

Power-Gating Aware Floorplanning

Hailin Jiang
ECE Department, UCSB
hailin@ece.ucsb.edu

Malgorzata Marek-Sadowska
ECE Department, UCSB
mms@ece.ucsb.edu

Abstract

Power-gating is a technique for efficiently reducing leakage power by shutting off the idle blocks. However, the presence of power-gating may also introduce negative effects, which are not considered in the earlier design stages. Ignoring those effects may result in suboptimal designs and potentially even nullify the intended power savings. In this paper, we propose a novel measure to efficiently capture the power-gating effects. We apply this measure in a floorplanner for power-gated chips. Experimental results show that the power-gating aware floorplanner can achieve 50% decap saving compared to a floorplanner unaware of power gating. Leakage power can be saved by inserting less decap, especially when thin-oxide decap are used due to the area constraint. Our approach can reduce leakage power consumed by decap from 36mW to 9mW when area overhead is limited to about 19% of the total chip area.

1. Introduction

With continuous shrinking of the minimal feature size accompanied by lowering of supply and threshold voltages, leakage power is emerging as a major challenge for present and future CMOS designs. Power-gating is an efficient technique for reducing leakage power by shutting off the idle blocks. Implementation of power-gating requires a multi-threshold CMOS process. Logic blocks are implemented using low- V_t , high-performance transistors whereas high- V_t transistors (so called sleep transistors) connect the gated blocks to the power supply [5].

Power-gating is not a cost-free technique. In [3], the authors discuss and quantify the negative effects of power gating which include noise, performance penalty, area, and power overhead. Recently, physical design-level techniques have been developed to eliminate some of these negative effects. In [1], to reduce the magnitude of the voltage fluctuations, the authors propose to turn on the sleep transistors in a non-uniform, step-wise manner. In [4], the authors present an efficient algorithm for placing power gating switches in the voltage islands. Their objective is to meet the electrical requirements of a design while guaranteeing the design specifications of the switches and minimizing the area overhead.

However, all these works focus on designing the components of power-gating circuitry, such as sleep transistors. In other words, there is no mention whether the earlier design stages require any special considerations related to power gating. Floorplanning is an early stage of physical design deciding relative placement and physical

dimensions of macros. Power-gating also works on the macro level. The candidate gated blocks are decided before floorpanning based on the functionality and performance of the circuits. However, there are several factors determined at the floorplanning (or placement) stage which affect the area and power of a gated circuit. They are the locations and neighbors of the gated blocks, the distances between gated- and non-gated blocks, placement of gated- and non-gated regions. Without considering the power-gating effects in floorplan, the negative aspects of power gating may overwhelm the potential gain and may make the technique not worth the effort. In this work, we present an efficient measure which captures power-gating effects in floorplan stage. We develop a power-gating driven floorplanner (PGFP) for designing power-gated chips. To the best of our knowledge, our work is the first addressing this problem.

The rest of this paper is organized as follows. In Section 2, we analyze the problems introduced by power-gating in the context of floorplanning and present a new metric to measure these effects. In Section 3, we describe our power-gating driven floorplanner. In Section 4, we show the experimental results. Section 5 concludes the paper.

2. Analysis of the problem

2.1. Power-gating

Traditionally, the on-chip power distribution network is constructed hierarchically. Power is delivered from the top-level metal layer, which is connected to the package, down through the internal layers, and finally to the active devices. The entire chip is fully powered at all times. In a power-gated system, the chip is divided into several domains and local power meshes are constructed for each of them. The local mesh for gated circuit is connected to the global mesh via sleep transistors. The sleep transistors are turned off to reduce the leakage power in a standby mode, and turned on in an active mode for normal operation.

When a block is active, the sleep transistor is turned on. Its size must be large enough to guarantee that the current drawn by the block does not cause excessive voltage drop. However, when large size sleep transistors are turned on and off, considerable power supply currents are drawn in a short period of time, which may result in large voltage fluctuations on the power grid. This *power-gating noise (PGN)* affects reliability of system-on-chips (SoCs), especially those manufactured in advanced CMOS tech-

nologies with narrow noise margins. We need to add extra decap to reduce the *PGN* and take into account additional leakage currents associated with the intentionally introduced capacitors.

