Impact of Guardband Reduction On Design Outcomes: A Quantitative Approach

Kwangok Jeong, *Student Member, IEEE*, Andrew B. Kahng, *Senior Member, IEEE*, and Kambiz Samadi, *Student Member, IEEE*

***Abstract*—The value of guardband reduction is a critical open issue for the semiconductor industry. For example, due to competitive pressure, foundries have started to incent the design of manufacturing-friendly ICs through reduced model guardbands when designers adopt layout restrictions. The industry also continuously weighs the economic viability of relaxing process variation limits in the technology roadmap (available: http://public.itrs.net). Our work gives the first-ever quantification of the impact of model guardband reduction on outcomes from the synthesis, place and route (SP&R) implementation flow. We assess the impact of model guardband reduction on various metrics of design cycle time and design quality, using open-source cores and production (specifically, ARM/TSMC) 90- and 65-nm libraries and technologies as well as an *industrial* embedded processor core implemented in 45 nm. Our experimental data clearly shows the potential design quality and turnaround time benefits of model guardband reduction. For example, in our open-source cores, on average we observe 13% standard-cell area reduction, 12% routed wirelength reduction, 13% dynamic power reduction and 19% leakage power reduction as the consequence of a 40% reduction in library model guardband; 40% is the amount of guardband reduction reported by IBM for a variation-aware timing methodology. For the embedded processor core we observe up to 8% standard-cell area reduction, 7% routed wirelength reduction, 5% dynamic power reduction, and 10% leakage power reduction at 30% guardband reduction. We also report a set of fine-grain SPICE simulations that accurately assesses the impact of process guardband reduction, as distinguished from overall guardband reductions, on yield. We observe up to 4% increase in number of good dies per wafer at 27% process guardband reduction (i.e., with fixed voltage and temperature). Our results suggest that there is justification for the design, EDA and process communities to enable guardband reduction as an economic incentive for manufacturing-friendly design practices.**

***Index Terms*—Design guardband, design of experiments, process variation, yield.**

I. Introduction

In sub-90-nm process technologies, there has been increased interest in design for manufacturability (DFM) techniques that address mounting variability and leakage power challenges. As we review below, several recent works attempt to "close the loop" from systematic or deterministic variability sources (litho, etch, CMP) back to design analysis (SPICE models of devices and gates, RC extraction of interconnects, etc.). However, DFM tools and methodologies that bring process awareness into design analysis and optimization will be of limited interest to design teams unless the signoff design attributes (quality-of-result, or QOR), and/or the design cycle (turnaround time, or TAT) actually improve. In particular, design teams require clear financial returns to go through the extra tool adoption, flow integration, and design effort that lead to more manufacturable tapeouts to the foundry. The challenge today is for the foundry and EDA sectors to collaboratively deliver opportunities for design-side customers to realize such financial benefits in return for deploying DFM approaches. To this end, quantified ROI (return on investment) analyses are required.

Another motivation for our work comes from the semiconductor technology roadmapping (ITRS) [2] community, which spans lithography, process integration, front-end process, interconnect, etc. technologies. In the ITRS effort, it has never been clear "how much variability can design tolerate?" For example, the 2005 edition of the ITRS increased the lithography critical dimension (CD) 3-sigma tolerance from its historical 10% value up to 12%. While this relaxation of the ITRS CD control requirement enables continuation of the foundry process roadmap, it was obtained without any rigorous analysis of net impact on design value extractable per wafer. Future balancing between process scaling and design technology "equivalent scaling" on the Moore's Law roadmap must be guided by more quantitative analyses.

Today, in the 65-nm and early 45-nm nodes, particularly for high-performance process flavors, silicon providers are likely to consider providing variant guardbands at the level of device (SPICE) model or interconnect RCX models, corresponding to different regimes of manufacturing-friendliness or "DFM score" in the tapeout. A first example might be the reduction of worstcase-bestcase (WC-BC) guardband for RC extraction, which is enabled by the deployment of new golden models for chemical-mechanical planarization (CMP), and which lead to new process-aware extraction and timing analysis (as well as process-driven dummy fill) flows. A second example might be the application of a different (narrower) SPICE model guardband for, e.g., a multifingered device that is laid out with optimal (restricted) pitch and poly dummy layout choices.

Manuscript received January 28, 2009; revised June 05, 2009; accepted June 16, 2009. First published September 15, 2009; current version published November 04, 2009. A preliminary version of this work appeared in the *Proceedings of ISQED*, 2008. Extensions beyond the previous version include discussion and experiments on impact of guardband reduction on memory (Section III-A), enhancement of the yield discussion (Section V-C), and addition of an industrial testcase, from Qualcomm, Inc., to the experimental results.

The authors are with the Electrical and Computer Engineering Department, University of California at San Diego, La Jolla, CA 92093 USA (e-mail: abk@cs.ucsd.edu; kambiz@vlsicad.ucsd.edu; kjeong@vlsicad.ucsd.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSM.2009.2031789

With respect to the preceding discussion and examples, significant overheads to the silicon provider are associated with this nascent paradigm shift in the foundry-designer business model. Among these overheads: commitment to additional model-to-silicon fidelity constraints, increased process technology characterization effort, opening up of another dimension of competition with other foundries, etc. Yet, the benefits to the foundry are clear: incentive for design customers and EDA partners to "do the right thing" for the manufacturing process, and the opportunity to offer differentiated value to customers. Clearly, a missing element for the concept of layout-specific design guardbanding to go forward is a *framework* to quantify the impact of guardband change on design QOR and TAT. Our present work seeks to fill this gap.

In this paper, we develop an experimental framework and then experimentally quantify the impact of model guardband reduction on outcomes of the synthesis, place and route (SP&R) implementation flow. We make the following contributions.

- We study small open-source standard-cell cores in 90- and 65-nm foundry technologies (ARM/TSMC) as well as an *industrial* embedded processor core implemented in 45 nm, and separately evaluate the impacts of guardband reductions in the FEOL (Liberty timing models) and in the BEOL (RCX in golden extraction such as with Star-RCXT).
- We assess impact of guardband reduction with respect to a number of metrics of design productivity (iterations, CPU times in synthesis, CTS and P&R phases, total design flow TAT, etc.), design closure (final timing fixes, etc.), and design quality (standard-cell area, routed wirelength, critical-path delay, dynamic and leakage power, etc.).
- We observe that the value of guardband reduction can be very significant. For example, we find that the 40% guardband reduction obtained by [10] with a "iso-dense" variational timing analysis methodology leads to typical reductions of 13% in standard-cell area, 12% in routed wirelength, 13% in dynamic power, 19% in leakage power, and 28% in SP&R turnaround time for open-source designs in both 90 and 65 nm. We also observe reductions of 8% in standard-cell area, 7% in routed wirelength, 5% in dynamic power, and 10% in leakage power for the embedded processor core in 45 nm at 30% guardband reduction.
- We decompose each separate impact of P, V, and T on delay. We observe that each axis of PVT has different delay impact. If any of P, V, and T are fixed for reasons such as test specifications (low V_{cc} margin) or customer requests, it will limit the guardband reduction.
- We quantify the impact of the guardband reduction on design yield. Our analysis shows up to 4% increase in the number of good dies per wafer with 27% guardband reduction. However, we notice a reduction in the number of good dies per wafer after 40% guardband reduction.

The remainder of this paper is organized as follows. Section II reviews several related aspects of the literature. Section III describes our scaling methodology for both FEOL and BEOL guardband reduction. Section IV describes the implementation flow, tools and testcases used in our experimental investigation. In Section V, we present experimental data that assesses impact of guardband reduction on a number of design-related metrics. Finally, Section VI gives conclusions.

II. Related Literature

We are not aware of any previous literature that quantifies impact of guardband reduction in a modern IC implementation flow, as we do. However, we note two related literatures that respectively address 1) taxonomies of variation sources and guardbanding in the modeling and analysis chain and 2) systematic process variation-aware design analyses.

1) Taxonomies of Variation Sources and Guardbanding: It is well-understood that variation can arise from environmental parameters (temperature, supply voltage, etc.), manufacturing processes that lead to device and interconnect changes, and reliability effects (hot-carrier degradation, NBTI, etc.). Scheffer [19], [20] gives a taxonomy of uncertainty and variation sources, with emphasis on the back end of the line (BEOL), i.e., the interconnect stack. This is in a similar spirit to the work of Nassif [17], which reviews sources and impacts of parameter variability across inter-die and intra-die sources. While such works as these taxonomize and quantify individual variation sources, they do not make connections back to quantified impacts within the chip implementation flow.

2) Systematic Process Variation-Aware Design Analyses: Prediction and compensation of systematic variations has traditionally been done by the manufacturing process, with only simple guardbanded abstractions (e.g., design rules) being passed on to the designers. However, the increasing magnitude and 2-D pattern dependence of these variations, their impact on design metrics, and the inability of manufacturing equipment and process techniques to fully mitigate them, are cause for serious concern in sub-100-nm technologies. If modeling and design guardbands used for timing and power signoff include compensatable systematic variations, the result is overdesign and a more difficult design closure task. With this in mind, a number of recent works have proposed systematic process variation-aware design analyses to 'close the loop' from manufacturing simulation back to the design flow.

