

The Design and Validation of IP for DFM/DFY Assurance

Robert Aitken

ARM, 141 Caspian Court, Sunnyvale, CA, USA 94089

rob.aitken@arm.com

Abstract

Design for Manufacturability (DFM) is becoming increasingly important as process geometries shrink. The System-on-Chip business model requires high quality, high yielding IP. This paper shows how DFM and DFY are integrated as part of IP delivery, using a set of metrics to identify and fix yield limiters without compromising power, area or performance.

1 Background and motivation

Physical IP, which includes standard cells, memories, I/O, and analog/mixed signal blocks makes up a key part of the infrastructure of modern systems on chip (SOC). Collections of physical IP, known as libraries, provide the building blocks for creating large complex functions in both application-specific and semi-custom domains. Together with automated design tools, libraries are a fundamental part of today's design process.

These days, with the increasing importance of manufacturability, it is important that IP incorporates DFM as a fundamental part of the design process. In order to balance manufacturability with other design constraints, it is helpful to have metrics to quantify and compare designs.

Substantial previous work has been done in the area of DFM and standard cell libraries. Heineken et al proposed generating yield optimized cells and using them in synthesis [1]. This approach has since been refined by several others, e.g. [2][3]. DFM considerations form a key part of the standard cell methodology described by Nardi [3]. Alternative fabrics have also been proposed to simplify the problem by reducing the number of cell types [5]. Others have treated standard cells as essentially fixed in order to optimize other parameters such as interconnect cost; e.g. [6].

A common feature of much previous work is its assumption that objective yield metrics exist and can be applied to designs. Two issues confound this assumption and make it difficult to use in the IP delivery model. First, standard cells are designed early in a process cycle, when reliable yield information is not available, even with perfect information exchange (e.g. within an IDM or foundry). Second, the disaggregated nature of the semiconductor industry means that barriers exist to information exchange. Details on yield, process recipe, and variability can be

difficult to come by during design, and also changes over time. Recent tool advances are helping (e.g. encrypted process files for silicon simulation), but there is still significant work to be done.

A basic outline of the library development process is given in Figure 1. Key features to note are that significant design occurs before first silicon for a library, and also that significant volume manufacturing does not occur until late in the process. As a result, subtle yield effects must be deduced or estimated before they are observed in practice.

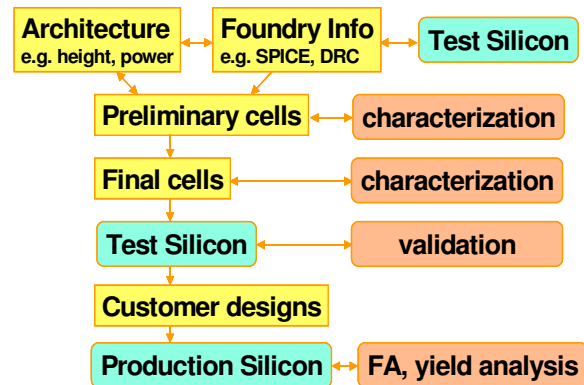


Figure 1. Simplified Standard Cell Library Development Process.

The remainder of the paper is organized as follows. Section 2 provides some background on DFM. Section 3 discusses the need for metrics and describes ideal metrics, together with issues involved in using them. Section 4 identifies some practical metrics. Section 5 shows an example of using these on a simple layout. Section 6 extends these ideas beyond standard cells. Finally, section 7 gives some conclusions and outlines future work in this area.

2 DFM Background

DFM and DFY (Design for Yield) mean many things, depending on the context. A good overview is available in Wong et al [10]. For the purposes of this paper, DFM will refer to layout design changes made to improve any aspect of manufacturability, from mask making through lithography and chemical-mechanical processing. DFY will refer to techniques specifically targeted to improving manufacturing yield.

The DFM challenge can be phrased simplistically as “Uniformity is good for manufacturability, but non-uniformity is the source of value in a design”. FPGAs solve this problem through programmability of regular structures. Memories repeat bit cells and other common layouts over and over. Standard cells push the bounds of regularity, but must retain enough uniformity to be manufacturable. Design rules enforce significant uniformity, but their binary nature (pass/fail) prevents them from accurately conveying the tradeoffs that exist in manufacturability.