In idle state, gated blocks are disconnected from the power supply, which prevents them from acting as decaps for the active circuits. Thus the non-gated blocks adjacent to the gated circuitry may need extra decap to guarantee that power grid noise remains within tolerable margins. The gated blocks not only stop providing decap to other blocks, but also act as obstacles for decap insertion. To eliminate the *PGN* in a gated chip, extra decap must be inserted which will enlarge the chip size and consume more leakage power.

The purpose of introducing power-gating is to save power. If the cost of power overhead caused by power gating is greater than the savings it provides, power-gating is useless. So our task is to reduce those negative effects as much as possible during each design stage to maximize the final power saving gained from the power-gating technique.

2.2. Floorplan effects on power-gating

To analyze the effect of a floorplan on power saving in a power-gated chip we use the example shown in Fig.1. The shaded area represents a gated-region with two blocks. We assume that each block has one sleep transistor. The block *B* has a higher current density (dark grey) than *A* (light grey). With the same distance of a gated-block to the gated-region boundary, upon wake-up, the higher current block (which has a larger sleep transistor) will cause a larger grid-current fluctuation for its neighbor non-gated circuits. With no other constraints, the floorplan in Fig.1(b) is better than the one in Fig.1(a) because it requires less extra decap. The locations of non-gated blocks also affect the power-gating result. With the same distance of non-gated blocks to the gated region boundary, during the wake-up period a higher current density block will be affected more. So we prefer a floorplan with block current densities decreasing towards the gated region boundary, from both gated and non-gated sides.

The shape of a gated region also affects the efficiency of power-gating. Suppose we have two gated regions of the same area but of different shapes as shown in Fig.1(c) and Fig.1(d). The floorplan in Fig.1(d) has a longer boundary between the gated and non-gated regions, which implies more boundary effects during the wake-up time. So without other constraints, we prefer a chip with gated-circuits placed close together.

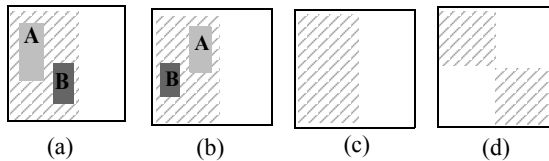


Figure 1. Floorplan effects on power-gating

2.3. Estimating decap in floorplan stage

Based on the observations discussed in Section 2.2, we propose a metric to predict the extra decap required by power-gating. For a given floorplan, we divide the chip into $M \times N$ bins. The bins occupied by gated blocks are labeled *gb*, and the bins occupied by non-gated blocks are labeled *ngb*. cd_i is the current density of a bin *i*, which is calculated as:

$$cd_i = \frac{\sum_{j=1}^{K_i} cd_{ij} \cdot a_{ij}}{A_i}$$

where cd_{ij} denotes current density of the *j*th block in the *i*th bin, and a_{ij} is the overlap area of the *j*th block and the *i*th bin. A_i is the bin area. K_i is the number of blocks in the *i*th bin. We also record the distances between the center of a gated (non-gated) bin and the nearest bins facing it on the north, south, west or east occupied by non-gated (gated) blocks. For example, in Fig.2, bin *i* is occupied by a gated block *A*. d_e and d_s are the distances between the center of *i* and the closet *ngb* on the east and south, respectively. We use the following formula to measure the extra decap required by introducing power-gating:

$$decapCost = \frac{gbinCost}{|totalBins|} + \frac{ngbinCost}{|ngBins|} \quad (EQ 1)$$

$$gbinCost = \sum_{i \in gbins} \sum_{k \in \{e, s, w, n\}} \sqrt{\frac{cd_i}{d_{ik}}}$$

$$ngbinCost = \sum_{i \in ngbins} \sum_{k \in \{e, s, w, n\}} \sqrt{\frac{cd_i}{d_{ik}}}$$

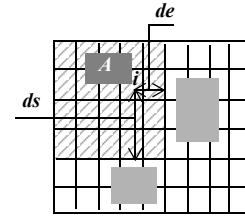


Figure 2. Measurement of extra decap

EQ1 is constructed such that if the higher current density blocks are placed closer to the boundary, the cost of extra power-gating decap is higher. The gated bin cost is divided by the total number of bins, and the cost of non-gated bins is divided by the number of non-gated bins, which implies that a smaller area of non-gated region requires larger decap area.