Balasinski *et al.* [6] propose a methodology of manufacturability qualification for ultra-deep submicron circuits, based on optical simulation of the layout, integrated with device simulation; see also [21]. Pack *et al.* [18] propose to incorporate advanced models of lithographic printing effects into the design flow to improve yield and performance verification accuracy. Gupta *et al.* [11] observe that lithography simulation permits post-OPC (optical proximity correction) estimation of on-silicon feature sizes at different process conditions. Yang *et al.* [24] address post-lithography based analysis and optimization, proposing a timing analysis flow based on residual OPC errors (equivalent to lithography simulation output) for timing-critical cells and their layout neighborhoods. Cao *et al.* [8] propose a methodology for standard-cell characterization consid-

TABLE I
INVERTER DELAY FOR DIFFERENT P, V, AND T CORNERS

Process		Voltage (V)	Temperature (C)	Delay (ps)
NMOS	PMOS			
Fast	Fast	1.0	-40	22.17
Fast	Fast	1.0	125	22.54
Fast	Fast	0.9	-40	27.21
Fast	Fast	0.9	125	26.16
Slow	Slow	1.0	-40	31.44
Slow	Slow	1.0	125	30.63
Slow	Slow	0.9	-40	42.78
Slow	Slow	0.9	125	38.89

ering litho-induced systematic variations. In [8], the objective is to enable efficient post-litho analysis by running litho-aware characterization. Furthermore, to minimize the difference between isolated and actual placement contexts of a given standard cell, vertical dummy poly patterns are inserted at the cell boundary. Finally, it is noteworthy that Gupta and Heng [10] perform "iso-dense aware" timing analysis (based on modeling of systematic through-focus Leff variation) to achieve up to 40% reduction of the BC-WC guardband in static timing analysis. Also, Sylvester *et al.* [27] observe that up to 60% of BEOL guardband can be eliminated by use of the realistic BEOL variation model.

Despite such vigorous research activity in this arena, a fundamental question remains open: What is the impact of the guardband on design quality? And, what is the specific return that we can expect to be realized by the design team from availability of, e.g., iso-dense aware timing analysis [10], post-lithography based analysis and optimization [24], or any other potential path to reduced guardband? The following sections describe our efforts toward a quantified answer to this question.

III. MODEL AND GUARDBAND SCALING

A. Impact of PVT on Circuit Delay

To quantify the impacts of guardband reduction on design process outcomes, we first quantify the existing guardband in foundry delay models. Guardband exists in the form of delay for each process, voltage, and temperature (PVT) corner (i.e., delay tables in Liberty model files). Since each axis of PVT will have different delay impact, we quantify the impact due to each P, V, and T corner separately.

1) Standard Cells: To assess the impact of each P, V, and T parameter on standard cell delay, we run SPICE simulations for a simple inverter cell across eight possible combinations of PVT, i.e., $\{P_{\text{slow}}, P_{\text{fast}}\} \times \{V_{\text{low}}, V_{\text{high}}\} \times \{T_{\text{low}}, T_{\text{high}}\}$.[1] Table I shows the delay values of 65-nm inverter cell for all PVT combinations. Of the $1.8X$ difference between worst-/best-case PVT corners, $1.46X$ is from process, $1.25X$ is from voltage and $0.97X$ is from temperature (i.e., due to reverse temperature effect).

[1]Slew (39.2 ps) and load capacitance (4.9 fF) values are selected from the third row and column indices of the 7 × 7 65-nm Liberty delay table.

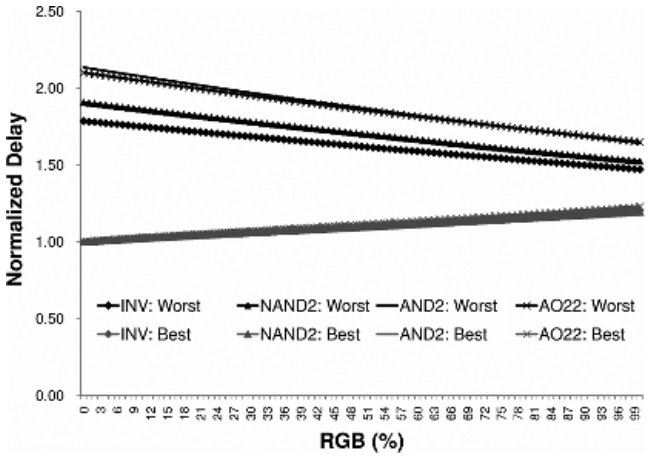

Fig. 1. Worst-/best-case delay changes of an inverter, a 2-input AND gate, a 2-input NAND gate, and a 2-input AND-OR gate versus process guardband.

If any of the P, V, and T parameters are fixed for reasons such as test specifications or customer requests, this will limit the actual achievable guardband reduction. Fig. 1 shows worst-/best-case delay changes *with only* process guardband reduction. To determine this, we perform a set of fine-grained SPICE simulations with fixed V and T. We create 100 SPICE models by interpolating between FF and SS models with step size of 1% (i.e., corresponding to 1% guardband reduction).

We then measure rise and fall delay of four standard cells including an inverter cell (INV1), a 2-input NAND gate (NAND2), a 2-input AND gate (AND2), and a 4-input AND-OR gate (AO22), using the corresponding interpolated SPICE models. Fig. 1 shows normalized worst- and best-case delay values of the above cells. We take average rise and fall delay, and normalize the worst- and best-case delay of each cell to the delay value of the cell at the original best-case process corner (i.e., 0% RGB), respectively. We observe that delay at worst-case (best-case) decreases (increases) with reducing process guardband. We observe that the decreasing (increasing) rate of delay change does not have a significant relationship with the functional complexity of cell. At 100% guardband reduction, FF and SS have the same SPICE model and hence, the delay difference in Fig. 1 is due only to temperature and voltage guardband.

Also, Fig. 2 shows the delay change percentage, for worst- and best-case corners, of the above four cells, when the process guardband reduces from 0% to 100%. We observe that the worst-case delay change of complex cells are larger than that of an inverter. The best-case delay change of a NAND2 is the smallest among the four cells and is within 1.07% of that of the inverter.

2) Memory Cells: Since SRAM occupies a significant portion of today's SoC designs, we also assess the impact of guardband reduction on SRAM performance. A 6T SRAM bitcell is composed of six transistors, two bitlines (BL and BLb), and one word line (WL). A bit of data is stored in the complementary internal nodes nl and nr, when WL is "1". The transistors are classified as pass (or access) transistors (C1 and C2), pull-down

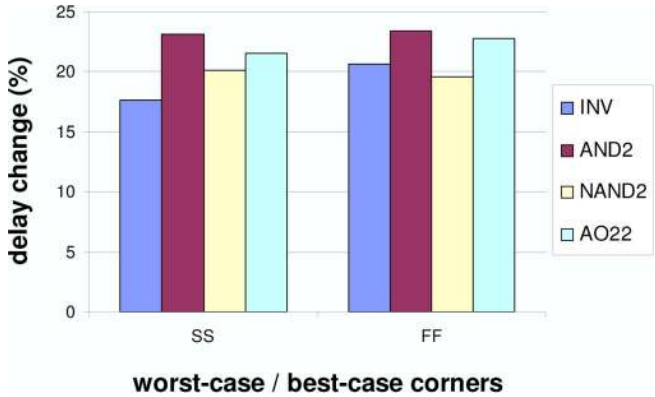

Fig. 2. Worst-/best-case delay change percentage across 0%–100% RGB.

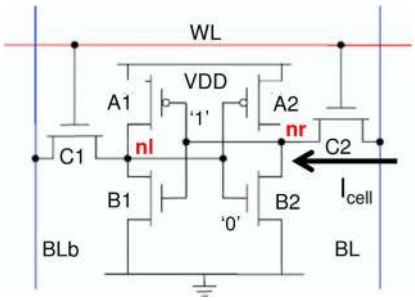

Fig. 3. Schematic circuit diagram for a 6T SRAM bitcell.

transistors (B1 and B2), and pull-up transistors (A1 and A2) as shown in Fig. 3.

During read operation, one of the pre-charged bitlines is discharged through a pass transistor and its associated pull-down transistor (i.e., C1-B1 or C2-B2), and the sense amplifier detects voltage difference between the two bitlines. I_{cell} is the maximum current that flows during the read operation, and can be used as an SRAM performance metric. We measure worst-/best-case I_{cell} of a 65-nm SRAM bitcell with the interpolated 100 SPICE models used for inverter delay simulation. Fig. 4 shows worst-/best-case I_{cell} changes *with only* process guardband reduction. According to the figure, best-case (worst-case) I_{cell} decreases (increases) with reducing process guardband. Since I_{cell} is inversely proportional to SRAM delay, I_{cell} increase at the worst-case corner implies SRAM delay decrease. However, the performance of SRAM depends not only on the I_{cell}, but also on the sense amplifier's reaction speed and digital logic signal propagation speed in the peripherals of SRAM. Fig. 5 shows the normalized delay of an SRAM bitcell, which is derived from I_{cell} simulation results, and the normalized delay of an inverter. We observe that the delay of an SRAM bitcell is more sensitive to the guardband reduction than that of an inverter. Hence, we can conclude the logic delay improvement from the worst-case guardband reduction can speed up both standard logic cells and embedded SRAMs.

B. Liberty Model Scaling

In corner-based design and signoff methodologies, there are best-case and worst-case design behaviors for which cells are

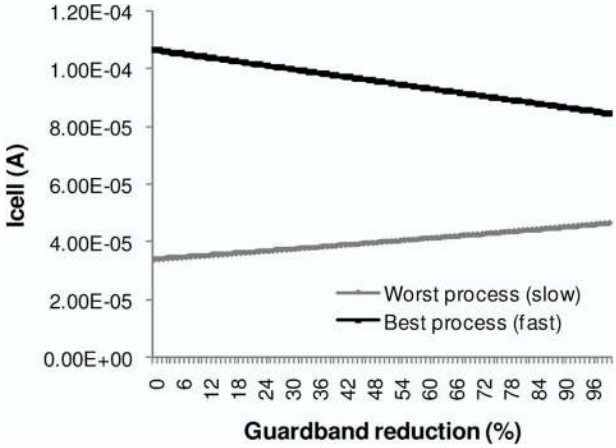

Fig. 4. Worst-/best-case I_{cell} changes of a 65-nm SRAM bitcell versus process guardband.

