Floorplan Driven Leakage Power Aware IP-Based SoC Design Space Exploration *

Aseem Gupta†, Nikil Dutt†, Fadi Kurdahi†
†Center for Embedded Computer Systems
University of California, Irvine
Irvine, CA 92697 USA
{aseemg,dutt,kurdahi}@uci.edu

Kamal Khouri‡, Magdy Abadir‡
‡Design Technology Organization
Freescale Semiconductor Inc.
Austin, TX 78729 USA
{kamal.khouri,m.abadir}@freescale.com

ABSTRACT

Multi-million gate System-on-Chip (SoC) designs increasingly rely on Intellectual Property (IP) blocks. However, due to technology scaling the leakage power consumption of the IP blocks has risen thus leading to possible thermal runaway. In IP-based design there has been a disconnect between system level design and physical level steps such as floorplanning which can lead to failures in manufactured chips. This necessitates coupling between system level and physical level design steps. The leakage power of an IP-block increases with its temperature which is dependent on the SoC's floorplan due to thermal diffusion. We have observed that different floorplans of the same SoC can have up to 3X difference in leakage power. Hence the system designer needs to be aware of this design space between floorplans and leakage power. We propose a leakage aware exploration (LAX) framework which enables the system designer to create this design space early in the design cycle and provides an opportunity to make changes in the system design. We show the size of the design space generated by applying LAX on ten industrial SoC designs from Freescale Semiconductor Inc. and observe that the leakage power can vary by as much as 190% for 65% difference in the inactive area.

Categories and Subject Descriptors: Design
General terms: Algorithms
Keywords: Leakage power, Temperature, Floorplan.

1. INTRODUCTION

Multimillion gate System-on-Chip (SoC) designs increasingly rely on Intellectual Property (IP) or third party blocks due to the growing complexity of SoC functionality and reduced time-to-market. However, process scaling has enabled these IP blocks to offer much higher computational power and performance at the expense of rising concerns about power consumption. The power per unit area (power density) in a SoC has become the limiting factor in manufacturing chips running at higher clock frequencies. Recently, several semiconductor companies have announced that they will no longer increase clock frequencies on their main product lines due to concerns about high operating temperatures resulting from power consumption [1]. Because of scaling down of the supply voltage (V_{DD}) and the transistor threshold voltage (V_{th}), there is an increase of up to 5X in the leakage power dissipation per generation [2].

Unfortunately in current IP-based design there is a disconnect between system level design and physical design including floorplanning. Traditionally this disconnect has led to over-designing or failures by manufactured chips in meeting design constraints. As leakage and thermal effects begin to dominate design concerns, the disconnect between IP-based and physical design becomes very critical, motivating the need for our approach in coupling floorplanning with IP-based design.

The following three observations specifically motivate our research:

1. Different IP blocks have different functionalities due to which they have different dynamic power dissipation and hence produce varying local temperatures. The die temperature for an IP-block in a SoC is not confined to the block itself and affects the temperatures of all its neighboring blocks due to thermal diffusion. Thus the temperatures of the blocks cannot be determined without considering the floorplan of the SoC.
2. The leakage current of a transistor has been shown to have a super-linear dependency on temperature [15].
3. Different floorplans of the same SoC have different IP-block temperatures. Since there is a super-linear dependency of leakage power on temperature, the floorplan has a direct effect on the leakage power of the SoC. Different floorplans of the same SoC thus have different leakage power.

The above observations, when combined, lead us to believe that IP-based design must consider the effect of SoC's floorplan for system level leakage power estimation and management, and that designers need to explore the design space between floorplans and leakage power. Early detection of leakage power violations allow the designer to redo system-level design, thus saving design time and cost in an effort to achieve single pass designs.

The main contribution of this paper is a leakage aware exploration (LAX) framework for IP-based SoC design. LAX creates an exploration design space between leakage power and floorplans for the system designer, allowing early identification of leakage-controlled designs.

* This work has been supported by Freescale Semiconductor Inc.

CODES+ISSS'06, October 22–25, 2006, Seoul, Korea.

2. RELATED WORK

Related work has appeared in two areas: floorplan driven IP-based design and thermal-aware floorplanning.