We test our metric on 100 power gated chips. The test chips are built in such a way that for a given floorplan, we choose randomly the locations and areas of gated circuits. The maximal number of gated regions is two. For each chip, we power gate the blocks in the gated regions and find the extra decap required to nullify the noise caused by power-gating. Then we compare the actual decap value with our metric. Fig.3 shows good correlation between the real decap insertion and our metric. We conclude that our metric can accurately capture the decap cost of

power-gating and can be used to guide a floorplanner for power-gated chips.

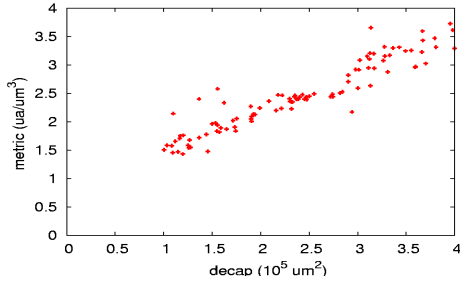


Figure 3. Correlation between our metric and real decap insertion

3. Power-gating driven floorplan

3.1. Sequence-pair

We use the sequence-pair representation of the floorplan topology. A sequence-pair (SP) of a module set M is a pair of sequences of the names of the modules in M . For example, $SP = (abc, bca)$ is a sequence-pair of $M = \{a, b, c\}$. An SP of M specifies the placement topology by the following rules:

- Horizontal constraint: $SP = (...a...b..., ...a...b...)$, module a is on the left side of module b ;
- Vertical constraint: $SP = (...a...b..., ...b...a...)$, module a is above module b .

The gated blocks should be placed together. Those blocks are a subset of M and can be represented as a superinduce. We need two types of sequence-pairs to represent a power-gated floorplan: SupSP for all the non-gated modules and gated supermodules, and SubSP for the subset of modules in each supermodule. Fig.4 shows an example. If blocks 3, 4, 5, 6 are gated blocks, we group them together as a supermodule a . $SupSP$ and $SubSP$ are used to represent the floorplan in Fig.4(b).

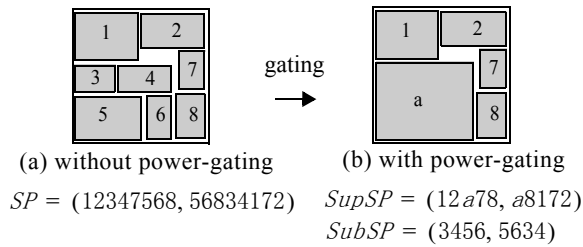


Figure 4. floorplan represented by sequence-pair

In this paper we discuss the case of one gated supermodule, but our method can be extended to the case of many supermodules.

3.2. Simulated annealing

The problem of power-gating aware floorplan can be formulated as follows. Given a set of modules (soft/hard)

with gating and connection information, find a floorplan which optimizes the following objective:

$$\text{MIN} \left[\sum_{i=1}^M \text{area}_i + \text{totalWirelength} + \alpha \cdot \text{totalDecap} \right] \quad (\text{EQ 2})$$

where M is the number of modules and α is a scaling factor to make the third and the first two terms of the same magnitude. totalDecap can be estimated by EQ1. totalWirelength is measured as a summation of bounding boxes of all connections.