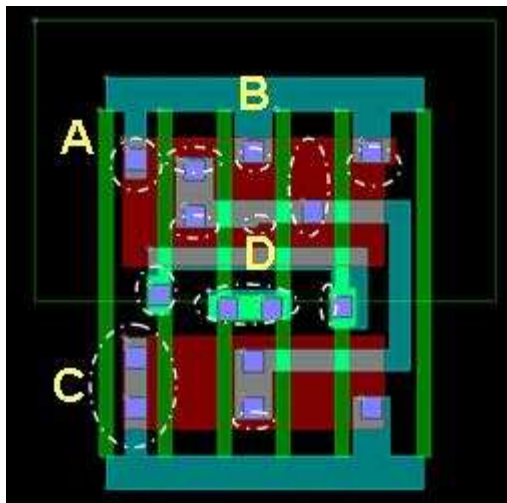


Figure 2 Pre-DFM Layout

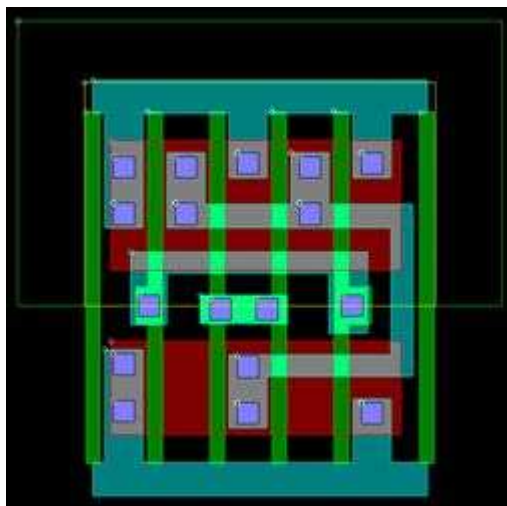


Figure 3 Post-DFM layout

In general, the changes involved in making a layout DFM compliant are subtle. The standard tradeoff applied is to make changes that do not increase the area of the cell. Consider, for example, the two layouts

shown in Figures 2 and 3. The single contact at A is made more manufacturable by doubling it and adding additional metal overlap. The contact at B cannot be doubled without increasing cell area. Instead, additional metal overlap is added instead. Contact C is already doubled, but additional overlap will help its manufacturability as well. D is an example of a small metal jog whose removal will simplify mask making and improve lithography. In each case, the effects on yield and manufacturability are minor, but will add up across a die.

None of the cases above required a change in area, and their performance impact is minor, so the changes can be implemented readily. On the other hand, making changes that increase area or lower performance cannot be done without accurate metrics to quantify the tradeoffs. The next section examines these.

3 The Need for Metrics

Each of the classic standard cell properties has associated metrics. Area is the simplest, and is usually measured in square microns. Timing was initially expressed as a simple delay; e.g. the propagation delay of the cell in nanoseconds. As technology advanced, this delay measurement became inadequate. First, delay was calculated for multiple process corners (slow, typical, and fast), to account for process spread. As wire delay has become a more important component of overall delay, more complex delay models have been developed, including equation based timing, non-linear delay tables, current source modeling and so on. It is not uncommon for an inverter today to have several hundred delay measurements associated with it.

Similarly, power metrics have evolved from a single number representing dynamic current to include leakage, load dependent power, state-dependent leakage and more.

These metrics share two important qualities: first, they move over time towards increasing accuracy, and second, there is an agreed upon procedure for calculating them. Both of these are needed in order to assure working silicon. The actual delay of a single standard cell is rarely measured for a chip. Instead, it is the cumulative delay effect of critical paths made up of these cells that matters. Similarly, the power associated with a given cell can be identified only with special hardware, but cumulative power consumption of all cells is what matters for battery life, heat dissipation, etc. If the low level metrics were unreliable, cumulative calculations would be difficult.

