JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 24, 115-127 (2008)

Short Paper

An Effective Decap Insertion Method Considering Power Supply Noise during Floorplanning*

CHAO-HUNG LU, HUNG-MING CHEN** AND CHIEN-NAN JIMMY LIU
Department of Electrical Engineering
National Central University
Taoyuan, 320 Taiwan
E-mail: {chlu; jimmy}@ee.ncu.edu.tw
**Department of Electronics Engineering and SoC Research Center*
National Chiao Tung University
Hsinchu, 300 Taiwan
E-mail: hmchen@mail.nctu.edu.tw

As VLSI technology enters the nanometer era, the supply voltage is continually dropped. This condition helps to reduce the power dissipation, but make the power integrity problem become worse. Employing decoupling capacitances (decap) at floorplan stage has been a common approach to alleviate the supply noise problem. However, the decap budget is often overly estimated in previous researches. Besides the decap budget computation, the available floorplan space does not fully used in previous works. In one floorplan, it usually has many available spaces except the empty space that could be used to insert the decap without increasing the floorplan area. Therefore, our goal in this work is to develop a better model to calculate the required decap to solve the power supply noise problem of area-array based designs and to increase he usage of the available space in the floorplan to reduce the area overhead caused by decap insertion. The experimental results are encouraging. Compared with other approaches, our algorithm can reduce 52.6% decap budget on the MCNC benchmarks but still keep the power supply noise in the given constraint. The final floorplan areas with decap are also less than the numbers reported in previous papers.

Keywords: decoupling capacitance, floorplanning, physical design, power estimation, area-array architecture

1. INTRODUCTION

As VLSI technology enters the nanometer era, the supply voltage is continually dropped. This condition helps to reduce the power dissipation, but also decrease the noise margin of the devices. Therefore, the integrity problem has become one of the major factors that affect chip yield. Basically, the integrity problems can be categorized into the signal integrity problem and the power integrity problem. In this paper, we focus on the power integrity problem that caused by the power supply noises, such as the IR-drop and

Received February 2, 2007; accepted July 13, 2007.
Communicated by K. Robert Lai, Yu-Chee Tseng and Shu-Yuan Chen.
* This work was supported in part by the National Science Council of Taiwan, R.O.C. under contract NSC 96-2220-E-009-016 and NSC 96-2221-E-008-120.

ΔI (delta-I , Ldi/dt) noise. Many researches have proposed various approaches to solve this problem at every design stages. The power/ground (P/G) network [4-8] is an important factor in the supply noise problem. The power supply noise can be greatly improved by a better P/G network and with minimal penalty cost. In [9, 10], the authors use optimal wire sizing to increase the integrity of the signals.

Besides sizing the power lines, employing decoupling capacitances (decap) has been a common approach to reduce the supply noise. Traditionally, the decap insertion process is performed after routing in the physical design flow. Unfortunately, the area of the decap is greatly increased because the module placement has been fixed. Therefore, more and more researches propose to insert decap before routing. In [11], the authors propose a two-step decap insertion method to improve power supply noise in placement level. This method includes one prediction method and one correction method. In the prediction step, the required decap is pessimistically estimated. Although the decap size can be adjusted in the correction step, it is still possible to obtain smaller area overhead if the decap insertion can be considered at earlier stage.

In [1] and [2], the authors propose the decap insertion methods at floorplan level to reduce supply noise. Unfortunately, the decap budget is often overly estimated in previous researches. They assume that the decap must be able to fully store the maximum current of the module, which is too pessimistic in our observation. Besides the decap budget computation, the available floorplan space does not fully used in previous works. In one floorplan, it usually have many available spaces except the empty space that can be used to insert the decap without increasing the floorplan area. Therefore, our goal in this work is to develop a better model to calculate the required decap to solve the power supply noise problem and to increase the usage of the available space in the floorplan to reduce the area overhead caused by decap insertion.