Pasricha et al. [3] propose floorplan-aware automated synthesis of bus-based communication architectures. They have coupled exploration of communication design space to synthesize a communication architecture which meets the performance requirements with floorplanning and detect timing violations early in the design flow. Thepayasuwan et al. [4] and Bergamaschi et al. [5] use the floorplanner to generate an early core placement estimate during SoC design. Drinic et al. [6] use a floorplanner to determine feasibility of the synthesized design by comparing estimates of wire length with an upper bound on wire length. However, none of these efforts address the leakage power of the SoC.

In the category of floorplanning for thermal management, Sankaranarayanan et al. have proposed thermal-aware floorplanning at the micro-architectural level for processors [7], [8]. Their objective is to reduce peak processor temperature and complement dynamic thermal management schemes. A study on floorplanner's effectiveness in lowering the maximum processor temperatures was done by Han et al. [9] and by Ekpanyapong et al. [10] for 3-D chips. Floorplanning for leakage power management has been examined by Healy [11]. Hung et al. [12] have used floorplanning with genetic algorithms to reduce the peak temperatures and optimize area. Reduction of clock tree power using activity based register clustering and thermal aware placement has been proposed by Cheon et al. [13] and Obermeier et al. [14]. All of the above works have focused primarily on reducing peak temperatures. However, as we see in the next section, IP-based design aimed at peak temperature reduction may not have a direct impact on leakage power.

3. BACKGROUND

3.1 Peak Temperature & Leakage Power of Floorplans

Our experiments on an industrial design show that a floorplan with a lower peak temperature can exhibit higher leakage power as compared to a floorplan with a higher peak temperature. Fig. 1 shows the distribution of peak temperature and leakage power for various floorplans of the same industrial SoC. We have used a system level temperature and floorplan-aware leakage estimator STEFAL, which is summarized briefly in Section 4.3. Floorplans A through D are marked in increasing order of their peak temperatures. Floorplan A has the minimum peak temperature; however, its leakage power is not the least. The lowest leakage power was observed for Floorplan B, but its peak temperature is higher by about 23^oC than that of Floorplan A. Neither Floorplan C, with maximum leakage power, is the floorplan with the highest peak temperature, nor does Floorplan D, with maximum peak temperature, have the highest leakage power. This example experimentally establishes the following facts:

(i) Different floorplans of a SoC have different leakage power dissipation, our motivation for the LAX framework.
(ii) There is no definite correlation between the leakage power and the peak temperature of a floorplan.
(iii) There exists an exploration space between leakage power and floorplans for SoCs.

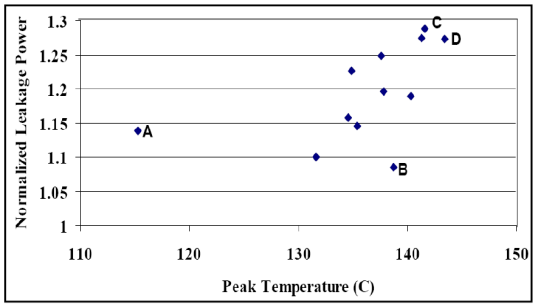

Figure 1: Peak Temperature and Leakage Power for Various Floorplans of a SoC

3.2 Thermal Characteristics of a Chip

The power consumed during transistor switching and due to leakage is dissipated as heat. The principal job of the package and cooling mechanisms is to facilitate the transfer of this heat from the die to the outside environment. A good rule-of-thumb is that the regions of the chip with high power densities usually have higher temperatures as well. But this may not always be true. The heat of a block is not confined to itself and tends to move from a high temperature region to a low temperature region, primarily by the mechanism of conduction. The temperature of a particular block in the chip depends on the power densities of the adjacent blocks as well. Han et al. [9] have showed that a block whose power density is 5X the power density of another block in the Alpha core, has its temperature lower than that of the other block by 12^oC. Hence the effects of thermal diffusion cannot be ignored and the floorplan becomes a key component while computing block temperatures from their known power densities.