We implement simulated annealing process to solve the floorplan problem. Given $SP = (S1, S2)$, we allow the following moves:

- *Move1*: switch two modules in $S1$, which implies that $a-b$ pair rotates by 90 degrees;
- *Move2*: switch two modules both in $S1$ or both in $S2$, which implies that $a-b$ pair rotates by 180 degrees;
- *Move3*: rotate an individual module.

Considering the supermodules included in a sequence-pair, we add the following rules:

- *Move1* or *Move2* can be applied on two non-gated modules;
- *Move1* or *Move2* can be applied on two supermodules;
- *Move1* or *Move2* can be applied on a non-gated module and a supermodule;
- *Move1* or *Move2* can be applied on two gated modules in the same supermodule;
- *Move2* can be applied on two gated modules in different supermodules with compatible gating schedules.

3.3. Sizing problem

Given a sequence-pair, we can determine the topology of a floorplan. Our objective is to find a geometric shape of each module minimizing the total area. The problem can be formulated as a geometric programming primal problem (PP):

$$\begin{aligned} \text{minimize:} \quad & W \cdot H \\ \text{subject to:} \quad & x_i + w_i \leq x_j \quad \forall e(i, j) \in G_H \\ & y_i + \frac{A_i}{w_i} \leq y_j \quad \forall e(i, j) \in G_V \\ & L_i \leq w_i \leq U_i \quad i = 1, 2, \dots, N \end{aligned}$$

where W and H are the chip width and height, respectively. A_i and w_i are the area and the width of a module i . x_i, y_i are the coordinates of a module i . L_i, U_i are the lower and upper bounds of the width w_i . $e(i, j)$ is an edge in the corresponding constraint graph (horizontal or vertical) for the sequence-pair.

We apply the Lagrangian Relaxation approach in [7] to solve this problem. We introduce Lagrange multipliers $\lambda_{i,j}$ and $\mu_{i,j}$ to incorporate constraints into the objective function. The following equations are repeatedly applied until w_i 's converge.

$$w_i^* = \text{sqr}t \left(A_i \cdot \left(\sum_{e(i,j) \in G_V} \mu_{i,j}^k \right) / \left(\sum_{e(i,j) \in G_H} \lambda_{i,j}^k \right) \right)$$

$$\begin{aligned}
w^k &= \min\{U_i, \max\{L_i, w^*\}\} \\
\lambda_{i,j}^{k+1} &= \max\{0, \lambda_{i,j}^k + \rho(x_i + w_i^k - x_j)\} \\
\mu_{i,j}^{k+1} &= \max\{0, \mu_{i,j}^k + \rho(y_i + A_i/w_i^k - y_j)\}
\end{aligned}$$

We assume that when power-gating is applied, the gated-blocks are individually power supplied. To maintain regularity of power mesh, the gated-blocks should be confined within a rectangular region. Fig.5 shows an example. For gated blocks 3,4,5,6 the floorplan of Fig.5(b) is better than the floorplan in Fig.5(a), even if the floorplan (a) is more compact.

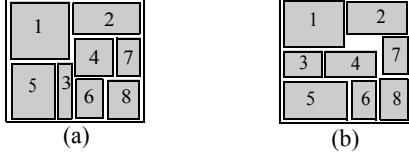


Figure 5. floorplan considering power-gating

3.4. Power-gating driven floorplanning algorithm

Our power-gating driven floorplan can be summarized as a two-step procedure embedded in simulated annealing process:

- Given a set of blocks of known areas, switching current profiles, aspect ratios, and gating status,
- Perform simulated annealing:
 - Random select a move; modules and supermodules are moved abiding to moving rules listed in Section 3.1;
 - **sizing-step1**: Calculate the shapes and relative locations of gated modules in a supermodule based on the approach discussed in Section 3.3;
 - **sizing-step2**: Calculate the shapes and locations of non-gated modules, and locations of supermodules on the entire chip; supermodules are treated as hard modules;
 - Estimate the cost of the floorplan considering the power-gating effects quantified by EQ2;
 - Decide the acceptance of a move;
- Until the temperature has cooled down.