In recent high-performance IC manufacturing, the flip-chip and area-array architectures [18] are often used. The traditional location of the I/O Pad is set around the core. In the area-array architecture, the I/O Pad is set over the core. Because the I/O Pads are distributed over the chip, the packaging issue should be considered in floorplan stage to reduce the distance from the core I/O to the signal bumps. In [12], the authors provide a power bump allocation method to effectively reduce the distance with from the supply voltage nodes to modules, thus reducing power supply noise. However, without decap insertion, resultant floorplans may still suffer from supply noise violations.

Based on the area-array architecture, we propose a two-step approach in this paper that includes a noise-driven floorplanning algorithm and a decap insertion algorithm to suppress power supply noise at the floorplan level, as illustrated in Fig. 1. First, we use a noise-driven floorplan algorithm to reduce the possible noise. The power supply noise would be substantially increased when two high-current blocks are placed at the adjacent locations. Therefore, high current consumption blocks are not abutted in our floorplan algorithm. The second step is the decap insertion method. We use a Noise-driven Decap Planning with Minimum Area Insertion (NDP_MAI) algorithm to reduce the noise after floorplanning. In this step, given an initial compacted floorplan and the current consumption for all blocks, the NDP_MAI algorithm can calculate the minimal decap budget and determine the locations of the inserted decap for each block to satisfy the noise constraints.

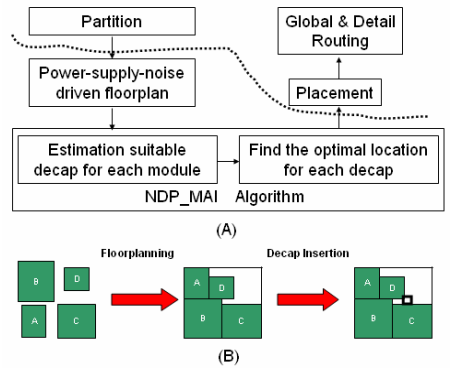


Fig. 1. The illustration of our two-step approach.

The key contributions of this paper are summarized as follows: (1) We have developed a better noise estimation model that can obtain less decap area compared with previous approaches; (2) We use the O-tree representation with strong adjacent module relation as our engine for noise-driven floorplanning, and successfully modify two operations *Delete* and *Insert* to avoid high-current blocks being placed at adjacent locations; (3) We have presented an algorithm to increase the usage of the available space in the floorplan for the required decap to meet the noise constraint.

The rest of the paper is organized as follows. Section 2 presents the problem formulation. Section 3 describe our noise estimation method and decap budget computation. The floorplanning approach and decap insertion algorithm are presented in section 4. Experimental results are shown in section 5. We conclude the paper in section 6.

2. PROBLEM FORMULATION

In the traditional chip design, the decap insertion method is used to solve the power supply noise problem after the routing level. In this paper, we propose to solve this problem in early design stage. Given a set of module, B_1, B_2, …, B_m, I_{gen}^k and $I_{\max}^k$, of each block , of each block B_i, the locations for each VDD source and the noise constraint for each module, find a feasible solution such that each block B_i obtains appropriate and minimal decap budget size *DBSi*, and minimum penalty area when *DBSi* is inserted.

3. ESTIMATING POWER NOISE AND REQUIRED DECOUPLING CAPACITANCE

3.1 Power Delivery Model and Noise Estimation

In this paper, the power source distribution is based on the area-array architecture [18]. 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 Fig. 2 (A). In designing chips, the I/O locations will greatly improve the performance than the traditional design. If the area-array architecture is used, the distance

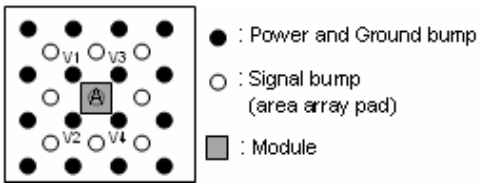


Fig. 2. Area-array footprint SoC.

from the core I/O to the connection point of the package would be substantially decreased and the performance would be significantly improved. Therefore, the area-array architecture is extensively used in high-performance chips.