4. DESIGN SPACE EXPLORATION FLOW

4.1 System Design Space Exploration

The proposed IP-based system design flow is shown in Fig. 2. The flow is horizontally demarcated into system level design and physical level design. IP-based design starts from a system description provided to the designer. The first step is to do behavior partitioning of the system into functional blocks which can be mapped to either software or hardware. After the system has been partitioned, IP-blocks which provide the required functionality have to be selected from an IP-library. Once all the blocks are available the SoC design is mapped on to the blocks and binding is done. The communication architecture and protocol selection is the next step. It also involves synthesis of the complete SoC. The area and the transistor composition of all the blocks is now available. Traditionally the IP-netlist hence obtained is handed off for physical design. Physical design also involves many steps before manufacturing and one of them is floorplanning. The above described IP-based SoC design flow is a simplistic view of the procedure and hides many other intermediate steps.

There has been a disconnect between system level and physical level design. After handoff the system designer has no influence over the physical design steps. However, often because of this compartmentalization the manufactured chips fail to meet design requirements. As we have seen in Section 3, different floorplans of a SoC have different leakage power. Hence the designer needs to be aware of the leakage power during IP-based system design. The leakage

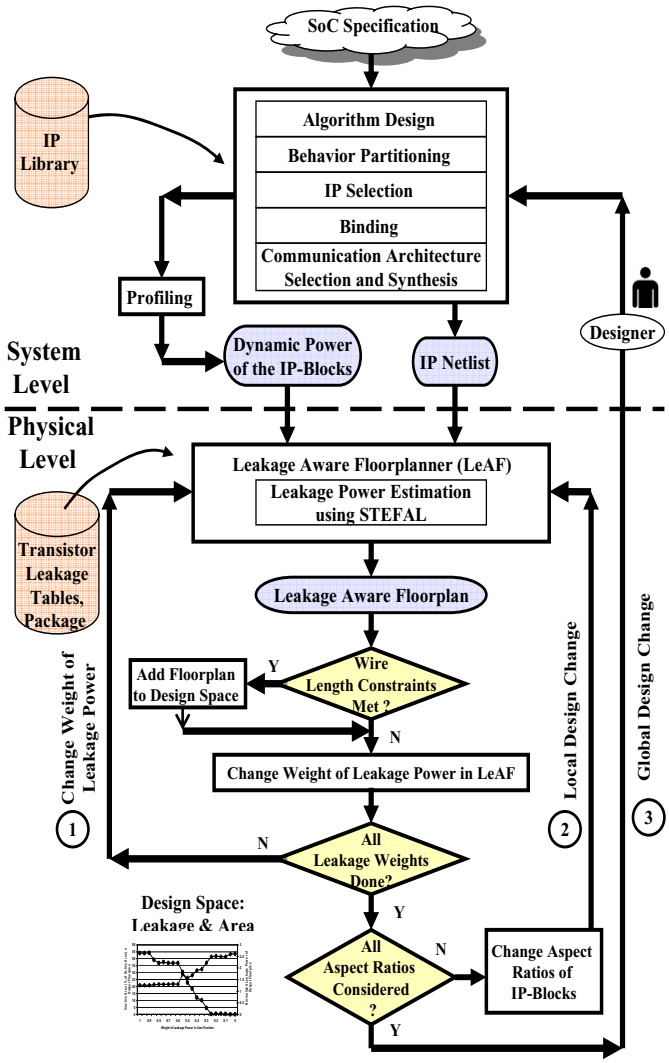

Figure 2: Leakage Aware Design Space Exploration Flow - LAX

power estimation should be temperature aware and floorplan aware. There is a design space for floorplans between area and leakage power which needs to be explored. In this work we propose leakage aware exploration (LAX) of design space during IP-based SoC design. LAX enables the designer to create the exploration design space between leakage power and inactive area (*sum of areas of dead space, interconnects, and connecting ports of blocks*) for different floorplans of the SoC. The designer now has an opportunity to select a floorplan which meets the area and leakage power constraints. The designer can also make system level decisions while being aware of floorplan dependent leakage power. Because of this coupling between system level and physical level design steps, the designer is aware of possible constraint violations early on and can make suitable changes in the design.