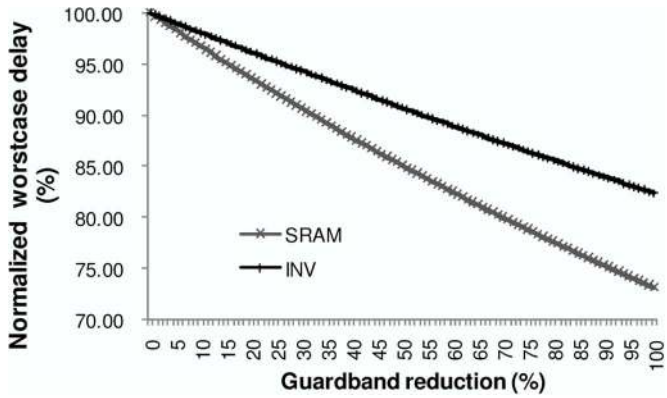

Fig. 5. Normalized worst-case delay of 65-nm inverter (INV) and SRAM bitcell versus process guardband.

characterized, and which are captured in respective Liberty (.lib) format libraries. In the Liberty format, each standard cell master has several attributes, such as pin type, loads, stimuli and lookup-table indices. The data available in the Liberty format include capacitance, thresholds/switching points, rise time, fall time, and power values of each cell in the library. Static timing analysis operates independently of characterization, reading both a Verilog netlist and multiple timing libraries. To use the delay changes from the guardband reduction, new characterization must be performed for each guardband value. However, the cell characterization process is very time-consuming. Instead, we can directly reduce the delay guardband by linear scaling of timing libraries used for SP&R, since delay varies linearly with guardband reduction as shown in Figs. 1 and 5. In our experiments, we run through a traditional timing-driven SP&R flow; hence, we scale only the input pin capacitances and timing tables, and we do not modify the power tables of the .lib files.

It is well-known that one can specify "PVT" scaling factors in the technology library environment, using so-called k-factors. These k-factors (so-called because they are attributes with names starting with $k\_$) are multipliers that scale defined library values, allowing consideration of the effects of changes in

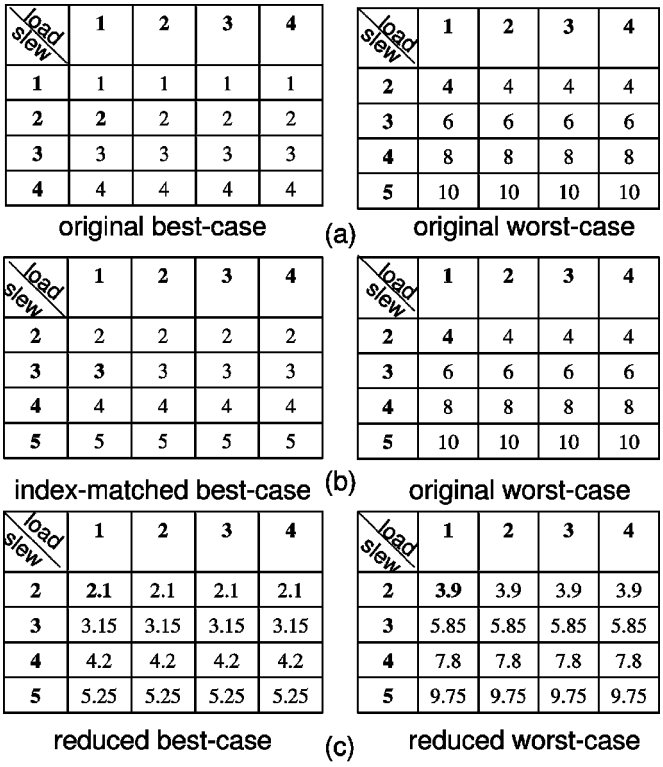

Fig. 6. Illustration of steps in guardband reduction for timing tables of the Liberty (.lib) files. (a) Original best/worst-case tables. (b) New best-case table with input slew time indices matched up with those of the worst-case table. (c) 10% guardband reduction, computed on an entry-by-entry basis, across all the table entries.

Input: best/worst-case libraries.
Output: index-matched best-case library.

for all the cells in the best-case library:
find the corresponding cell in the worst-case library.
interpolate/extrapolate the new best-case timing table entries using the best/worst-case values.
copy the slew rate index of the worst-case table on to that of the best-case table.

Fig. 7. Index matching procedure.

process, voltage and temperature [1]. However, in our methodology we do not use k-factors since they cannot correctly capture guardband reduction. Instead, we apply an entry-by-entry library scaling methodology in which 1) the difference between values of a certain table entry in two libraries (e.g., worst-case and best-case) is computed and 2) then, the amount of required guardband reduction is applied to this difference and the corresponding (e.g., best- and worst-case) table values are modified accordingly.

Fig. 6 illustrates the steps required to scale timing tables within the Liberty files.

- *Goal: Entry-by-entry BC-WC guardband reduction.* Fig. 6(a) shows an example of timing tables within best- and worst-case Liberty files.[2] Our goal is to apply a uniform percentage of guardband reduction to each entry-by-entry difference (i.e., the amount of guardband associated with each delay value) between best-case and worst-case delay values, which are characterized under the corresponding PVT conditions.[3] Note that we cannot simply reduce values of worst-case delays, and increase values of best-case delays, by fixed percentages; this will not result in a uniform guardband reduction.
- *Index matching step.* In a production timing library, it is common for, e.g., the input slew time indices of the best-case library to be different from the indices of the worst-case library. Hence, before we can scale entry-by-entry guardband values, we must first match up the indices of corresponding tables in the best-case and worst-case libraries. We achieve this by interpolation/extrapolation from the original index values of both tables, as illustrated by the "index-matched best-case" table in Fig. 6(b)
- *Calculation of entry-by-entry guardband reduction.* After unifying the library table indices, we can compute the entry-by-entry difference (i.e., original amount of guardband) and apply the necessary guardband reduction. For example, in Fig. 6(b), we see that for input slew time $= 2$ and capacitive load $= 1$, the best-case and worst-case delay values are 2 and 4, respectively. To reduce the guardband by 10%, we first find the difference between corresponding values (i.e., $4 - 2 = 2$). Then, we add 5% of this difference to the best-case value, and subtract 5% of this difference from the worst-case value. The resulting guardband-reduced BC/WC values are seen in Fig. 6(c). We more formally describe our index-matching and guardband reduction procedures in Figs. 7 and 8.[4]
- *Scaling of pin capacitance guardband.* Note that input pin capacitance values can be considered as 1×1 tables. Hence, the same guardband reduction methods are applied to them as well.

C. Interconnect Model Scaling

It is commonly accepted that interconnect has become a dominant factor in determining circuit performance. In sub-100-nm processes, litho- and CMP-induced variations in conductor width, conductor thickness, and inter-layer dielectric (ILD) height within the BEOL stack can cause significant variation of interconnect parasitics.

In the corner-based design methodology, extreme values of resistance and capacitance are used to obtain worst-case and best-case corners in timing analysis. For example, in best-case analysis we use the smallest capacitance value, and in worst-case analysis we use the largest capacitance value. Resistance behaves inversely to capacitance, hence minimum resistance is used in worst-case analysis and maximum resistance is used in best-case analysis. In addition to process variations, operating conditions such as temperature affect resistance and capacitance

[2]The tables shown in Fig. 6 are for illustrative purposes. Neither their indices nor their entries represent realistic values.

[3]PVT condition for best (worst) case is fast (slow) transistors, high (low) supply voltage and low (high) temperature.

[4]In Fig. 8, the factor 1/200 arises because half of the $x\%$ guardband reduction is applied to each of the best-case and worst-case values.

TABLE II
R AND C COMPARISON AND SCALING METHOD FOR 90-nm INTERCONNECT

		Best corner	Worst corner
P&R	Resistance ratio	1	1.11
	Resistance scaling factor for x% of guardband reduction	$1+\frac{x}{200}\cdot(1.11-1)$	$1-(1-\frac{1}{1.11})\cdot\frac{x}{200}$
	Capacitance ratio	1	1.15
	Capacitance scaling factor for x% of guardband reduction	$1+\frac{x}{200}\cdot(1.15-1)$	$1-(1-\frac{1}{1.15})\cdot\frac{x}{200}$
Signoff	Resistance ratio	1	1.13
	Resistance scaling factor for x% of guardband reduction	$1+\frac{x}{200}\cdot(1.13-1)$	$1-(1-\frac{1}{1.13})\cdot\frac{x}{200}$
	Capacitance ratio	1	1.17
	Capacitance scaling factor for x% of guardband reduction	$1+\frac{x}{200}\cdot(1.17-1)$	$1-(1-\frac{1}{1.17})\cdot\frac{x}{200}$

Input: index-matched best/worst-case libraries and x% guardband reduction.
Output: guardband reduced best/worst-case libraries.

for all the common cells in the best/worst-case libraries:
  for each entry in a best-case table ($value_{best}$):
    $value_{best} = value_{best} + \frac{x}{200}(value_{worst} - value_{best})$.
  for each entry in a worst-case table ($value_{worst}$):
    $value_{worst} = value_{worst} - \frac{x}{200}(value_{worst} - value_{best})$.

Fig. 8. Guardband reduction procedure.

values. In 90-nm copper technology, large temperature variation (e.g., from $-40°C$ to $125°C$) can lead to 50% increases in resistance. From Table II, including the process and temperature effects, we see that at the worst interconnect corner, the values of capacitance and resistance are greater than those at the best interconnect corner by 17% and 13%, respectively.

We implement model guardband reduction for interconnect resistance and capacitance as follows.