In the general chip design, any VDD bump would supplies the current to all modules according to the direct proportion of the distance from the bump to the module. In the area-array architecture, four neighboring VDD bumps (right-top, right-down, left-top and left-down) of the module would supply the main current. Therefore we compute the noise from these VDD bumps only. Because the power network is a mesh grid, the VDD bumps would follow the grid to supply the current. If we compute the noise for each current path, we would expense great times to obtain the accurate result. In [2], it has an experimental result about the computation of the path. If we calculate the noise from four neighbor VDD bumps and consider the shortest paths and the second shortest paths only, the error is less than 10% (compared with HSPICE). This computation method is fast and the error could be control in one expectable range. We use this method to compute the power supply noise in the floorplan level. We can use Kirchhoff's voltage law to represent the noise calculation for each module:

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

where $V_{noise}^{(k)}$ denotes the power supply noise at module k, P_j denotes the path from VDD to node j, 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.

3.2 Decoupling Capacitance Budget Computation

In [1] and [2], the authors assume that the decap should fully supply the maximum current of the module, as shown in the white region in Fig. 3. Based on this environment, the decap budget would be overly estimated. Actually, the VDD pin continually provides a current as the gray region in Fig. 3 when the chip is running. Therefore, the required decap size can be significantly reduced.

The required decap size can be easily 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}, $k = 1, 2, \ldots, M$, 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 is the amount of electric charge of the C^k. Then Q^k can be obtained by the following equation based on the triangle model shown in Fig. 3.

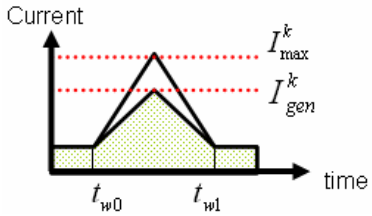


Fig. 3. Switching current consumption profile of module k.

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

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

$$C^k = Q^k / V_{con} \quad (3)$$
$$S_{decap} = C_{decap} / C_{ox} \quad (4)$$
$$C_{ox} = \varepsilon_{ox} / t_{ox} \quad (5)$$

where V_{con} is the noise constraint of the voltage, C_{decap} is the decap budget and S_{decap} is the silicon area of C_{decap}. In order to verify our decap calculation model, we perform a simple experiment using SPICE and compare the required decap with [2]. In our experiment, the supply voltage is set to be 2.5V and the power supply noise limit is set to be 0.04V. If we adopt the [2] method to compute the require decap, the computation result is 112pF. The decap budget is 96pF if using Eq. (2) to compute. Our method could obtain the less required decap than [2].

4. SUPPLY NOISE AWARE FLOORPLANNING WITH MINIMAL DECAP INSERTION

4.1 O-tree Based Power Supply Noise Aware Floorplanning

To obtain the better result of one noise-driven floorplan, a suitable and controllable floorplan representation is needed. We compare six floorplan representations, SP, B*-tree, O-tree, TCG, CBL, DBL. Finally, we choose O-tree to be our representation. The main reason is the adjacent relation could be directly obtained. Therefore, the high current consumption modules could be easily placed at a distance.

O-tree is composed of a horizontal tree and a vertical tree, as shown in Figs. 4 (b) and (d). The horizontal (vertical) tree could use $\tau(\alpha, \beta)$ to represent the data structure, as shown in Fig. 4 (c). τ denotes the tree type and α denotes the paternity of the tree structure, and β denotes the permutation of modules. If the module has horizontally (vertically) touch with another module, such as the modules H and L in Fig. 4, it could be easily observed in the horizontal (vertical) representation. If we use another representation, the adjacent relation of each module must spend high cost to find. Therefore, we adopt the O-tree to be our floorplan representation.