We now describe this coupling in more detail. As shown in Fig. 2, the IP netlist (output from system-level design) and dynamic power profiling information is input to leakage aware floorplanner (LEAF). LEAF is a slicing tree based simulated annealing floorplanner which optimizes floorplans for leakage power along with area and wire length. A brief summary of LEAF is provided in Section 4.2. The dynamic power is used to estimate temperature dependent leakage power within LEAF using another tool *STEFAL* described in Section 4.3. The leakage power is temperature dependent and its weight in the floorplanner's cost function is a variable $W_{Leakage}$. The leakage aware floorplan output by LEAF is added to the design space only if it meets the wire length constraint. If the floorplan does not meet the wire length constraint it is not added to the design space and is neglected. In either case more floorplans are generated by changing $W_{Leakage}$ in the cost function of LEAF. This iterative loop, labeled ① in Fig. 2, is used to populate the design space with many floorplans. For each floorplan that meets wire length constraint the design space has the leakage power and the inactive area.

After many iterations on loop ① the design space is populated for the designer. The designer can now choose a floorplan which meets the area and leakage power requirements of the design. However, the design space may be empty because none of the output floorplans met the wire length constraint or none of the floorplans in the design space may meet designer's area and leakage power requirements. In either case a new design space has to be generated for exploration by the system designer.

There are two ways to regenerate a new design space which are labeled ② and ③ in Fig. 2. A local design change such as changing the maximum aspect ratio (*ratio of length and width*) for some of the IP-blocks can be made in loop ②. IP-vendors usually provide a narrow range for the aspect ratios of the IP and for some IPs the aspect ratio may even be fixed. We show in the results section the impact of changing the maximum aspect ratio of some of the IP-blocks on the exploration space. Changing the maximum aspect ratio is a local design change because it does not change the IP-netlist. Because of changes in the maximum aspect ratio of some of the IP-blocks, the floorplanner outputs different floorplans and the design space is altered. If all the aspect ratios have been considered and the local design changes are unable to meet constraints then a global design change can also be made by the designer in loop ③. This can be achieved by changing the behavior partitioning of the system, selecting alternate IP-blocks, etc.

4.2 Leakage-Aware Floorplanner (LEAF)

Leakage aware floorplanner (LEAF) is a flavor of simulated annealing based slicing tree floorplanner [16]. The floorplanner starts with an initial Simulated Annealing Temperature (SAT) and a randomly generated initial floorplan solution. The goal of the floorplanner is to find a floorplan with the lowest value of a cost function among many candidate floorplans. Candidate floorplans are generated by randomly making any of these three moves on the Normalized Polish Expression of the current solution: Swap two adjacent operands; Complement a chain of non-zero length; and Swap two adjacent operand and operator.

The floorplanner evaluates the candidate floorplan using a cost function. The cost function of LEAF is a function of total area, wire length and leakage power:

$$\begin{aligned} Cost = & W_{Area} * S_{Area} * Total\_Area \\ & + W_{Length} * S_{Length} * Wire\_Length \\ & + W_{Leakage} * S_{Leakage} * Leakage\_Power \end{aligned} \quad (1)$$

where $W_{Area}, W_{Length}, W_{Leakage}$ are the respective weights and $S_{Area}, S_{Length}, S_{Leakage}$ are the respective scaling fac-

tors. The dimensions of area, wire length, and leakage are different and in order to make a fair comparison among them using weights, they need to be multiplied by respective scaling factors. The weights are relative:

$$W_{Area} + W_{Length} + W_{Leakage} = 1 \quad (2)$$

The leakage power is estimated using the candidate floorplan and the dynamic power profile of the IP-blocks by the *STEFAL* tool. The candidate floorplan is automatically accepted as the current floorplan solution if it has a lower cost function else the probability of acceptance depends on the SAT and the cost function. As SAT goes towards zero the smaller cost function solutions are increasingly chosen. The floorplanner accepts both smaller and higher cost function solutions which saves the method from becoming stuck at a local minima. For each value of SAT, the floorplanner generates candidate floorplans for MAX_STEPS number of times. The value of SAT is decreased by a step size which is a directly proportional function of SAT. When the value of SAT reaches a minimum the iterations are discontinued and the current floorplan solution is generated as output.