- We first extract resistance and capacitance from a sample design for best and worst corners using a signoff extractor (Synopsys Star-RCXT).
- We compare the mean of the worst-corner values with that of the best-corner values.
- Finally, for a given percentage reduction in guardband, we find proper scaling factors for each corner by a method similar to that described above for Liberty scaling. The scaling equations and the relative values of interconnect capacitance and resistance for 90-nm technology are summarized in the Table II.[5]

IV. IMPLEMENTATION FLOW AND TESTCASES

A. Timing-Driven Implementation Flow

Fig. 9 shows the traditional SP&R flow that we have scripted for "push-button" use in our experiments. The steps in Fig. 9 represent the major physical design steps. At each step, we require that the design must meet the timing requirements before it can pass on to the next step. (This is standard practice, since the later in the design flow, the harder it is to fix a given timing violation.) In other words, in the event of any timing violation, our implementation flow goes back to the previous step through a return path and fixes the violation.

In the flow, we first synthesize RTL codes with worst-corner libraries. This synthesis step, when different reduced-guardband libraries are used, produces initial netlists with different total standard-cell area. We fix the utilization ratio in all testcases at the floorplan stage. We optimize timing inside the P&R tool using its embedded RCX and delay calculation engines. Since the designer's concern is generally to obtain the best performance within given environments and constraints, we concentrate on fixing setup violations at this stage of the implementation flow. Once all setup violations are cleared, it is necessary to fix hold violations using the best-case library. While attempting to fix hold violations, sometimes new setup violations are created, and iteration over the above steps is required until all violations are cleared at both the best and worst timing corners.

B. Testcases and Tools

We use four benchmark designs in our experiments. The first two are the *aes* and *jpeg* cores, obtained as RTL from the open-source site *opencores.org* [3]. The third testcase is *5Xjpeg*, which is composed of 5 copies of the *jpeg* core. The fourth is an embedded processor core provided by Qualcomm, Inc. [4]. For the first three testcases we perform our experiments using front-end libraries in TSMC 90- and 65-nm technologies. For the fourth testcase we use foundry 45-nm libraries. The *aes* core typically synthesizes to approximately 16 K instances; target clock frequency is 400 MHz in 90 nm and 600 MHz in 65 nm. The *jpeg* (resp. *5Xjpeg*) core typically synthesizes to approximately 64 K (resp. 320 K) instances; target clock frequency is 300 MHz in 90 nm and 500 MHz in 65 nm. The embedded processor has approximately 67 K instances; target frequency is 500 MHz in 45 nm. We use *Cadence RTL Compiler v05.20-s009_1* to synthesize the open-source designs and use *Synopsys Design Compiler v2007.12-SP4* to synthesize the embedded processor. We use *Cadence SOC Encounter v5.2* and *Cadence SOC Encounter v7.1 usr2* to execute the P&R flow on open-source and embedded processor testcases, respectively. Initial row utilizations are 40%, 60%, 60% and 65% for the *aes*, *jpeg*, *5Xjpeg* and embedded processor designs, respectively. Note that final row utilizations may change depending on timing optimization steps (e.g., buffering, sizing, etc.) that are executed during the P&R flow. We use *Synopsys Design Compiler v2006.06-SP3* for scan insertion and *Synopsys Star-RCXT v2006.06-SP1* for RCX. Finally, *Synopsys PrimeTime v2005.12-SP3* is used for static timing analysis.

[5]Note that since the P&R tool (Cadence SOC Encounter) and the signoff extraction tool (Synopsys Star-RCXT) have discrepancies in their computed interconnect resistance and capacitance values, we compute separate scaling factors for each. (Analogous scaling factors are separately computed for P&R and signoff extraction in the 65-nm technology.)

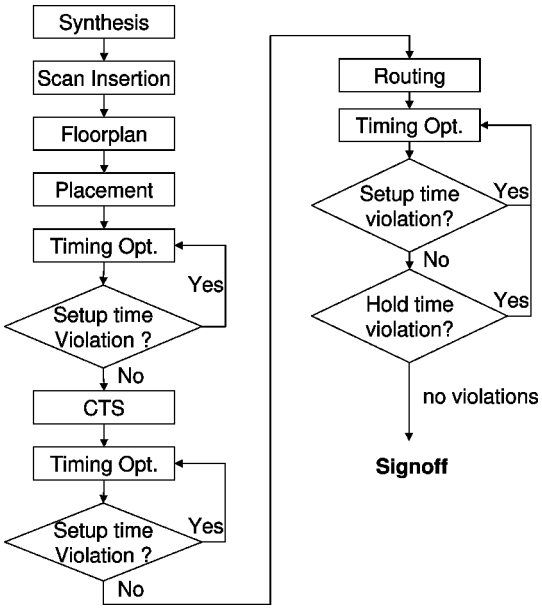

Fig. 9. Implementation (synthesis, place, and route) flow.

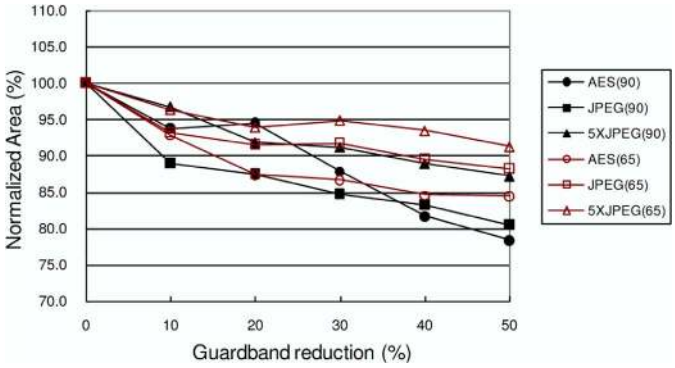

Fig. 10. Area versus guardband reduction.

V. Experimental Results and Discussion

In our experiments, for *aes*, *jpeg*, and *5Xjpeg* testcases we run the entire implementation flow with six sets of libraries corresponding to model guardband reductions of 0%, 10%, 20%, 30%, 40%, and 50%. We do this for each of three cases: 1) only back-end-of-line (BEOL) guardband reduction, 2) only front-end-of-line (FEOL) guardband reduction, and 3) both BEOL and FEOL guardband reduction—in order to separately observe the impact of FEOL and BEOL guardband reduction. Last, we do this for each of 90- and 65-nm technologies. As a result, each testcase is implemented with the scripted flow of Fig. 9 a total of $6 \times 3 \times 2 = 36$ separate times, 18 times in each technology. However, for the embedded processor testcase we only consider Case (2), hence we only implement the testcase 6 times in 45 nm.

In the following, we use "F" or "FE" as shorthand for FEOL; "B" and "BE" are shorthand for BEOL. We also give detailed tables of numerical data for the 90-nm *jpeg* core implemented with 300 MHz target frequency and for the 45-nm embedded processor core with 500 MHz target frequency. Other results are presented more compactly in graphical form.

A. Impact on Quality of Results

We assess impact of guardband reduction with respect to design quality metrics of area, routed wirelength, dynamic and leakage power. Table III shows the impact of guardband reduction on the area (i.e., the sum of all standard cell areas within the design) for the 90-nm *jpeg* core implemented with 300 MHz target frequency. Table IV shows the impact of guardband reduction on total wirelength. For power estimation, we consider two different scenarios: 1) foundries reduce the guardband through process enhancement or 2) foundries simply reduce their guardband without process enhancement or changing operating condition. Tables V and VI show the impact of guardband reduction on dynamic and leakage power, respectively, for Scenario (2). We note that the power reduction comes from the reduced area. Power values, especially leakage power, cannot be obtained by linear interpolations/extrapolations as used in delay scaling. Although we did not re-characterize cell power, we expect worst-case power to increase and best-case power to decrease since power and delay typically change in opposite directions. We also expect that power reduction from the area reduction will still be valid for Scenario (1). Figs. 10–13 show the impact of guardband reduction of both FE and BE on area, routed wirelength, dynamic power and leakage power for *aes*, *jpeg* and *5Xjpeg* designs using 90- and 65-nm technologies, respectively. We observe that area, wirelength, power metrics are "well-behaved"; they improve (decrease) as the percentage guardband reduction increases. At the level of 40% guardband reduction achieved by the variational timing approach from IBM [10], reductions of nearly 18% area, over 21% wirelength, 20% dynamic power and 29% leakage power are achieved, on average.[6] Somewhat surprisingly, guardband reduction for interconnect (BEOL) parasitics has much less impact on design quality than guardband reduction for FEOL models. In addition, Tables VII and IX show the impact of guardband reduction with respect to area, and dynamic and leakage power, for the 45-nm embedded processor core. We observe that at 30% guardband reduction, area, dynamic power, and leakage power reduce by 8%, 5%, and 10%, respectively.

1) Analysis of an Example Critical Path: It is instructive to look more closely at the effect of guardband reduction on timing modeling and analysis. Table X shows the average cell delays in a critical path of the 90-nm *jpeg* implementation, for both best-case and worst-case corners, across different guardband reductions. We see that a 10% reduction in guardband increases (decreases) the best (worst) average stage delay by only 4 ps (3% of the average stage delay). Also, the delay differences across

[6]In [15], Kahng and Mantik observed the existence of "inherent noise" in IC implementation tools, and documented up to 12% change in quality of result (e.g., total post-route wirelength) due to the tools' sensitivity to such noise sources as input renaming, randomization, scaling, etc. We note this previous work because it implies a limit to cleanliness of experimental data as we trace impact of guardband reduction through the tool flow. Also, inherent tool noise may swamp any benefits of guardband reduction in certain design regimes (e.g., with respect to tightness or looseness of timing and area constraints).