The process of calculating metrics for standard cells is known as characterization. In general, SPICE is used as the “gold standard” for characterization, since it has proven over time to describe cell behavior,

Property	Metric	Accuracy Needed	Accuracy Available	Implications
Area	Square microns	High	High	Self-evident to calculate. Chip area is a property of library routability
Power	Dynamic and leakage current	Medium	Medium to High. Highly dependent on SPICE model. Challenging for leakage due to process variation	Except in special cases, only the cumulative effect of thousands of cells is measurable, not individual values
Performance	Delay	High	High, but requires complex analysis at cell, block, and chip level. Dependence on variation not fully understood.	Performance depends on a small number of cells (typically 10-100) forming a set of critical and near-critical paths, so accuracy is vital.
<i>Manufacturability</i>	<i>Yield</i>	<i>Depends</i>	<i>Depends</i>	<i>Failure of a single cell can cause a chip to fail, but failure rates depend on factors that are difficult to characterize. Catastrophic failure can be predicted with some accuracy, but interacting failure modes are very difficult to model, let alone predict.</i>

Table 1. Standard Cell Metrics

especially delay, accurately enough to enable high volume manufacturing. Table 1 summarizes, for each of the important standard cell properties, the metric used to calculate it, the accuracy that successful manufacturing requires, the accuracy that is available in state of the art technology, and an “implications” column, which comments on the issues facing efforts to match accuracy to requirements.

The last row in Table 1 is for yield and manufacturability. The entries are vague because there is much that is not universally agreed upon for yield and standard cells, including a suitable metric (percentage, or net die per wafer; across lots or per lot, etc.), an objective means of calculating it, and even the data that would be used in a calculation. This paper attempts to catalog some of the challenges involved and make suggestions about what could be used.

3.1 The Ideal Case

Ideally a yield number could be associated with each standard cell and a synthesis tool could use this, along with power, performance, and area to produce a circuit optimal for the designer’s needs. Several efforts have been made in this area; e.g. [3], and it is certainly

a goal to strive for. Two observations are now offered for any DFM metric

Observation 1: a yield metric must be able to adapt to changing process conditions in order to be useful over time.

The major difficulty is that a manufacturing process is not static. As problems are found, they are fixed. Recipes are altered. Equipment ages or is replaced. Consider the following (somewhat contrived) example: Suppose two metal 1 layouts are available for a certain cell, shown in Figure 4. If the target process is highly susceptible to metal 1 shorts, the ideal layout will be the one on the left. On the other hand, if contacts are more of a problem, the ideal layout will be the one on the right. Suppose shorts are the biggest problem. Now suppose that after some time, process engineers identify the issue that causes the shorts. The ideal layout becomes that on the right. A single yield number associated with either would be inadequate. Recharacterization of the library might help a subsequent design, but it is too late for one that has already gone to silicon. Some process changes are predictable in advance (e.g. steady decline in defect density over a product lifetime), while others are not (e.g. changes caused by moving to a different equipment set).

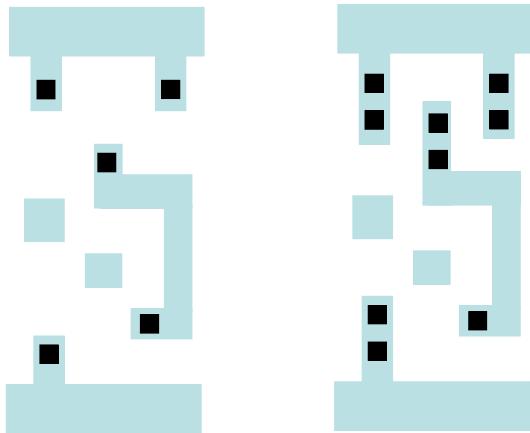


Figure 4. Sample layouts optimized for shorts and for contact failures

Observation 2: a yield metric must have an objective definition before it can be used for comparison.

A second issue is objectivity. Before embarking on a design, it is common for designers to evaluate several libraries and to select the one(s) that provide the best results for the target design. As we have seen, standard library metrics have objective definitions: given a SPICE model and set of assumptions (e.g. delay of a transition is the time taken for a signal to move from 10% of VDD to 90% of VDD), then the simulated delay of a cell should be the same, regardless of who calculates it. Yield, on the other hand, is inherently statistical. The yield of a cell depends on its surrounding context, including other cells, position on a die, position in a reticle, and position on a wafer. Exact values for each of these are unlikely, so approximations must be used, based on some test data. Different organizations may have different access to data, may have used different methods to collect it, and may have collected it at different times. Comparisons can be challenging. For example, critical area is a relatively objective metric, but to be converted to yield must include a failure rate, and this is subject to all the difficulties just mentioned.