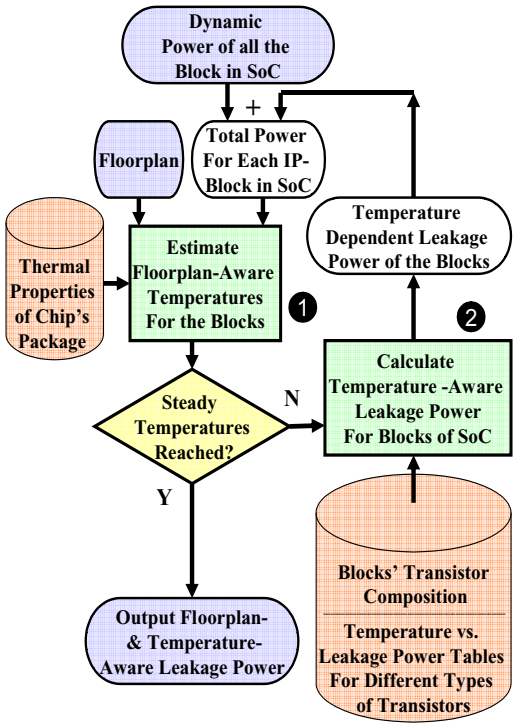

Figure 3: Floorplan-Aware Leakage Estimation

4.3 STEFAL

Fig. 3 shows the flow of the system level temperature and floorplan aware leakage (*STEFAL*) estimator tool used to calculate the leakage power for the cost function in Eqn. (1) for LEAF. STEFAL takes the inputs of the dynamic power profiles of all the IP-blocks and the floorplan for which the cost function has to be calculated. The procedure *Estimate Floorplan-Aware Temperatures for the Blocks*, labeled ❶, estimates the temperatures of the blocks from the input floorplan and total power (Dynamic + Leakage) for the blocks, along with the library of thermal properties of the chip's package using a freely available tool HotSpot [15]. The procedure *Calculate Temperature Aware Leakage Power for Blocks of the SoC*, labeled ❷, calculates temperature-aware leakage power using the block temperatures estimated by ❶, library of blocks' transistor composition, and the library of temperature vs. leakage power tables for different types of transistors. Transistors can be of many different types such as combinations of p-type or n-type and high V_{th} or low V_{th} and other manufacturing details. Each IP-block is composed of many different types of transistors. The transistor composition for the blocks contains details about the number of different types of transistors used within each block. Different types of transistors have different leakage power and exhibit different temperature sensitivity. The temperature vs. leakage power tables are pre-generated library tables at 65nm technology and have the leakage power values for each type of transistor from 27^oC to 150^oC in increments of 0.1^oC.

For each block, the leakage power calculated by Procedure ❷ is added to its dynamic power to get its total power which is then used to estimate the block temperatures using Procedure ❶. This is done iteratively because of the positive feedback between leakage and temperature [17]. If the maximum difference in the block temperatures of two consecutive iterations is less than a small value ϵ, the SoC has reached steady state temperature and the iteration is terminated. STEFAL then outputs the total leakage power of the SoC.

4.4 Assumptions

In this work we make several assumptions for LAX, LEAF, and STEFAL:

- We used industrial SPICE-like simulation models to generate the temperature vs. leakage power tables for 16 different types of transistors that were used in an industrial design. More types of transistors can also be added to the library.
- All the IP-blocks used in our designs were composed only of these known types of transistors.
- Currently we do not consider the interconnect power dissipated in the SoC and assume that the dynamic power remains constant with the floorplan.
- We assume an equal probability of a 0 or 1 at the input of a transistor.
- Currenly we do not consider the impact of physical-aware synthesis nor the impact of clock tree power for different floorplans.
- In case of some SoCs the power profiles of different benchmarks have to be averaged to get an average dynamic power profile. This was the approach followed by [7],[8],[9] to do floorplanning for peak temperature reduction.