TABLE III
AREA VERSUS GUARDBAND REDUCTION FOR 90-nm *jpeg* DESIGN AT 300 MHz

Area	0%		10%		40%		50%	
	mm^2	%	mm^2	%	mm^2	%	mm^2	%
F	0.367	100	0.356	97.0	0.339	92.3	0.331	90.3
B	0.367	100	0.367	100.1	0.357	97.5	0.355	96.7
F+B	0.367	100	0.355	96.9	0.339	92.4	0.331	90.3

TABLE IV
TOTAL WIRELENGTH VERSUS GUARDBAND REDUCTION FOR 90-nm *jpeg* DESIGN AT 300 MHz

WL	0%		10%		40%		50%	
	mm	%	mm	%	mm	%	mm	%
F	1609.2	100	1608.6	99.9	1544.3	96.0	1512.2	94.0
B	1609.2	100	1617.2	100.5	1586.6	98.6	1590.1	98.8
F+B	1609.2	100	1593.3	99.0	1539.0	95.6	1514.8	94.1

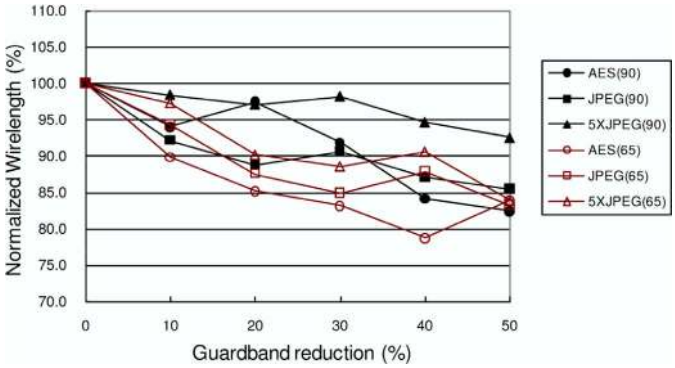

Fig. 11. Total wirelength versus guardband reduction.

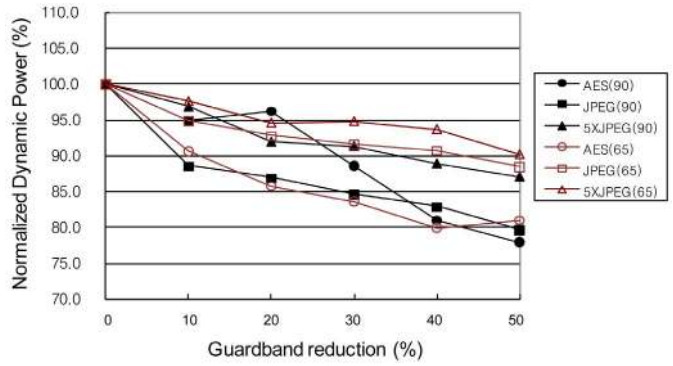

Fig. 12. Total dynamic power versus guardband reduction.

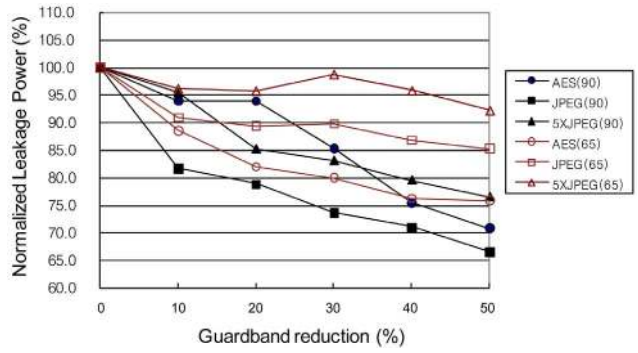

Fig. 13. Guardband reduction versus total leakage power.

different guardband reductions in the BEOL are very small compared to the differences in the FEOL. Possibly, the impact of BEOL guardband reduction (despite being expected and evident from our data) will not always be visible due to inherent noise in EDA implementation tools [15]. The results of Table X are in alignment with the trends we observe for area and wirelength versus guardband reduction above.

B. Impact on Design Cycle Time

Table XI shows the substantial impact of guardband reduction on total SP&R flow runtime for the 90-nm *jpeg* testcase. Also, Fig. 14 shows the impact of guardband reduction on total SP&R flow runtime for *aes*, *jpeg*, and *5Xjpeg* designs using 90- and 65-nm technologies. The data shows up to 41% reduction in SP&R flow runtime with a 40% guardband reduction. Table XII shows that total SP&R flow runtime decreases by 7% with 30% guardband reduction for the 45-nm embedded processor core. In real-world design contexts, such a substantial reduction in SP&R runtime can, at a minimum, reduce tapeout schedule risk, and permit additional optimization iterations and design space explorations. A substantial reduction in SP&R flow runtime can also reduce time to market for an IC product.

Another very clear benefit from guardband reduction can be seen from analysis of violations in signoff analysis. Recall that if there are violations at the signoff stage, then it is necessary to go back to the P&R stage and fix them. The number of design iterations needed to fix violations is reflected by a variety of "figure of merit" parameters that are often tracked by designers, e.g., total number of violations, worst negative slack (WNS), and total negative slack (TNS). These three metrics represent different views of the design timing characteristics.

- Total number of violations represents how many violating points the designer needs to worry about.
- WNS represents the largest timing violation.
- TNS indicates how difficult fixing all the current violations in a design can be.

From these numbers, we can estimate the difficulty of meeting timing constraints, and how much iteration will be required. For example, from the total number of violations and TNS of hold time analysis, the designer can estimate how many buffers are needed to fix the violations, and indirectly estimate how much the standard-cell area will increase as a result. Or,

TABLE V
DYNAMIC POWER VERSUS GUARDBAND REDUCTION FOR 90-nm *jpeg* DESIGN AT 300 MHz

P_{dyn}	0%		10%		40%		50%	
	mW	%	mW	%	mW	%	mW	%
F	114.1	100	102.8	90.02	93.6	82.00	91.5	80.19
B	114.1	100	111.2	97.46	106.1	93.01	107.5	94.21
F+B	114.1	100	101.0	88.52	94.5	82.81	90.8	79.59

TABLE VI
LEAKAGE POWER VERSUS GUARDBAND REDUCTION FOR 90-nm *jpeg* DESIGN AT 300 MHz

P_{leak}	0%		10%		40%		50%	
	mW	%	mW	%	mW	%	mW	%
F	0.250	100	0.210	84.00	0.175	69.89	0.167	66.56
B	0.250	100	0.243	97.04	0.226	90.15	0.229	91.88
F+B	0.250	100	0.204	81.61	0.178	71.06	0.166	66.50

TABLE VII
AREA VERSUS GUARDBAND REDUCTION FOR 45-nm EMBEDDED PROCESSOR CORE AT 500 MHz

Area	0%		10%		30%		50%	
	mm^2	%	mm^2	%	mm^2	%	mm^2	%
F	0.175	100	0.174	99.48	0.163	92.81	0.155	88.79

TABLE VIII
DYNAMIC POWER VERSUS GUARDBAND REDUCTION FOR 45-nm EMBEDDED PROCESSOR CORE AT 500 MHz

P_{dyn}	0%		10%		30%		50%	
	uW	%	uW	%	uW	%	uW	%
F	112.29	100	110.12	98.07	107.58	95.81	100.98	89.93

TABLE IX
LEAKAGE POWER VERSUS GUARDBAND REDUCTION FOR 45-nm EMBEDDED PROCESSOR CORE AT 500 MHz

P_{leak}	0%		10%		30%		50%	
	uW	%	uW	%	uW	%	uW	%
F	2.063	100	2.064	100.05	1.867	90.50	1.685	81.68

TABLE X
CRITICAL PATH DELAY VARIATIONS ACROSS DIFFERENT GUARDBAND REDUCTIONS

Cases	GB reduction	Timing corner	Total path delay (ns)	Average stage delay (ns)
	0%	WC	3.520	0.147
		BC	1.435	0.060
F	10%	WC	3.406	0.142
		BC	1.525	0.064
	40%	WC	3.069	0.128
		BC	1.813	0.076
	50%	WC	2.960	0.123
		BC	1.910	0.080
B	10%	WC	3.515	0.146
		BC	1.437	0.060
	40%	WC	3.502	0.146
		BC	1.443	0.060
	50%	WC	3.497	0.146
		BC	1.445	0.060
F+B	10%	WC	3.410	0.142
		BC	1.523	0.063
	40%	WC	3.085	0.129
		BC	1.804	0.075
	50%	WC	2.979	0.124
		BC	1.899	0.079

the designer can use the WNS value to see how close a design is to becoming feasible with respect to timing constraints.

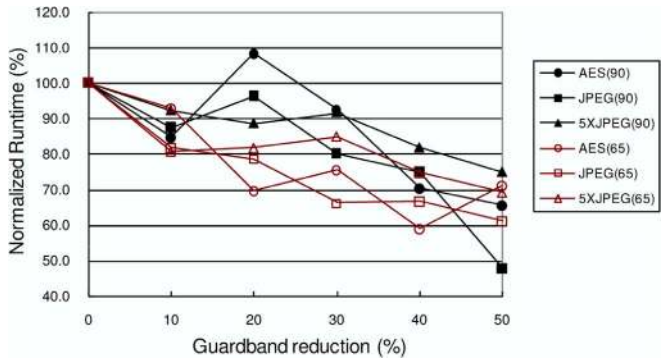

Fig. 14. Guardband reduction versus total SP&R flow runtime.

