VDDs is  $1000\Omega/\mu m$  and the power supply mesh is  $333.3\Omega/\mu m$ . We use Zhao et al. [2002]’s method to generate the current

Table II. Peak Noise after Floorplanning and the Decap Insertion

Circuit	Our Method			[Zhao et al. 2002]		
	Peak Noise(V) (noise-driven)	Peak Noise (V)(post)	Run Time(s)	Peak Noise(V) (noise-driven)	Peak Noise (V)(post)	Run Time(s)
apte	2.05	0.13	2	2.05	0.11	< 1
hp	1.62	0.13	4	1.63	0.11	< 1
xerox	1.60	0.13	4	1.61	0.10	< 1
ami33	0.29	0.12	12	0.38	0.09	2
ami49	0.12	0.12	16	1.70	0.10	5

Our floorplanner can provide better noise and the result of our decap insertion method conform the given constraint.

Table III. Our Estimation Requires Less Decap to Suppress Noise

Circuit	Our Decap Budget(nF)	Our Peak Noise(V)	[Zhao et al. 2002] Decap Budget(nF)	[Zhao et al. 2002] Peak Noise(V)	Decrease Decap
apte	14.77	0.13	22.12	0.11	33%
hp	3.6	0.13	5.01	0.11	28%
xerox	6.53	0.13	9.55	0.10	31%
ami33	0.17	0.12	0.36	0.09	52%
ami49	7.85	0.12	14.54	0.10	46%

Decaps obtained from our model and Zhao et al. [2002] are on the same resultant floorplan, and we use Eq. (1) to compute the peak noise.

consumption information of each module:  $I_{gen}^k$  denotes the target current limit of module  $k$ ,  $I_{gen}^k = A_k \times D_c$ , where  $A_k$  is the area of module  $k$ , and  $D_c$  is the worst case current density;  $I_{max}^k$  denotes the maximum current of module  $k$  and it is assigned as a random value to be  $1.05I_{gen}^k \sim 2I_{gen}^k$ .

Table II compares our method with Zhao et al. [2002], showing the peak noise at noise-aware floorplanning (noise-driven) and the post-floorplanning decap insertion (post). Experimental results show that our floorplan method obtains better results than Zhao et al. [2002] does. The main reason for this is that the high current consumption modules are placed apart when we use Eq. (1) to compute the peak noise. The time complexity of our floorplan method is slightly higher than the method in Zhao et al. [2002]. There are two reasons: (1) The time complexity of the original **Insert** operation is  $O(m)$  and the new **Insert** operation is  $O(m^2)$ . The time complexity of new operations is higher than original O-tree operations; (2) Zhao et al. [2002] use the sequence pair-based floorplanner to plan blocks. The sequence pair method modifies the list order to change the floorplan result. The time complexity for each change should be lower than the original O-tree method. In the post-floorplanning decap insertion, all results conform to the given constraint, 0.13V, and both run-times are very fast.

Table III clearly shows that our computation methodology obtains better decap budgets. For ami33 and ami49, the required decap budgets are reduced by 52% and 46%, respectively, and the noise conforms to the given constraint. In average, this approach reduces the decap budget by 38%. The main reason for this is that Zhao et al. [2002] ignores the supply current from VDD pins, and the total charge is supplied by the decap when the power supply noise occurs. This table also compares the peak noise. In all benchmarks, our peak noise is lower

Table IV. Experimental Results for Some MCNC Benchmarks with Various Approaches for Comparison

Circuit	Modules	Zhao et al. [2002] Increased Area ( $\mu m^2$ )	Zhao et al. [2002]'s Decap + Our Insertion Increased Area ( $\mu m^2$ )	Our Methodology Increased Area ( $\mu m^2$ )
apte	9	356832	292618	19036
hp	11	76608	68184	157640
xerox	10	152061	121648	75006
ami33	33	8228	5082	3824
ami49	49	266616	201486	154990

The third column result and the fourth column result shows the advantage of our decap insertion method. The fourth column result and the fifth column result shows the advantage of our proposed framework (including floorplan, decap computation and insertion).

than the constraint level. On the other hand, we use the real circuit to verify the accuracy of our experimental result. Simulations have been performed on ami33 with HSPICE. The peak noise before decap insertion is 0.29V. If we use our proposed method, the peak noise is 0.11V. After applying Zhao et al. [2002]'s method for decap insertion, the peak noise is 0.06V. These results are close to our results in Table III.

Table IV compares the area information of the proposed methodology with Zhao et al. [2002]. In the third column, we completely use the Zhao et al. [2002] method to compute the increased area. In the fourth column, we partially adopt the Zhao et al. [2002] method (including floorplan and decap computation) and our decap insertion method to obtain the increased area. The third column and the fourth column show the advantage of our decap insertion method. Our increased area reduction comes from the fact that our method distributes computed decap into four smaller decaps to minimally increase the floorplan area. The incremental area of our proposed method is shown in the last column. Compared to the numbers reported in previous works, the proposed floorplanning framework creates better initial floorplans to work on, followed by the effective NDP\_MAI approach to inserting enough decaps. In addition, we consider incorporating the proposed decap budgeting and insertion into the floorplaner, and compare that with our method. The experiment has been performed on ami49, in our method the peak-noise is 0.12V, the run time is 12 seconds (shown in Table II) and the increased area is 154990 ( $\mu m^2$ ), but in integrated method the peak-noise is 0.10V, the run time is increased to 921 seconds and the increased area is 112652 ( $\mu m^2$ ). The results in both peak-noises conform the constraint, and the integrated method can obtain better increased area reduction. Unfortunately, the penalty of the run time is very huge.