5. RESULTS

We applied LAX to ten industrial designs from Freescale Semiconductor's PowerQUICC family of SoCs. Because of the proprietary nature of the data we have normalized the results. To create the design space LAX executed LEAF for different values for the weight of leakage power ($W_{Leakage}$) as shown in loop ① in Fig. 2. We varied $W_{Leakage}$ from 0 (floorplanner becomes leakage-unaware) to 1 (floorplanner optimizes only for leakage power) in steps of 0.1 thus outputting 11 floorplans. We collected data on the leakage power and the inactive area (as a % of active area) of the resultant floorplans. All the resultant floorplans for which we provide data here meet the wire length constraints of the respective designs. The design points for a higher value of $W_{Leakage}$

usually have a higher inactive area because from Eqns. (1), (2) the floorplanner optimizes less for area. Though different designs have different area constraints, floorplans with less than 30% inactive area are generally acceptable [18]. Conservatively, we assume that the floorplans with less than 20% inactive area will be feasible and give supplementary comments about the region of the design space with such floorplans. The relevance of this region of the design space with less than 20% inactive area rests with the designer.

5.1 Design-I

This design is for a SoC with 8 blocks. The leakage power constraint for this design is 1.6 (normalized) and the inactive area constraint is 25%. Only for this design we incremented $W_{Leakage}$ in steps of 0.05 thus outputting 21 floorplans. The exploration space available to the designer when the maximum aspect ratio of some of the IP-blocks is 3 is shown in Fig. 4. We observed that none of the floorplans met the leakage and area constraints of this design. The floorplan at $W_{Leakage} = 0.5$ meets leakage power constraint but not inactive area constraint. Thus a local design change as explained in loop ② of Fig. 2 should be made to the design. The maximum aspect ratio for some of the more flexible IP-blocks was increased to 6. The altered design space is shown in Fig. 5. We now find that the floorplan at $W_{Leakage} = 0.5$ meets both leakage power and inactive area constraints. The difference in the leakage power for leakage only-aware ($W_{Leakage} = 1$) and leakage-unaware ($W_{Leakage} = 0$) floorplans are 190% and 108% for maximum aspect ratios of 3 and 6 respectively. The differences in the leakage power of the smallest and the largest floorplans in the 20% inactive area region are 64.7% and 83.6% for maximum aspect ratios of 3 and 6 respectively. We have shown the region of the design space with less than 20% inactive area. Thus LAX did provide the designer with a vast design space. Our future work will examine ways to suggest these design changes to the designer in an intelligent manner.

5.2 Design-II

This design is the largest SoC made available to us with 49 blocks. For this SoC the outputs from $W_{Leakage} = 0$ to $W_{Leakage} = 0.4$ had an inactive area of less than 6%. The difference in the leakage power of the floorplans at the two values of $W_{Leakage}$ is 29.2%. Thus for this SoC, LAX provided the designer with multiple floorplans which had very small difference in their inactive area, but their leakage power differed by 29.2%. The floorplan for $W_{Leakage} = 1$ had an inactive area of 103.53% and its leakage power was 67.4% less than the floorplan for $W_{Leakage} = 0$. This design offered an interesting design space because of its size and nature. Out of the 49 IP-blocks in the SoC, many blocks were of small area which packed the space between larger IP-blocks. Thus the floorplans between $W_{Leakage} = 0$ to $W_{Leakage} = 0.4$ had a very small inactive space area of less than 6%.

5.3 Results on Designs-III to X

The results for exploring industrial Designs-III through X using LAX are summarized in Table 1. Columns 2 and 3 have the inactive area and the normalized leakage power of the design space at $W_{Leakage} = 0$ respectively while for $W_{Leakage} = 1$ they are shown in Columns 4 and 5 respectively. The difference in the leakage power between the two