Experimental results reported in [7] show that applying sizing procedure only once at the end of the annealing process yields good result while running 50x faster as compared to sizing-with-annealing process. We apply the same idea in our floorplanner. We perform the two-step-planning in each iteration treating the modules as hard objects whose locations are updated. At the end of the annealing process we size the blocks. This strategy improves the run time dramatically whereas it still maintains the floorplan quality.

3.5. Decap insertion

To verify our floorplanning approach considering power-gating, we develop a tool to optimally insert decap

to reduce power-gating noise. The model of a mesh is shown in Fig.6. The upper power mesh is modeled as a linear resistive network excited by constant voltage sources (we ignore the package inductance in this work). The lower power mesh is composed of a linear RC network excited by time-varying current sources which represent the switching currents drawn by the logic circuits. We use a simple triangular waveform to model the current profiles. Decaps come from on-chip non-switching capacitors C_{ckt} or oxide capacitors C_{ox} which are purposely inserted. The on-chip decaps are extracted from each logic block and are known before floorplanning and we assume that they are evenly distributed in each block. If C_{ckt} is not sufficient for noise immunity, more C_{ox} is added. In this work, we use both the thin oxide ($9 \text{ \AA}$) and the thick oxide ($13 \text{ \AA}$) C_{ox} modules. The thin-decap provides 44% more capacitance than the thick decap of the same area [2], but leaks more by an order of magnitude.

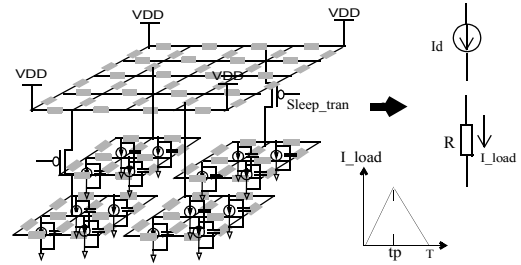


Figure 6. Power gating modeling

The model of sleep transistor depends on its mode of operation. When the sleep transistor is switching, it works in saturation mode and can be modeled as a current source with a triangle waveform. When a gated block is active, the sleep transistor which controls it works in the linear mode and can be modeled as a resistor. The optimization tool can be applied either on power-gated or non-gated chip, depending on if sleep transistor is modeled.

Given a floorplanned chip, we insert the extra decap such that the noise immunity is achieved and the total capacitance is minimal:

$$MIN \sum_{i=1}^N C_{ox}^i$$

where C_{ox}^i is the capacitance needed at location i , referred to as decap budget of decap i . N is the number of locations where decap can be inserted. We assume that decap can be inserted on the right and on the top of each block. So $N = 2M$, where M is the total number of blocks. We apply the algorithm described in [6] to solve this optimization problem.

Post-floorplan refinement might be implemented if the existing dead space is not enough for decap insertion. For example, in Fig.7(a), the required decap area is larger than the available dead space, so we extend the dead space in y direction (because δy is smaller than δx for the same area extension in this case) by moving macro 2. In this work, decap locations and decap budgets are deter-

mined during the optimization step and do not change in post-floorplan step.

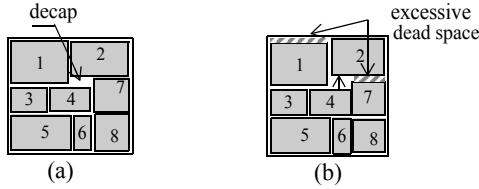


Figure 7. refinement in post-floorplan

4. Experiments

We used circuits from the MCNC and ISCAS benchmarks as basic macros and mapped them to the MCNC floorplan benchmarks. There are some floorplan benchmarks, such as *apte*, *xerox*, *hp*, which have too few macros (fewer than 11) to apply power-gating, so in this paper we show the results on *ami33* (33 macros) and *ami49* (49 macros). The switching current profiles are determined by simulating each circuit and approximating current waveforms by waveforms of triangular shape. The connections among the macros are taken from the *ami33* and *ami49* data sets.