Table XIII shows various figures of merit for the 90-nm *jpeg* post-P&R result obtained with a 0% guardband reduction, when evaluated using other (10%, 40%, 50%) guardband reductions. The table gives number of violations, worst negative slack, and total negative slack, with respect to both setup and hold constraints using signoff flow. Here, we can see very substantial benefits from guardband reduction, e.g., with a 40% guardband reduction, the vast majority of timing violations are erased, and the WNS and TNS metrics are also reduced substantially (by

TABLE XI
GUARDBAND REDUCTION VERSUS TOTAL SP&R FLOW RUNTIME FOR 90-nm *jpeg* DESIGN AT 300 MHz

Runtime	0%		10%		40%		50%	
	sec	%	sec	%	sec	%	sec	%
F	7129	100	5653	79.3	4068	57.1	4061	57.0
B	7208	100	7327	101.7	7507	104.1	5755	79.8
F+B	6950	100	5729	82.4	4366	62.8	4061	58.4

TABLE XII
GUARDBAND REDUCTION VERSUS TOTAL SP&R FLOW RUNTIME FOR 45-nm EMBEDDED PROCESSOR CORE AT 500 MHz

Runtime	0%		10%		30%		50%	
	sec	%	sec	%	sec	%	sec	%
F	13032	100	12155	93.27	12353	94.79	10662	81.81

TABLE XIII
GUARDBAND REDUCTION VERSUS NUMBER OF VIOLATIONS, WORST NEGATIVE SLACK (WNS) AND TOTAL NEGATIVE SLACK (TNS)

			Guardband reduction			
			0%	10%	40%	50%
F	Setup	# of viols	235	3	0	0
		WNS (*ns*)	-0.126	-0.016	0	0
		TNS (*ns*)	-9.95	-0.03	0	0
	Hold	# of viols	4414	675	526	287
		WNS (*ns*)	-0.116	-0.045	-0.030	-0.028
		TNS (*ns*)	-259.68	-15.19	-4.20	-1.06
B	Setup	# of viols	235	231	203	198
		WNS (*ns*)	-0.126	-0.121	-0.11	-0.10
		TNS (*ns*)	-9.95	-8.97	-6.29	-5.43
	Hold	# of viols	4414	4410	4404	4400
		WNS (*ns*)	-0.116	-0.116	-0.116	-0.116
		TNS (*ns*)	-259.68	-259.39	-259.59	-258.34
F+B	Setup	# of viols	235	3	0	0
		WNS (*ns*)	-0.13	-0.011	0	0
		TNS (*ns*)	-9.95	-0.02	0	0
	Hold	# of viols	4414	676	524	298
		WNS (*ns*)	-0.116	-0.045	-0.030	-0.034
		TNS (*ns*)	-259.68	-15.24	-4.30	-1.11

up to 100%). This will clearly improve timing convergence by reducing design iterations.

C. Impact of Guardband Reduction on Design Yield

Guardbanding exists in today's design methodologies to help guarantee high yield in spite of process variability. In this subsection, we quantify the impact of guardband reduction on design yield. We believe that such quantification will be an essential part of manufacturing-aware design methodologies in the future.

Overall yield is modeled as the product of *random* defect yield, which depends on die area, and *parametric* yield, which is independent from die area

$$Y = Y_r \cdot Y_p. \tag{1}$$

1) Random Defect Yield (Y_r) *:* A variety of models exist for the spatial distribution of random electrical faults across a wafer, and random defect yield Y_r. The fundamental difference between these models is the assumed distribution of the random defects [16]. Commonly, random defects are characterized by defect density parameter d, and clustering parameter α. The average number of defects on a chip of area A is Ad. The number of defects x in a random chip is an integer-valued random variable, and the observed phenomenon of defect clustering is effectively modeled by assuming a negative binomial probability density function for x [7]

$$\begin{aligned} p(x) &= Prob(\text{number of defects on chip} = x) \\ &= \frac{\Gamma(\alpha+x)\left(\frac{Ad}{\alpha}\right)^x}{x!\Gamma(\alpha)\left(1+\frac{Ad}{\alpha}\right)^{\alpha+x}} \end{aligned} \tag{2}$$

where $\Gamma(x)$ is the Gamma function. The yield with respect to random defects is obtained as the probability $p(0)$ of having no defect on a chip. Substituting $x = 0$ in (2)

$$Y_r = \left(1 + \frac{Ad}{\alpha}\right)^{-\alpha}. \tag{3}$$

If we use $\alpha = \infty$, which corresponds to the case of unclustered defects, (3) gives a *Poisson* density function with mean Ad, and the yield with respect to random defects is pessimistically estimated as

$$Y_r = e^{-Ad}. \tag{4}$$

From (4), we conclude that random defect yield (Y_r) will increase with decreasing area (A) accomplished by guardband reduction. Other widely used random defect yield models are Murphy and Bose-Einstein as shown in (5) and (6), respectively [16]

$$Y_r = \left(\frac{1 - e^{-Ad}}{Ad}\right)^2 \tag{5}$$

$$Y_r = \frac{1}{(1+Ad)^n}. \tag{6}$$

In Bose-Einstein model n is the complexity factor. A comparison of the above yield models shows that for small defect densities $(0.2/\mathrm{cm}^2)$), all three models predict similar yield results. Even for larger defect densities (i.e., 1 and $2/\mathrm{cm}^2$), for die areas less than $100\ \mathrm{mm}^2$, the deviations are within 5% [16].[7]

[7]In this paper, we assume a Poisson model for random defect yield estimation.

TABLE XIV
RANDOM DEFECT YIELD FOR 65-nm *5Xjpeg* DESIGN

RGB	Chip area (cm^2)	Y_r from Eq. (4)	Y_r from Eq. (5)	Y_r from Eq. (6)	Y_r from EYES
0	0.014562	0.99709	0.99709	0.99709	0.9850
10	0.014205	0.99716	0.99716	0.99716	0.9855
20	0.014084	0.99719	0.99718	0.99719	0.9867
30	0.014093	0.99719	0.99718	0.99718	0.9832
40	0.014219	0.99716	0.99716	0.99716	0.9865
50	0.013992	0.99722	0.99720	0.99720	0.9880

In addition, hypothetically, reduced chip area could decrease wire spacing which would then increase the likelihood of short defects. Hence, we perform random defect yield analysis using EYES [5]. The EYES (Edinburgh Yield Estimator–Sampling) tool uses a sampling technique to estimate the properties of the IC layout as a whole. We use a Poisson yield model and account for both open and short faults in the same layer. We do not consider inter-layer faults such as dielectric and pinhole faults. In our experiments, we analyze random defect yield of GDSII for *5Xjpeg* implemented with each reduced guardband. Table XIV shows the random defect yield values from (4), (5), and (6) and EYES for 65-nm *5Xjpeg* design. In Bose-Einstein model we use the physical chip area, A, and an average defect density, d.[8] We use defect density of 0.2 $(/\mathrm{cm}^2)$ for all the equations as well as EYES experiments.

Due to the small size of the sample design, the resulting yield values are not significantly different for each guardband. However, it is clear that random defect yield does not decrease with the guardband reduction.[9]

In the simple approach, the critical area, to be used in the above models, is equal to the die area. However, there needs to be a refinement by adding up the active area of the logic, memory, and IO cells and assigning different defect density values to each of these components. Assuming that the wafer fab provides a single, average d, we can use a simple approach that assigns a 30% addition to d for memory blocks and a 20% reduction to d for IO cells. Indeed, the proper way is to get yield information from chips with logic only, and memory only, and then calculate defectivities for each [16]. Therefore, (4) is modified as follows:

$$Y_r = e^{-(A_{\mathrm{memory}} d_{\mathrm{memory}} + A_{logic} d_{logic} + A_{IO} d_{IO})} \tag{7}$$

where, A_{memory}, A_{logic}, A_{IO} denote memory, logic, and IO cell physical area, and d_{memory}, d_{logic}, d_{IO} denote memory, logic, and IO cell defect density values, respectively.

2) Parametric Yield (Y_p): Yield with respect to parametric variation, Y_p, can be estimated by considering a normal distribution with best-case and worst-case corners being set at -3σ and 3σ, respectively. The 3σ window can be taken to define the original guardband (i.e., 0% guardband reduction, with range 6σ).[10] Then, assuming no change in manufacturing variability, a $K\%$ design guardband reduction would result in a reduced range of $(6\sigma)(100-K)/100$. To calculate the parametric yield impact of design guardband reduction with no change of manufacturing variability, we may use the error function (erf, i.e., cumulative distribution of the normal distribution) for the appropriate range. For example, $Y_p(RGB\%)$ with respect to 0% guardband reduction can be computed as

$$\begin{aligned} Y_p(0) &= \frac{1}{2}\left(1 + erf\left(\frac{3}{\sqrt{2}}\right)\right) - \frac{1}{2}\left(1 + erf\left(\frac{-3}{\sqrt{2}}\right)\right) \\ &= 0.9973. \end{aligned} \tag{8}$$

3) Yield Impact Calculation: To assess the impact of guardband reduction on design yield, we track the change in the number good dies per wafer as we reduce the design guardband. To calculate the number of good dies per wafer, we first compute the gross number of dies per wafer as described in [23]

$$N_{\mathrm{gross}} = \pi\left(\frac{r^2}{A} - \frac{2r}{\sqrt{2A}}\right) \tag{9}$$

where A represents the die area which is fabricated on a wafer with radius r. In the above equation the second term accounts for wasted area around the edges of a circular wafer. Using (1) and (9), then the number of good dies per wafer is

$$N_{\mathrm{good}} = Y \cdot N_{\mathrm{gross}}. \tag{10}$$

There are two main scenarios for the guardband reduction.

1) We are able to improve the process so as to reduce the amount of guardbanding. This scenario corresponds to performing "iso-dense" timing analysis [10].
2) We simply apply a reduced guardband during the design process, even though the actual variability of the manufacturing process remains the same. This scenario corresponds to the $Y_p(RGB\%)$ calculation above.