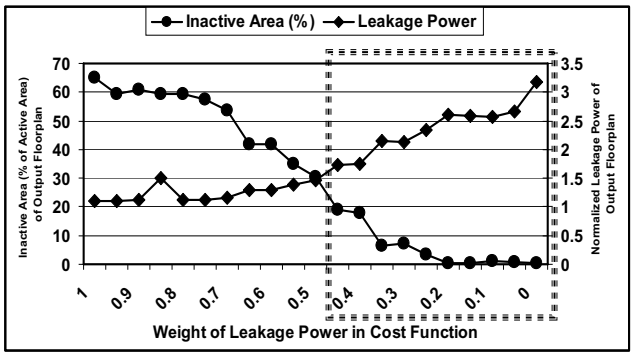

Figure 4: Results for Design-I, Maximum Aspect Ratio = 3

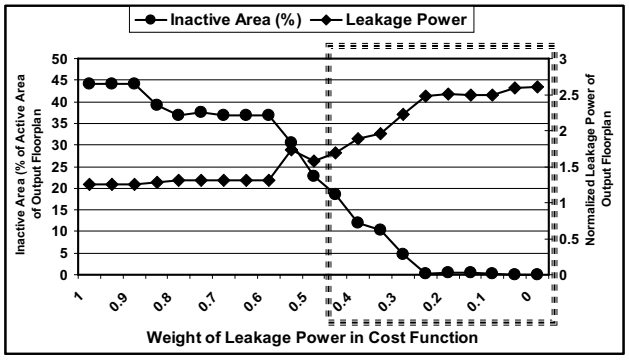

Figure 5: Results for Design-I, Maximum Aspect Ratio = 6

ends of the design space (Columns 3 and 5) is in Column 6. The values in Column 6 represent the size of the exploration space for leakage power. We observe that among these designs LAX created the largest leakage aware exploration space of 132.3% for Design-III. Out of the total 11 floorplans, the number of floorplans with less than 20% inactive area are in Column 7. For most of the designs, 9 out of 11 floorplans have less than 20% inactive area and Design-V has 10 such floorplans. This indicates that a majority of points in the design space did in fact have feasible inactive area. The maximum normalized leakage power among these floorplans is shown in Column 8. Column 9 shows for each design, the difference in the leakage power between leakage-unaware floorplan (Column 3) and the feasible floorplan with maximum leakage (Column 8). We observe a maximum difference of 35.69% in the leakage power for Designs-III. Design-IX showed interesting results because LAX output 3 floorplans with 0% inactive area for $W_{Leakage} = 0, 0.5, 0.7$. It is not necessary that the inactive area will always decrease along with $W_{Leakage}$ because LEAF is a simulated annealing based floorplanner which employs random search using a cost function. These results on industrial designs further support the rationale behind leakage power aware exploration of design space for IP-based design.

5.4 Discussion on Results

From the experiments on ten industrial SoC designs from Freescale Semiconductor we observed that: there is a significant variation in the leakage power in the design space and indeed, LAX created a significantly large exploration space for the system designer. This design space can be conveniently generated for IP-based SoCs during system level design because the system designers usually have access to

Table 1: Results for applying LAX to Designs-III through X

(1) Design (# of blocks)	(2) Inactive Area for $W_{Leak} = 0$	(3) Normalized Leakage for $W_{Leak} = 0$	(4) Inactive Area for $W_{Leak} = 1$	(5) Normalized Leakage for $W_{Leak} = 1$	(6) Diff. in leakage between (3)&(5)	(7) Number of Floorplans with $\leq 20\%$ Inactive Area	(8) Maximum leakage of floorplans in (7)	(9) Diff. in leakage between (3)&(8)
Design III(23)	0.98%	4.6	158.5%	1.98	132.3%	7	3.39	35.69%
Design IV(10)	2.23%	4.1	68.94%	2.61	56.81%	9	3.47	18.06%
Design V(6)	0.98%	3.07	34.84%	2.24	36.85%	10	2.48	23.74%
Design VI(12)	0.30%	4.74	78.53%	3.01	57.43%	9	4.62	2.68%
Design VII(17)	0.22%	10.76	93.98%	5.34	101.54%	8	8.78	22.63%
Design VIII(31)	1.08%	11.19	120.0%	5.42	106.14%	8	8.95	25.01%
Design IX(9)	0%	1.51	46.05%	1.0	51.12%	9	1.32	14.8%
Design X(18)	0.31%	3.69	67.38%	2.27	62.82%	9	3.33	11.0%

[a]Total number of floorplans for each design = 11

synthesized version of the IP-blocks. The coupling between system level and physical level design allows the system designer to identify leakage power violations early on and make suitable changes in the design. After a violation was identified in Design-I, the system designer altered the design space by making a post-netlist local design change and was able to meet design constraints.