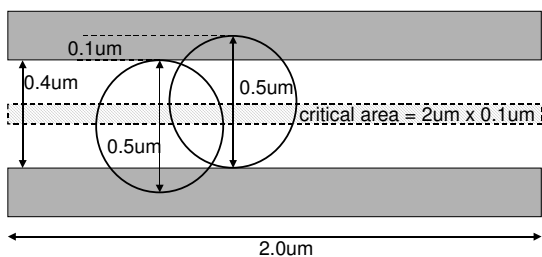


Figure 5. Calculating critical area

4 Example DFM Metrics

4.1 Critical area

Critical area is a common metric used to evaluate the susceptibility of a given layout to defects [7]. The critical area for shorts is the area, for a given radius particle, where a particle's center could land and cause a short in two wires. In Figure 5, the critical area for a 0.5-micron particle is 0.2 square microns, as shown.

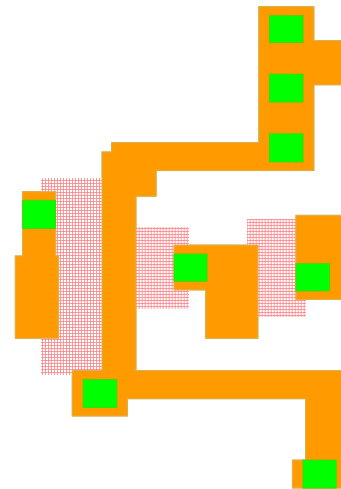


Figure 6. Layout 1, 3 input NOR

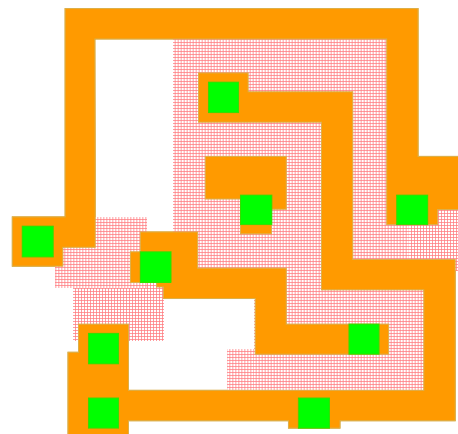


Figure 7. Layout 2, 3 input NOR

Critical area is a monotonically increasing function of particle size – bigger particles cause more defects. Particle size itself, though, is typically modeled as an inverse cube distribution, where bigger particles are less likely. Critical area is affected by layout

features such as layer density and complexity.

Figures 6 and 7 show two partial metal layouts for a three input NOR functionality. The critical area for shorts is much smaller for the layout of Figure 7 than that of Figure 6, showing that even for simple functions critical area is an important criterion.

Combining critical area with a particle distribution leads to weighted critical area. Alternatively, a single value can be chosen, such as the 50th percentile layer defect size (the size where half the expected defect particles are larger and half are smaller) and plot critical area. An example is given in Figure 8. Similar to other outlier-based analysis [8], the extreme cells at either end of the curve should be subjected to additional analysis – those with high critical area to improve yield, and those with low critical area to assess layout effectiveness.

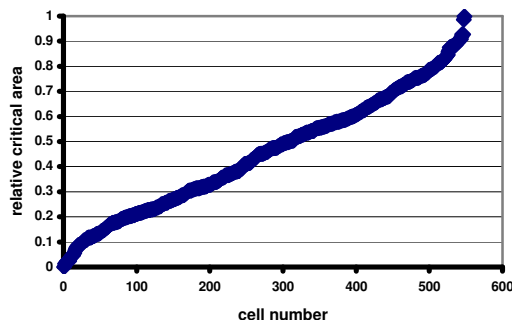


Figure 8. Relative critical areas for metal 1 shorts

4.2 Optical effects

IC manufacturing processes at current technology nodes make use of light with a wavelength of 248nm or 193nm for photolithography. This means that sub-wavelength features are the rule rather than the exception, and require resolution enhancement technology (RET) for successful printing. Two methods used to achieve this are optical proximity correction (OPC) and phase shifting masks (PSM). OPC works by modifying the aperture used for lithography, and PSM, as the name suggests, perturbs the phase of the light waves. In each case, mask artwork is modified so that the printed image is what the original layout designer desires.