Scenario (1) implies that Y_p remains at 0.9973, while overall yield increases because we benefit from decreased random defect yield loss due to decreased die area as well as from reduced die area itself. Table XV shows the number of good dies per wafer for each guardband reduction. For this analysis, we assume that a typical 65-nm SoC design that has 0.85 cm^2 die area and is composed of 0.48 cm^2 of standard logic cells and

[8]If chip area and a general defect density is used instead of critical area and specific defect density per critical area, then the complexity factor of Bose-Einstein equation is equal to 1 [16].

[9]Chip size is determined by the resulting standard cell area after synthesis. Due to the inherent noise of optimization, the chip size trend shows some glitches, e.g., at 40% guardband reduction.

[10]We understand that these assumptions are appropriate to current practice. Our discussion can be easily modified to use a different kσ window.

TABLE XV
NUMBER OF GOOD DIES PER WAFER FOR THE SCENARIO (1) GUARDBAND REDUCTION

RGB (%)	Expected area (cm^2)		Y_p (%)	Y_r (%)	Y (%)	#gross dies/wafer	#good dies/wafer
	Logic	Fixed					
0	0.480	0.370	99.7	82.5	82.3	759	624
10	0.449	0.370	99.7	83.0	82.8	789	653
20	0.438	0.370	99.7	83.2	83.0	801	665
30	0.430	0.370	99.7	83.3	83.1	809	672
40	0.417	0.370	99.7	83.6	83.3	823	686
50	0.408	0.370	99.7	83.7	83.5	833	695

TABLE XVI
NUMBER OF GOOD DIES PER WAFER FOR THE SCENARIO (2) GUARDBAND REDUCTION

RGB (%)	Expected area (cm^2)		Y_p (%)	Y_r (%)	Y (%)	#gross dies/wafer	#good dies/wafer
	Logic	Fixed					
0	0.480	0.370	99.7	82.5	82.3	759	624
10	0.449	0.370	99.3	83.0	82.4	789	651
20	0.438	0.370	98.4	83.2	81.9	801	656
30	0.430	0.370	96.4	83.3	80.3	809	650
40	0.417	0.370	92.8	83.6	77.5	823	638
50	0.408	0.370	86.6	83.7	72.5	833	604

0.37 cm^2 of fixed blocks, i.e., embedded SRAM, analog cores and IO cells. We use a 300 mm wafer diameter to calculate the number of dies, 0.2/cm^2 and 0.21/cm^2 defect density values (for logic cells and fixed blocks, respectively) to calculate random defect yield. Area reduction values are the average results from 65-nm testcases of our experiments.[11] Our use of an average area reduction is justified since all testcases across 90 and 65 nm show results that are monotone in guardband reduction value, and that have standard deviation (for any given guardband reduction value) of less than 5% (see Fig. 10). The table shows that 40% guardband reduction results in approximately 10% increase in the number of good dies.

Scenario (2), which is the focus of our discussion henceforth, changes $Y_p(RGB\%)$ as described above and is more pessimistic because no process improvement is assumed: the design guardband reduction increases random defect yield Y_r due to reduced die area, but this trades off against decreased Y_p.[12] Table XVI shows the number of good dies per wafer for each guardband reduction with the same assumptions used for Scenario (1). We observe that Y_p keeps decreasing as we reduce guardband, shown in Column 4 in the table, but we observed that decreased die area increases the number of good dies per wafer even without process enhancement.

Fig. 15 shows the change in the number of good dies per wafer over the guardband reduction for different defect clustering. From these plots, we can see that the number of good dies per wafer is maximized at around 20% of guardband reduction. In this figure, the assumption is that the entire design consists of logic cells. This trend will not be changed by the clustering of defects. Fig. 16 shows level curves of the number of good dies per wafer, plotted against guardband reduction (y-axis) and area reduction (x-axis). The dashed trace shows (area reduction, guardband reduction) points that we have realized experimentally. We see that the number of good dies increases by up to 4.1%, then starts to decrease, until the onset of yield degradation beyond 40% reduction in guardband.

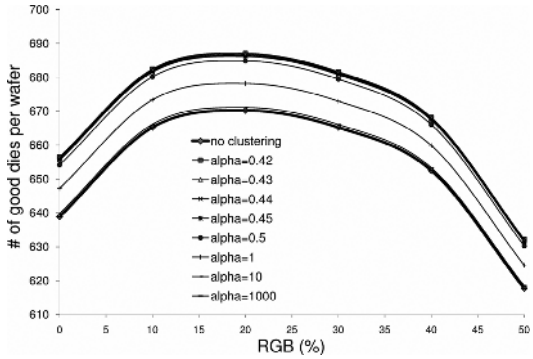

Fig. 15. Change in number of good dies per wafer, versus guardband reduction (%) and defect clustering.

We also estimate the impact of reduction of only the process guardband, since operating voltage and temperature can be fixed due to the design's requirements, as mentioned earlier in Section III-A. To calculate the area reduction from the process guardband reduction, we map the delay reduction percentages to the area reduction percentages from our experimental results on the logic area reduction in Fig. 10 and worst-case delay reduction in Fig. 1. Fig. 17 shows simple linear regression results on the area reduction versus guardband reduction. We then compute Y_p and Y_r. Fig. 18 shows the change in the number of good dies per wafer over the guardband reduction for two different assumptions: 1) with fixed blocks of which

[11]After guardband reduction we redesign (i.e., floorplanning), and if it does not result in chip area reduction, the random defect yield loss improvement will decrease.

[12]There is a third scenario, where the design floorplan is fixed so that standard-cell area reduction (due to reduced design guardbanding) does not result in any die area reduction. In this third scenario, wirelength reduction in the standard-cell blocks will result in lower metal density, which will reduce particle defect yield loss (since critical area is a function of wire density [13]). Hence, even when there is no change in die area with guardband reduction (e.g., with fixed-floorplan or pad-limited designs), we can expect a certain amount of Y_r improvement which increases the number of good dies per wafer. However, we do not currently have the tool infrastructure or foundry critical-area analysis decks needed to study this scenario.

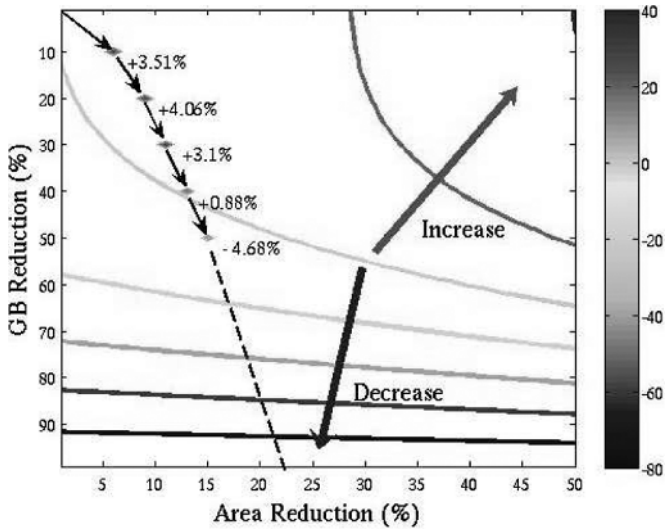

Fig. 16. Change (%) in number of good dies per wafer, versus guardband reduction (%) and area reduction (%).

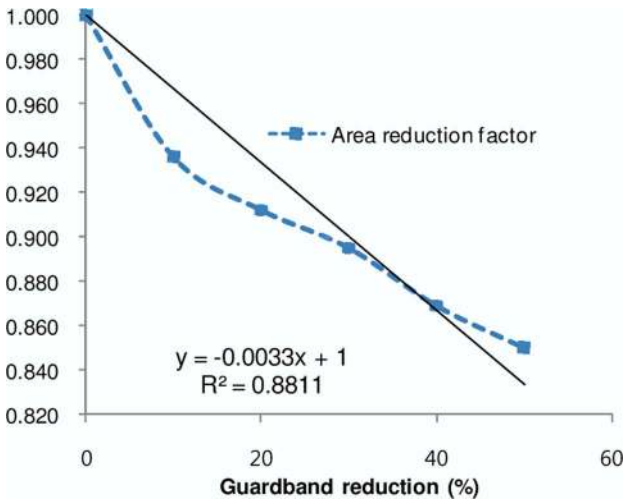

Fig. 17. Linear fit for area reduction (%) versus guardband reduction (%).

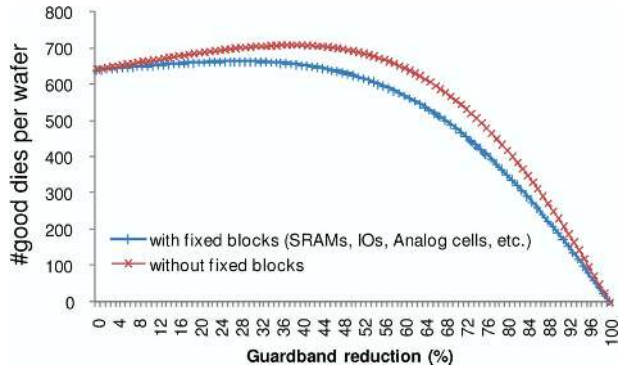

Fig. 18. Change in number of good dies per wafer only for the process guardband reduction (%).

size are not changed with guardband reduction and 2) without fixed blocks which implies that hard macros are newly designed corresponding to the guardband reduction or a design without hard macros. This plot reflects again a typical SOC in 90 and 65 nm, with die area 0.85 cm^2, and 0.48 cm^2 of the die being logic that is affected by the guardband reduction and 0.37 cm^2 of hard macros that may or may not be affected by guardband reduction. We observe that the number of good dies per wafer is maximized at around 24% process guardband reduction which results in 3.6% increase in the number of good dies per wafer,[13] even with over half of the design's area being fixed. The number of good dies per wafer can increase up to 10% at 38% process guardband reduction, if we redesign hard macros according to the guardband reduction or if a design is composed of pure logic cells.