We executed LAX using cygwin, which provides a linux-like environment for windows operating operating system, on a Intel Centrino 1.8 GHz processor with 512 MB RAM. The time to generate exploration space varies with the size of design because of the step of floorplanning. The exploration space for a design of average size of around 20 blocks can be generated using LAX in about 24 hours. Out of this time the temperature estimation and other steps take only a few minutes. The bulk of the time is spent in simulated annealing iterations of LEAF. We believe that the benefits of the exploration space outweigh the computational needs. Future work can examine using faster floorplanners to reduce the time taken by LAX.

6. CONCLUSION

In this paper we attempted to address the disconnect between system level and physical level design steps during IP-based design. We motivated the need to couple system level design with floorplanning in order to manage leakage power of the SoC. We proposed leakage aware exploration (LAX) which enables the system designer to create this design space early in the design cycle and provides an opportunity to make changes in the system design. We showed the size of the design space generated by applying LAX to ten industrial SoC designs and observed that the leakage power can vary by as much as 190% for 65% difference in the inactive area. We also observed the impact of making post-netlist design changes on the design space. In our future work we plan to consider the interconnect power and the clock tree power of the design, examine intelligent ways to suggest design changes to the system designer, and to use faster floorplanners. We also plan to reduce the time needed to run LAX by guiding LAX towards an acceptable solution instead of a comprehensive search.

7. REFERENCES

[1] Stephan Ohr, "Efforts heat up to remove processor hot spots," *EE Times*, February 2005.

[2] V. De et al., "Technology and design challenges for low power and high performance," *Proc. of Intnl. Symp. on Low Power Electronics and Design*, 1999.

[3] S. Pasricha et al.,"Floorplan-Aware Automated Synthesis of Bus-based Communication Architectures," *Design Automation Conference*,2005.

[4] N. Thepayasuwan et al.,"Layout Conscious Bus Architecture Synthesis for Deep Submicron Systems on Chip," *In Proc. of DATE*, 2004

[5] R. A. Bergamaschi et al., "SEAS: a system for early analysis of SoCs," *In Proc. of CODES-ISSS*, 2003.

[6] M. Drinic et al., "Latency-guided on-chip bus network design," *In Proc. of ICCAD*, 2000

[7] K. Sankaranarayanan et al., "A Case for Thermal Aware Floorplanning at the Micro-architectural Level," *Journal of Instruction Level Parallelism*, 2005.

[8] K. Sankaranarayanan et al., "Microarchitectural Floorplanning for Thermal Management: A Technical Report," *University of Virginia*, 2005.

[9] Y. Han et al.,"Temperature Aware Floorplanning," *Workshop on Temperature Aware Computer Systems*, June 2005.

[10] M. Ekpanyapong et al., "Thermal Aware 3-D Microarchitectural Floorplanning," *Technical Report, Georgia Tech*, December 2004.

[11] M. Healy., "Performance and Temperature Aware Floorplanning Optimization for 2D and 3D microarchitectures," *Ph.D. Dissertation*, May 2006.

[12] W.L. Hung et al.,"Thermal-Aware Floorplanning Using Genetic Algorithms," *Int. Symp. on Quality Electronic Design*, 2005.

[13] Y. Cheon et al., "Power-Aware Placement," *Design Automation Conference*, 2005.

[14] B. Obermeier et al., "Temperature Aware Global Placement," *Asia and South Pacific Design Automation Conference*, 2004.

[15] K. Skadron et al., "Control-theoretic Techniques and Thermal-RC Modeling for Accurate and Localized Dynamic Thermal Management," *Proc. of Intnl. Symposium on High-Performance Computer Architecture*, 2002.

[16] M. Sarrafzadeh and C. K. Wong, *An Introduction to VLSI Physical Design,*McGraw-Hill Higher Education, 1996.

[17] L. He et al.,"Considering the Interdependence of Temperature and Leakage Interdependence of Temperature and Leakage,"*Design Automation Conference*,2004.

[18] Personal communication with designers at Freescale Semiconductor Inc.