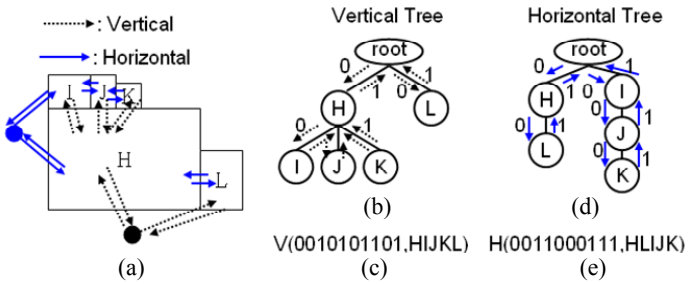


Fig. 4. An O-tree example, (a) One floorplan result; (b) Vertical tree of A; (c) Vertical tree representation; (d) Horizontal tree of A; (e) Horizontal tree representation.

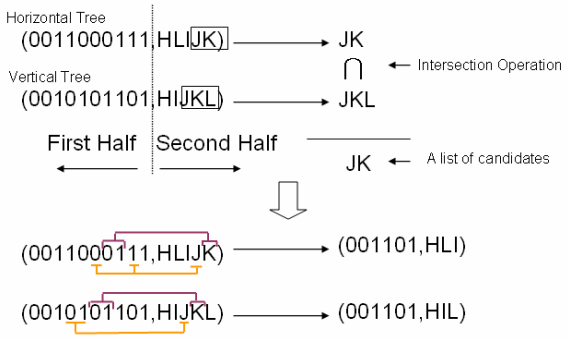


Fig. 5. Using new *delete* method to delete the module J.

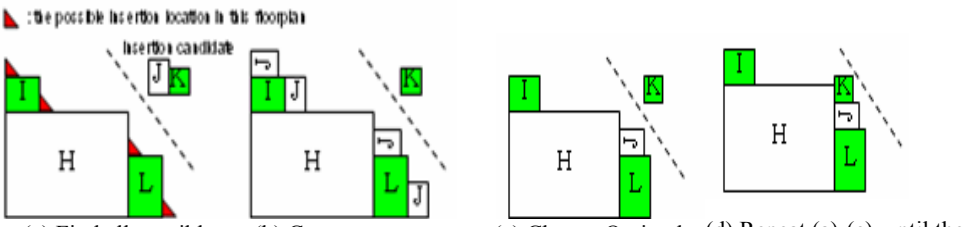
The original O-tree operation is shown in Fig. 5 (a). If module J is deleted, the original *Delete* operation would control remaining modules to make a LD-packing floorplan. Two high-current modules (I and K) are placed at a tight location. Therefore, the current consumption of the chip would not be balance. In the special region, it would consume more power than other regions and must uses more decap to solve power supply noise. The similar situation is happened at the *Insert* operation because it only considers the area and the wire length in original operation. According to the previous description, we know the original Otree operations could not control the neighbor block. Therefore, we need new transformation operations. These operations could avoid the high current consumption modules be placed at the continual location. Naturally, our decap insertion approach could averagely plan the decap location according to the result of the floorplan. We propose two new transformation operations:

– *Delete*: The original operation delete the selected module only. The new operation would delete the select module and top-right modules of the select center together.

– *Insert*: The original operation consider area factor only. The module would be inserted into the low noise location and the extensive area could be minimized.

Our *Delete* operation could be divided into several steps. Firstly, we choose the delete module v from the β location of the horizontal and vertical representation. Further,

all modules after v in β are chosen. We could obtain two block set ϕ and φ. Therefore, we use Intersection Operation to compute the result of ϕ and φ and obtain the candidate list of deletion modules. The original representation has the data of all modules. The final step of our *Delete* operation is to continuously delete the representation until the data of the delete module list is not exist in the vertical and horizontal representations. Notice that the data of α and β should be delete at the same time. To clearly explain, we use an example to explain the *delete* operation. The horizontal and the vertical representation are shown in Fig. 5. The horizontal representation is set as (0011000111, HLIJK) and the vertical representation is set as (0010101101, HIJKL). In this case, we choose module J to be the delete module, v. Therefore, the block set of ϕ includes module J and K, and the block set of φ includes module J, K and L. The delete candidate list, JK, could be obtained after these two block sets to be Intersection Operation, JK\JKL. Finally, the data of module J and K in the representation are deleted and the result of horizontal representation is changed as (001101, HLI) and the vertical representation is modified as (001101, HIL).