VI. Conclusion and Future Directions

In this paper, we establish an experimental framework and then experimentally quantify the impact of model guardband reduction on outcomes of the synthesis, place and route (SP&R) implementation flow. We assess the impact of model guardband reduction on various metrics of design cycle time and design quality, using open-source and industrial embedded processor core with production 90-, 65-, and 45-nm technologies and libraries. We observe typical (i.e., average) reductions of 13%, 12%, 13%, and 19% in standard-cell area, total routed wirelength, dynamic power, and leakage power metrics from a 40% reduction in library model guardband (i.e., open-source testcases) and observe up to 8%, 7%, 5%, and 10% reductions in standard-cell area, total routed wirelength, dynamic power, and leakage power for the embedded processor core at 30% guardband reduction. We also observe 100% reduction in number of timing violations for a netlist that is synthesized with original library and extraction guardbands; this improvement can prove to be a significant factor in timing closure and design cycle turnaround time. Last, we quantify the impact of the guardband reduction on design yield. Our (Scenario 2 with fixed blocks) analysis shows up to 4% increase in the number of good dies per wafer with 27% guardband reduction. Interestingly, this increase in the number of good dies comes without any assumption of improved manufacturing capability (i.e., variability reduction). In addition, statistical analysis and optimization methodologies may not provide, by themselves, sufficient improvement of circuit metrics (e.g., [25] cites a 2% power reduction from statistical optimization; see also [26]). Therefore, our results suggest that there is justification for the design, EDA and process communities to enable guardband reduction as an economic incentive for manufacturing-friendly design practices.[14]

Our future work is in two directions: 1) to assess the impact of RGB on memory embedded designs and 2) to assess the feasibility of simultaneous guardband reduction and voltage lowering to find the best combination of guardband and supply voltage which optimizes for the area, yield, and power.

[13]4% increase in the number of good dies is significant. For example, if a design needs 50 K wafers to produce 30 M good units, and the cost per wafer is \$3 K, the 4% represents a reduction of 2 K wafers for the same number of good units, and the cost saving is \$6 M.

[14]As we have noted above: Although there exist clear decreasing trends in area and wirelength with respect to guardband reduction, due to the noise in the commercial tools, small guardband reductions (e.g., by $< 10\%$) may not always change flow outcomes as noticeably or consistently.

ACKNOWLEDGMENT

The authors would like to thank Dr. R. Radojcic and Mr. D. Lisk of Qualcomm CDMA Technologies, Inc., for providing the industrial testcase as well as for useful conversations that have strengthened the experimental results of our work.

REFERENCES

[1] Version 2006.06 Liberty User Guide, vol. 1.
[2] International Technology Roadmap for Semiconductors [Online]. Available: http://public.itrs.net/
[3] OPENCORES.ORG [Online]. Available: http://www.opencores.org/
[4] Qualcomm, Inc. [Online]. Available: http://www.qualcomm.com/
[5] EYES [Online]. Available: http://www.icyield.com/eyes.html/
[6] A. P. Balasinski, L. Karklin, and V. Axelrad, "Impact of subwavelength cd tolerance on device performance," *Proc. SPIE*, vol. 361, pp. 361–368, 2002.
[7] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing: For Digital, Memory and Mixed-Signal VLSI Circuits*. Boston, MA: Kluwer, 2000.
[8] K. Cao, S. Dobre, and J. Hu, "Standard cell characterization considering lithography induced variations," in *Proc. DAC*, 2006, pp. 801–804.
[9] M. Garg, A. Kumar, J. van Wingerden, and L. Le Cam, "Litho-driven layouts for reducing performance variability," in *Proc. ISCAS*, 2005, pp. 3551–3554.
[10] P. Gupta and F.-L Heng, "Toward a systematic-variation aware timing methodology," in *Proc. DAC*, 2004, pp. 321–326.
[11] P. Gupta, A. B. Kahng, S. Nakagawa, S. Shah, and P. Sharma, "Lithography simulation-based full-chip design analyses," *Proc. SPIE*, vol. 6156, pp. 61560T1–61560T8, 2006.
[12] P. Gupta, A. B. Kahng, Y. Kim, S. Shah, and D. Sylvester, "Modeling of non-uniform device geometries for post-lithography circuit analysis," *Proc. SPIE*, vol. 6156, pp. 61560U1–61560U10, 2006.
[13] H. T. Heineken and W. Maly, "Interconnect yield model for manufacturability prediction in synthesis of standard cell based designs," *Proc. ICCAD*, pp. 368–373, 1996.
[14] K. Jeong, A. B. Kahng, and K. Samadi, "Quantified impacts of guardband reduction on design process outcomes," in *Proc. ISQED*, 2008, pp. 890–897.
[15] A. B. Kahng and S. Mantik, "Measurement of inherent noise in EDA tools," in *Proc. ISQED*, 2002, pp. 206–211.
[16] R. Kumar, *Fabless Semiconductor Implementation*. New York: McGraw-Hill, 2008.
[17] S. Nassif, "Modeling and forecasting of manufacturing variations," in *Proc. IWSM*, 2000, pp. 2–10.
[18] R. C. Pack, V. Axelrad, A. Shibkov, V. V. Boksha, J. A. Huckabay, R. Salik, W. Staud, R. Wang, and W. D. Grobman, "Physical and timing verification of subwavelength-scale designs: I. lithography impact on MOSFETs," *Proc. SPIE*, vol. 51, pp. 51–62, 2003.
[19] L. Scheffer, "Why are timing estimates so uncertain? what could we do about this?," in *Workshop Notes*, vol. TAU-2002. [Online]. Available: http://www.lscheffer.com/Uncertain.pdf
[20] L. Scheffer, "An overview of on-chip interconnect variation," in *Proc. SLIP*, 2006, pp. 27–28.
[21] A. Shibkov and V. Axelrad, "Integrated simulation flow for self-consistent manufacturability and circuit performance evaluation," in *Proc. SISPAD*, 2005, pp. 127–130.
[22] D. Tsien, C. K. Wang, Y. Ran, P. Hurat, and N. Verghese, "Context-specific leakage and delay analysis of a 65nm standard cell library for lithography-induced variability," *Proc. SPIE*, vol. 6521, pp. 65210F–65210F, 2007.
[23] N. H. E. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*. Boston, MA: Addison-Wesley, 2005.
[24] J. Yang, L. Capodieci, and D. Sylvester, "Advanced timing analysis based on post-OPC extraction of critical dimensions," in *Proc. DAC*, 2005, pp. 359–364.
[25] S. M. Burns, M. Ketkar, N. Menezes, K. A. Bowman, J. W. Tschanz, and V. De, "Comparative analysis of conventional and statistical design techniques," in *Proc. DAC*, 2007, pp. 238–243.
[26] F. N. Najm, "On the need for statistical timing analysis," in *Proc. DAC*, 2005, pp. 764–765.
[27] D. Sylvester, O. S. Nakagawa, and C. Hu, "Modeling the impact of back-end process variation on circuit performance," in *Proc. VLSITSA*, 1999, pp. 58–61.

Kwangok Jeong (S'07) received the B.S. and M.S. degrees in electrical engineering from Hanyang University, Seoul, Korea, in 1997 and 1999, respectively. He is currently pursuing the Ph.D degree at the VLSICAD Laboratory, University of California at San Diego.

He worked at CAE Team, SoC R&D Center, Samsung Electronics, from 1999 to 2006. His research interests include physical design and VLSI design-manufacturing interface.

Mr. Jeong received three Best Paper Awards in 2001, 2002, and 2004, and an Honorable Outstanding Researcher Award in 2005.

Andrew B. Kahng (SM'07) received the A.B. degree in applied mathematics (physics) from Harvard College, Cambridge, MA, and the M.S. and Ph.D. degrees in computer science from the University of California at San Diego (UCSD), La Jolla.

He joined the UCLA Computer Science Department as an Assistant Professor in July 1989, and became Associate Professor in July 1994 and Full Professor in July 1998. In January 2001, he joined UCSD as Professor in the CSE and ECE Departments. He served as Associate Chair of the UCSD CSE Department from 2003 to 2004. In October 2004, he co-founded Blaze DFM, Inc. and served as CTO of the company until resuming his duties at UCSD in September 2006. He has published over 300 journal and conference papers. Since 1997, his research in IC design for manufacturability has pioneered methods for automated phase-shift mask layout, variability-aware analyses and optimizations, CMP fill synthesis, and parametric yield-driven, cost-driven methodologies for chip implementation.

Dr. Kahng was the founding General Chair of the 1997 ACM/IEEE International Symposium on Physical Design, co-founder of the ACM Workshop on System-Level Interconnect Prediction, and defined the physical design roadmap as a member of the Design Tools and Test technology working group (TWG) for the 1997, 1998 and 1999 renewals of the International Technology Roadmap for Semiconductors. From 2000 through 2003, he was Chair of both the U.S. Design Technology Working Group, and of the Design International Technology Working Group, and continues to serve as co-chair of the Design ITWG. He has also served as a member of the EDA Council's EDA 200X task force. He has been an executive committee member of the MARCO Gigascale Systems Research Center since its inception in 1998. He has received NSF Research Initiation and Young Investigator awards, 13 Best Paper nominations, and six Best Paper Awards.

Kambiz Samadi (S'01) received the B.S. degree in computer engineering from California State University, Fresno, in 2004, and the M.S. degree in electrical and computer engineering from the University of California, San Diego, in 2007. He is currently pursuing the Ph.D. degree in computer engineering.

His research interests include on-chip interconnection modeling for system-level design, 3-D IC integration, and VLSI design-manufacturing interface.

Mr. Samadi is a recipient of 2004 Honorable Mention Outstanding Undergraduate Award from the Computing Research Association (CRA).