## 5. CONCLUSION AND FUTURE WORK

The techniques of floorplanning and decap insertion can be used to reduce power supply noise in early design stages. The proposed framework can allocate a suitable decap size. Furthermore, based on floorplan analysis and space priority, the decap insertion approach is proposed to insert the decap budget very effectively. Experimental results show that the proposed method improves the increase ratio of the floorplan area when inserting decaps.

In the modern chip design, the power gating technology [Hailin et al. 2005; Chiou et al. 2006] is frequently used to improve the leakage current. Therefore, a parallel process of the power gating and the decap will be explored to further improve the decap budgeting and insertion approach.

#### ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for providing valuable comments and suggestions that greatly improved this article.

#### REFERENCES

- CHANG, Y.-C., CHANG, Y.-W., WU, G.-M., AND WU, S.-W. 2000. B\*-trees: A new representation for nonslicing floorplans. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 458–463.
- CHEN, H.-M., HUANG, L.-D., LIU, I. M., AND WONG, D. 2005. Simultaneous power supply planning and noise avoidance in floorplan design. In *IEEE Trans. Computer-Aided Des. Integr. Circ. Syst.* 578–587.
- CHIOU, D., CHEN, S., CHANG, S., AND YEH, C. 2006. Timing driven power gating. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 24–28.
- DUTTA, R. AND MAREK-SADOWSKA, M. 1989. Automatic sizing of power/ground (P/G) networks in VLSI. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 783–786.
- FU, J., LUO, Z., HONG, X., CAI, Y., TAN, S. D., AND PAN, Z. 2005. VLSI on-chip power/ground network optimization considering decap leakage currents. In *Proceedings of the IEEE Asia and South Pacific Design Automation Conference*. IEEE Computer Society Press, Los Alamitos, CA, 735–738.
- GUO, P.-N., CHEN, C.-K., AND YOSHIMURA, T. 1999. An O-tree representation of nonslicing floorplan and its application. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 268–273.
- HAILIN, J., MAREK-SADOWSKA, M., AND SANI, R. 2005. Benefits and costs of power-gating technique. In *Proceedings of the IEEE International Conference on Computer Design*. IEEE Computer Society Press, Los Alamitos, CA, 559–566.
- HONG, X., HUANG, G., CAI, Y., GU, J., DONG, S., AND CHENG, C. 2001. Corner block list: An effective and efficient topological representation of non-slicing floorplan. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 764–769.
- KAHNG, A. B., LIU, B., AND TAN, S. X.-D. 2006. Efficient decoupling capacitor planning via convex programming methods. In *Proceedings of the International Symposium on Physical Design*. 102–107.
- KIRKPATRICK, S., GELATT, C., AND VECCHI, M. 1983. Optimization by simulated annealing. *Science*. 671–680.
- LIN, J.-M. AND CHANG, Y.-W. 2001. TCG: A transitive closure graph-based representation for general floorplans. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York.
- MITSUHASHI, T. AND KUH, E. 1992. Power and ground network topology optimization for cell based VLSIs. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 524–529.
- MURATA, H., FUJIYOSHI, K., NAKATAKE, S., AND KAJITANI, Y. 1995. Rectangle-packing based module placement. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*. ACM, New York, 472–479.
- SINGH, J. AND SAPATNEKAR, S. 2005. Congestion-aware topology optimization of structured power/ground networks. *IEEE Trans. Computer-Aided Des. Integr. Circ. Syst.* 683–695.
- SU, H., SAPATNEKAR, S., AND NASSIF, S. 2002. An algorithm for optimal decoupling capacitor sizing and placement for standard cell layouts. In *Proceedings of the International Symposium on Physical Design*. 68–73.

- TAN, X. AND SHI, C. 2001. Fast power/ground network optimization based on equivalent circuit modeling. In *Proceedings of the IEEE/ACM Design Automation Conference*. ACM, New York, 550–554.
- WEN, S.-J. AND YAN, J.-T. 2002. Double-bound list: A new placement representation with application to simulated-annealing-based floorplan. Unpublished.
- YAN, J.-T., LIN, K.-P., AND CHEN, Y.-H. 2005. Decoupling capacitance allocation in noise-aware floorplanning based on DBL representation. In *Proceedings of the International Symposium on Circuits and Systems*. 23–26.
- YAN, J.-T., LU, C.-H., AND WU, C.-W. 2004. Simultaneous wiring and buffer block planning with optimal wire-sizing for interconnect-driven floorplanning. In *Proceedings of the IEEE Midwest Symposium on Circuits and Systems*. IEEE Computer Society Press, Los Alamitos, CA, 25–28.
- YEH, C. AND MAREK-SADOWSKA, M. 2005. Timing-aware power noise reduction in layout. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*. ACM, New York, 627–634.
- ZHAO, S., ROY, K., AND KOH, C. 2002. Decoupling capacitance allocation and its application to power-supply noise-aware floorplan. *IEEE Trans. Computer-Aided Des. Integr. Circ. Syst.* 21, 81–92.

Received May 2007; revised September 2007 and January 2008; accepted May 2008