(a) Find all possible locations. (b) Compute cost. (c) Choose Optimal locations. (d) Repeat (a)-(c), until the Insertion candidate list null.

Fig. 6. Using new *insert* operation to insert all candidate modules.

The new *Insert* operation could be divided into three parts: (1) find all possible locations; (2) compute cost; (3) choose optimal location. If one module is inserted in a floorplan, it has many locations could be chosen and the first step is to discover these candidate locations in one LD-packing floorplan. The possible insertion location is all down- left corner of the floorplan result. As shown in Fig. 6, all possible insertion location is set at the down-left corner. Every possible insertion location has the different cost and step 2 is to compute the cost for each candidate location. The cost function can be represented as follows:

$$C_{new} = D_1(A_{new} - A_{original}) + D_2(I_a + I_b + I_c - I_{th}) \quad (6)$$

where C_a denote the cost when the module A is inserted in this location, D_1 a 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 *Ith* denotes the threshold value for local current consumption. D_2 is to set a large value for penalizing high local current consumption. Every candidate location must be computed two times because costs are different when the insert module is directly inserted or be rotated. Note that Eq. (6) consider the area and the power consumption only. When other objectives (ex: wire length) must be considered in the floorplanning approach, this equation

could be extended. Finally, the module is inserted in the minimal cost location. We use a simple example to explain our *Insert* operation. In Fig. 6 (a), module I, H and L compose a result of the floorplan and module J and K are the insert candidate of modules. Four red triangles denote the candidate location in the floorplan. In (b), we compute the cost after module J is inserted in the all 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 shown in (c). Because the candidate list is not null, the *Insert* operation must be repeatedly used, as shown in (d).

Noise Aware 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

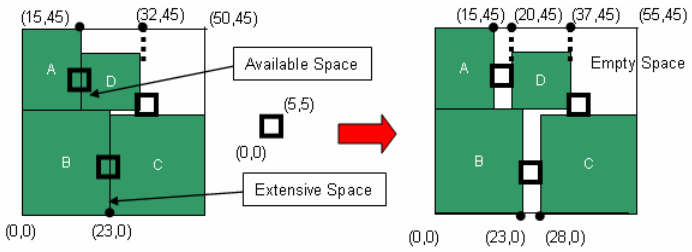
```
{   Set Temperture and Final_Temperture;
    While (Temperture > Final_Temperture)
      {  Random choose one block, Bx, from F;
         Using new Delete method to delete Bx;
         All deletion blocks are added into a candidate list;
         While (the candidate list is not NULL)
                {  Random choose one block, By, from the list;
                   Using Eq. (6) and new Insert method to insert By;
                   Delete By information in the list;   }
                ΔC = Cost(New Floorplan) – Cost(Floorplan);
                If (ΔC < 0) Floorplan = New Floorplan;
                Else
                {   If (Random(0,1) > e^(ΔC/Temperture))
                        Floorplan = New Floorplan;   }
                    Cooling(Temperture)   } }
```

Fig. 7. Power supply noise aware floorplan algorithm.

4.2 Identification of Space Priority for Decap Insertion

The space in a floorplan could be divided into three types: the *extensive space*, the *empty space*, and the *available space*. In the floorplan example shown in Fig. 8, the longest path are B $\rightarrow$ C and B $\rightarrow$ A. Module A, B and C are called the longest path modules. The overlapped channel between two consecutive longest path modules is defined as the *extensive space*. If one capacitance is inserted in the *extensive space*, the insertion operation will increase floorplan area. In Fig. 8, the hollow square denotes one capacitance block. The original floorplan area in Fig. 8 (a) is $2250\mu m^2$ and the floorplan area in Fig. 8 (b) is $2475\mu m^2$ after the decap is inserted into the *extensive space*.