We integrated C++ with TCL to implement our power-gating driven floorplanner. We ran all the experiments on a P4 2.4GHz PC with Linux. The test circuits were implemented in 0.13 μ m technology. The temperature was 25 $^{\circ}$ C.

We selected the gating status of macros in several ways to illustrate various situations. In this experiment, we applied five gating schedules: (g1) gated macros were selected randomly; (g2) high current density macros were gated; (g3) low current density macros were gated; (g4) tightly connected macros were gated, and (g5) weakly connected macros were gated. In each case, the gated macros occupied half of the chip area.

For each gating schedule, we implemented the following three floorplanning procedures:

- Regular floorplan (RFP): we applied the traditional non-slicing floorplan approach [7] on no-power-gated chips.
- Two-step-sizing floorplan (TSSFP): we implemented our two-step-sizing floorplan procedure to group the gated macros together, but we ignored the metric of gating effects in the cost function EQ2.
- Power-gating driven floorplan: (PGFP): we repeated the two-step-sizing procedure taking the gating effects into account.

Table 1 lists the comparison results in terms of dead space, run time, and decap budget, respectively. Dead space indicates the area not occupied by macros and we report it as percentage of the total area. Decap budget is the amount of capacitance required for achieving noise-immune chips. Not all dead space is used for decap insertion.

Comparing with no-power-gating floorplanning, the power-gating floorplanning (including both TSSFP and PGFP) costs more dead space. However, dead space can be utilized to insert decap, if necessary. Generally, power-gating driven floorplan saves more decap on gating high current density macros (g2) and gating tightly connected macros (g4), comparing with floorplanner unaware of power-gating. Decap increases by about 20% for power-gating without considering it at the floorplan stage. Experimental results show that our approach can decrease this number to 10%, which implies leakage power saving from inserting less decap.

Fig.8 shows the floorplan results with and without considering power-gating. Gated macros are contained inside the marked bounding box. We use colors from light to dark to show the distribution of current densities on the chip. The area colored light implies low current density and the area colored dark implies high current density. By applying power-gating driven floorplanning, more of the lower-current-density-macros are placed close to the boundary. The floorplanner tries to reduce as much as

Table 1. Comparison among floorplanners

		RFP			TSSFP			PGFP			decap budget	
		dead space (%)	run time (s)	decap budget (pf)	dead space (%)	run time (s)	decap budget (pf)	dead space (%)	run time (s)	decap budget (pf)	TSSFP vs. RFP	PGFP vs. RFP
ami33	g1	0.4	54	579	2.1	27	667	4.2	83	620	1.15	1.07
	g2				1.9	27	689	4.2	94	642	1.19	1.11
	g3				2.2	27	648	3.9	89	614	1.12	1.06
	g4				1.3	28	677	3.3	82	619	1.17	1.07
	g5				2.5	27	671	3.8	96	625	1.16	1.08
ami49	g1	2.4	415	722	13.3	354	888	18.1	501	801	1.23	1.11
	g2				13.5	363	902	17.3	519	808	1.25	1.12
	g3				14.2	369	859	19.7	526	794	1.19	1.10
	g4				8.7	349	895	16.2	498	794	1.24	1.10
	g5				14.8	377	873	19.2	512	823	1.21	1.14
aver											1.19	1.09

possible the length of the boundary between the gated and non-gated regions.

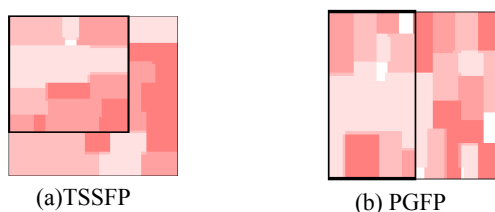


Figure 8. Current distribution of gating chip (ami33)

In the post-floorplan step, we allocate/extend the dead space for extra decap insertion. We configure our decap usage in such a way that we keep the capacitance at each location the same but we change the ratio between the thin- and thick-oxide decaps. Noise is not affected by different configurations because the capacitance does not change. Configuration of 100% thin-oxide decaps can achieve minimal area but consumes the most leakage power. Configuration of 100% thick-oxide can save the most power but also costs the most area. Given the dead space constraint, we compare our power-gating driven floorplanner with a floorplanner unaware of power-gating under different decap configurations. Table 2 reports the results. Column 1 shows the dead space constraint, Columns 2 and 4 show the leakage power consumption of decap, columns 3 and 5 show the decap configuration for the thin-oxide decap usage. We notice that if the dead space is too small, PGFP cannot provide feasible solution because it always consumes more dead space than TSSFP. However, if a design allows certain ratio of dead space, the power-gating driven floorplan may save much more power than a floorplan unaware of power-gating. For example, if the decap space constraint is 19% of the total area, the leakage power consumption is 36.1 mW by applying TSSFP whereas it is 9.2 mW by applying PGFP, which results in 26.9 mW saving of leakage power. We also notice that designs favor using thick-oxide decap as much as possible to reduce leakage power if area is not a concern.

Table 2. Leakage power with different decap configuration under dead space constraint(ami49)

dead space (%)	TSSFP		PGFP		power saving (mw)
	power (mw)	thin_oxide usage (%)	power (mw)	thin_oxide usage (%)	
13	93.2	100	N/A	N/A	N/A
17	60.0	60	39.4	40	21.6
18	49.3	50	24.3	20	25.0
19	36.1	30	9.2	0	26.9
22	10.2	0	N/A	N/A	N/A

5. Conclusions

In this work, we explore the power-gating technique and present an efficient measure for predicting power-

gating effects in floorplan stage. We develop a power-gating driven floorplanner (PGFP) to assist in designing of power-gated chips. Experimental results show that the power-gating aware floorplanner can achieve 50% decap saving compared to a floorplanner unaware of power gating. We also study how different configurations of decap types affect the floorplan in terms of area and power. Our power-gating driven floorplanner can significantly save power when area constraints are imposed by the design. We believe that in future designs with technology scaling down, power-gating will become a standard technique. Considering power-gating in early design stages will become necessary.

6. Acknowledgements

This work was supported in part by SRC grant # 1069 and in part by MICRO through IBM. We gratefully acknowledge equipment grant from Intel.

7. Reference

- [1] K.Choi, Y.Xu and T.Sakura, "An effective yet feasible power-gating scheme achieving two orders of magnitude lower standby leakage", *Symposium on VLSI Circuits*, pp.312-315, 2005.
- [2] H.H.Chen, J.S.Neely, and M.F.Wang, "On-chip decoupling capacitor optimization for noise and leakage reduction", *Proc. Symposium on Integrated Circuits and System Design*, pp.251-255, 2003
- [3] H.Jiang, M.Marek-Sadowska, S.R.Nassif, "Benefits and costs of power-gating technique", *Proc. ICCD*, pp.559-566, 2005
- [4] J.N.Kozhaya, L.A.Bakir, "An electrically robust method for placing power gating switches in voltage islands", *Proc. CICC*, pp321-324, 2004
- [5] S.Mutoh and et al, "A 1V power supply high-speed digital circuit technology with multi threshold voltage CMOS", *IEEE Journal of Solid-State Circuit*, pp. 847-854, 1995
- [6] H.Su, S. S. Sapatnekar, and S.R.Nassif, "Optimal decoupling capacitor sizing and placement for standard-cell layout designs", *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pp. 428-436, April, 2003
- [7] F.Y.Young, C.N.Chu and Y.C.Wong, "Handling soft modules in general nonslicing floorplan using Lagrangian Relaxation", *IEEE Trans on Computer-Aided Design of Integrated Circuits and Systems*, vol.20, pp.687-692, 2001
- [8] S.Zhao, K.Roy and C.K.Koh, "Decoupling capacitance allocation and its application to power-supply noise-aware floorplanning", *IEEE Trans on Computer-Aided Design of Integrated Circuits and Systems*, vol.21, pp.81-92, 2002