A channel space with overlapped edges of the empty room is called the *empty space*. If one capacitance is inserted in this space, all the blocks will be located on the original positions and the floorplan area will not be increased. The space in a floorplan is called the *available space* if it is not *empty space* or *extensive space*. If a capacitance is inserted in this space, the area of the floorplan would not be extended, but the location of some



(a) (b)

Fig. 8. The increased area due to inserted decap.

blocks could be changed. For example, if one capacitance is inserted between the adjacent blocks A and D in Fig. 8 (a), the block D would be horizontally moved to its right as shown in Fig. 8 (b). Initially, the block A is the right block of the block D. After inserting decap, the right block of the block D is a decap block but the area of the floorplan is not extended.

In summary, if a decap is going to be inserted in a given floorplan, we will search for its insertion location according to the following order to minimize the total area overhead incurred by the decap insertion.

Empty Space > *Available Space* > *Extensive Space*

4.3 Decap Insertion for Power Integrity

After obtaining one resultant floorplan from the first step, we will calculate the required decap size for insertion. In order not to increase the floorplan area excessively, we insert the decap with four smaller decaps instead of inserting one big decap in the extensive space because smaller capacitances are more possible to be inserted to the empty space directly. Given a floorplan, assume that one block needs decap to improve the supply noise. First, we use Eqs. (2-5) to compute the optimal decap size when the location of the decap and the VDD source of the module is separated. Second, the decap is separated into four smaller decap by the Manhattan distance from the VDD source to the power bump. For each smaller decap, the feasible insertion region is from a power bump to the VDD source. By this way, those smaller capacitances could be more possible to be inserted into the trivial empty space and the charge time of the capacitance could be substantially decreased.

If the decap location and the power bump location are not close enough, the decap must add charge to compensate the power consumption of the wire. The compensation methodology is shown as follows:

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

where l is the distance from the decap location to the power bump and c is the wire capacitance per unit length and V is the supply voltage. Finally, the minimum cost locations can be obtained to insert the decap. We called this methodology NDP_MAI, whose pseudo code is described in Fig. 9.

NDP_MAI Algorithm

```
{  While (The power supply noise for block, B_k, not estimated)
   {  If (I^k_max > I^k_gen)
      {  Using Eqs. (2-5) to compute decap DBS_k;
         The original decap is partitioned into four smaller decap based on the distance from
         VDD pins to the connection point of module k;
         For each smaller decap
         {  Find the feasible region of the decap;
            Fix all the space in the feasible region;
            For each space in the feasible region
                Compute area cost when decap insert in this space;  }
            Select the minimum cost space to insert decap;  } } }
```

Fig. 9. Noise-driven decap planning with minimal area insertion algorithm.

5. EXPERIMENTAL RESULTS

The proposed power supply noise aware floorplanning methodology and NDP_MAI algorithm have been implemented using C++ language and executed on an AMD 3200 machine with 1G memory. The set of useful parameters of the formulation in the power delivery model is based on the 0.25 μm technology. Five MCNC benchmark circuits are used to test the performance of proposed methodology. The power supply voltage is 2.5V and the distance between two continuous VDD is 1000Ω/μm and the power supply mesh is 333.3Ω/μm. The I_{gen}^{k} for the module k is $A_k \times D_c$, where A_k is area of module k, D_c is the worst case current density. The I_{gen}^{k} is assigned as a random value to be $1.05I_{gen}^{k}$ - $2I_{gen}^{k}$. Since the MCNC benchmark does not include noise constraint, the noise constraint is set to be 0.25V. In our experiments, the operation time t_{w0} and t_{w1} of the switching current waveform are set to be 0.3ns and 0.8ns.

To compare our method with [2], the computed decap budgets are listed in Table 1. In [16], the experimental results only show the floorplan area and runtime. So we cannot compare the decap budget with 1. In Table 1, it is clear that our computation methodology obtain more suitable decap budgets compared to the results reported in [2]. The experimental results show that our decap budget obtained average improvement of 52.6%. In [1], the authors do not provide the peak noise information in the paper. We only compare the peak value with [2] in Table 1. As shown in the experiment results, all cases can meet the given constraint.

Table 1. The peak noise for all modules in our approach and [2]. This shows our estimation needs less decap to suppress noise.

Circuit	Decap Budget (nF)	Decap Budget [2] (nF)	Decrease Ratio	Our Peak Noise (V)	[2]Peak Noise (V)
Apte	7.01	13.46	47%	0.25	0.24
Hp	1.65	2.75	40%	0.24	0.23
Xerox	3.09	5.71	46%	0.25	0.21
Ami33	0.08	0.27	70%	0.23	0.19
Ami49	3.61	9.08	60%	0.24	0.20

Table 2. Experimental result for MCNC benchmark circuits. This shows that our approach inserts less required decap while still meeting design constraints.

Circuit	Modules	Area (μm^2) (Floorplan)	Area (μm^2) (Insert Decap)	Increased Area (μm^2)	Increased Area [2] (μm^2)	Increased Area 1 (μm^2)
Apte	9	47761324	47780360	19036	469916	354000
Hp	11	9940140	10097780	157640	317503	67000
Xerox	10	20630210	20705216	75006	269374	144000
Ami33	33	1241440	1245266	3824	390	11000
Ami49	49	37504880	37659870	154990	218000	217000

Table 2 shows the comparison of the floorplan areas between the other methods and our proposed NDP_MAI algorithm. In Table 2, the experimental results of each benchmark circuit includes the area after floorplanning (Area(Floorplan)), the area after inserting the decap (Area(Insert Decap)) and the increased area after decap insertion (Increased Area). Compared to the numbers reported in previous papers, our proposed NDP_MAI algorithm has less area overhead to insert the required decap in most cases and the noise constraint could be conformed in all MCNC benchmark at the same time. The aspect ratio of the floorplan is the main reason why the result of the ami33 benchmark is worse than [2].

6. CONCLUSION

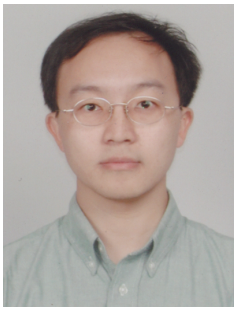
The techniques of floorplanning and decap insertion could be used to reduce the power supply noise in early design stage. Based on our framework, the suitable decap size is found. Furthermore, based on the analysis of the floorplan and space priority, the NDP MAI algorithm is proposed to insert decap budget. Experimental results show that the proposed method can improve the increase ratio of the floorplan area when inserting decap.

REFERENCES

1. J. T. Yan, K. P. Lin, and Y. H. Chen, "Decoupling capacitance allocation in noise-aware floorplanning based on DBL representation," in *Proceedings of International Symposium on Circuits and Systems*, 2005, pp. 23-26.
2. S. Zhao, K. Roy, and C. K. Koh, "Decoupling capacitance allocation and its application to power-supply noise-aware floorplan," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 21, 2002, pp. 81-92.
3. J. M. Lin and Y. W. Chang, "TCG: a transitive closure graph-based representation for general floorplans," in *Proceedings of IEEE/ACM Design Automation Conference*, 2001, pp. 764-769.
4. T. Mitsuhashi and E. S. Kuh, "Power and ground network topology optimization for cell based VLSIs," in *Proceedings of IEEE/ACM Design Automation Conference*, 1992, pp. 524-529.
5. X. D. S. Tan and C. J. Richard, "Fast power/ground network optimization based on equivalent circuit modeling," in *Proceedings of IEEE/ACM Design Automation*

Conference, 2001, pp. 500-554.
6. H. Su, K. Gala, and S. Sapatnekar, "Fast analysis and optimization of power/ground networks," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, 2000, pp. 477-480.
7. J. Singh and S. S. Sapatnekar, "Congestion-aware topology optimization of structured power/ground networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 24, 2005, pp. 683-695.
8. J. Fu, Z. Luo, X. Hong, Y. Cai, S. D. Tan, and Z. Pan, "VLSI on-chip power/ground network optimization considering decap leakage currents," in *Proceedings of IEEE Asia and South Pacific Design Automation Conference*, 2005, pp. 735-738.
9. K. H. Erhard, F. M. Johannes, and R. Dachauer, "Automatic sizing of power/ground networks in VLSI," in *Proceedings of IEEE/ACM Design Automation Conference*, 1992, pp. 362-367.
10. J. T. Yan, C. H. Lu, and C. W. Wu, "Simultaneous wiring and buffer block planning with optimal wire-sizing for interconnect-driven floorplanning," in *Proceedings of IEEE Midwest Symposium on Circuits and Systems*, 2004, pp. 25-28.
11. C. Y. Yeh and M. S. Malgorzata, "Timing-aware power noise reduction in layout," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, 2005, pp. 627-634.
12. H. M. Chen, L. D. Hung, I M. Liu, and D. F. Wong, "Simultaneous power supply planning and noise avoidance in floorplan design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2005, pp. 578-587.
13. Y. C. Chang, Y. W. Chang, G. M. Wu, and S. W. Wu, "B*-trees: a new representation for nonslicing floorplans," in *Proceedings of IEEE/ACM Design Automation Conference*, 2000, pp. 58-463.
14. P. N. Guo, C. K. Cheng, and T. Yoshimura, "An O-tree representation of nonslicing floorplan and its application," *IEEE/ACM Design Automation Conference*, 1999, pp. 268-273.
15. X. Hong, G. Huang, Y. Cai, J. Gu, S. Dong, and C. K. Cheng, "Corner block list: an effective and efficient topological representation of nonslicing floorplan," in *Proceedings of IEEE/ACM Design Automation Conference*, 2001, pp. 764-769.
16. H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, "Rectangle packing based module placement," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, 1995, pp. 472-479.
17. S. J. Wen and J. T. Yan, "Double-bound list: a new placement representation with application to simulated-annealing-based floorplan," 2002.
18. C. H. Lu, H. M. Chen, and C. N. J. Liu, "On increasing signal integrity with minimal decap insertion in area-array SoC floorplan design," in *Proceedings of IEEE Asia and South Pacific Design Automation Conference*, 2007, pp. 25-28.

Chao-Hung Lu (呂昭宏) received the B.S. degree in Minghsin University of Science and Technology , Taiwan, R.O.C. and the M.S. degree in Chung Hua Uiversity , Taiwan, R.O.C. He is currently a candidate for doctor's degree in the Department of Electrical Engineering at National Central University. His research interests include physical design.



Hung-Ming Chen (陳宏明) received the B.S. degree in Computer Science and Information Engineering from the National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1993 and the M.S. and the Ph.D. degrees in Computer Sciences from the University of Texas, Austin, in 1998 and 2003, respectively. He is currently an Assistant Professor of Electronics Engineering with the National Chiao Tung University. His research interests include very large scale integration of computer-aided design, especially on physical design and system-on-a-chip methodology, design and analysis of algorithms, and conbinatorial optimizations.

Chien-Nan Jimmy Liu (劉建男) received The B.S. and Ph.D. degrees in Electronics Engineering from National Chiao Tung University, Taiwan, R.O.C. He is currently an assistant professor in the Department of Electrical Engineering at National Central University. His research interests include functional verification for HDL designs, high-level power modeling, and analog behavioral models for system verification. He is a member of Phi Tau Phi.