Defining a metric for RET is complicated by several factors. First, recipes are foundry proprietary. Changes in yield and manufacturability give foundries an edge over their competitors, and so RET recipes are

jealously guarded. Data encryption or escrow organizations could both help with this issue. Second, OPC/PSM rules change frequently. Even if it were possible to update library artwork for every revision, this would disrupt user design flows and thus be unacceptable. Finally, there is a data volume issue: post-OPC layouts contain significantly more shapes and are thus significantly larger than pre-OPC data. For users already burdened by huge tapeout file sizes, including OPC information with a library would be unacceptable.

Still, OPC can be considered as part of library design. Certain structures are inherently vulnerable to optical effects and therefore allowance should be made in their design to ensure that subsequent OPC will be able to treat them correctly. An example is shortening of polysilicon “fingers” (Figure 9). The presence of a nearby structure can prevent OPC correction, resulting in an incorrectly printed object. In general, a cooperative relationship between IP vendors and foundries can ensure that foundry IP is protected while guaranteeing a manufacturable design. Data encryption may also help.

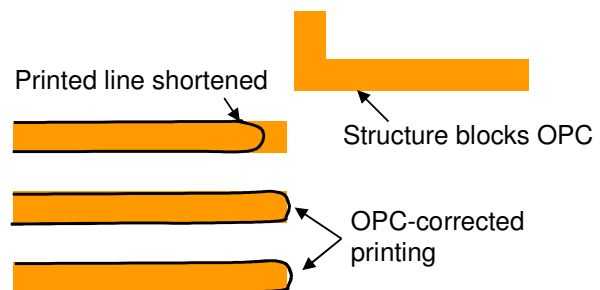


Figure 9. Layout influence on optical correction

5 Example DRC-based Metrics for DFM

Because DFM is complicated, foundries have developed special rules and recommendations for DRC. These fall into several major categories, including:

1. Improved printability. These include line end rules, regularity requirements, diffusion shape near gate rules, contact overlap rules, etc.
2. Reduced mask complexity. These include rules about “jogs”, or small changes in dimensions, structures which could confuse line end algorithms, and space needed for phase shift mask features
3. Reduced critical area. These include relaxed spacing, increased line thickness, etc.
4. CMP rules. These include density fill, as well as layer relationship rules.

Sometimes a rule serves multiple purposes, and sometimes the purposes conflict: increasing contact overlap also increases critical area for shorts. In order to allow numerical treatment of rules, a weighting approach is desirable. Each rule can be given a certain weight, and relative compliance can be scored. An example is given in the table below for four simplified rules in polysilicon and a simple scoring system (0% for meeting minimum value, 50% credit for an intermediate value, 100% for a recommended value; and the inverse of these values for “negative” rules, of which “avoid jogs” is an example – non complying structures subtract from the score). Note that the rule values are not meant to represent any actual process.

Rule	Weight	Scoring		
		0%	50%	100%
1. Increase line end	.4	0.05	0.1	0.15
2. <i>Avoid jogs in poly</i>	.3	Jog > 0.1	Jog > 0.05	Jog < 0.05
3. Reduce critical area for poly gates	.2	0.15 spacing	0.2 spacing	0.25 spacing
4. Maximize contact overlap	.1	0.05 on 2 sides	0.05 on 3 sides	0.05 on 4 sides

Table 2. Rules example

Figure 10 shows a sample layout, together with areas that comply with the minimum rule (e.g. 1B is a minimum line end, 4B is a minimal contact) and the recommended rule (1A is greater than the recommended value, 4A is an optimal contact). In scoring this cell, there are 6 line ends, two minimum (0%), 1 intermediate (50%), and 3 maximum (100%), for a total of 3.5 out of 6. There are 4 small jogs, for a score of -4. Gate spacing is scored at 2.5 out of 4 (two maximal, one intermediate), with contacts at 1 out of 3. Weighting these values gives a total of 0.8 out of a possible total of 3.5. Minor changes to the cell layout, as shown in Figure 11 increase the weighted total to 2.65, much closer to the ideal. None of these changes increased cell area. Improving some values further would require an area increase to avoid violating other rules. These are indicated by “C”. The results are summarized in Table 3.

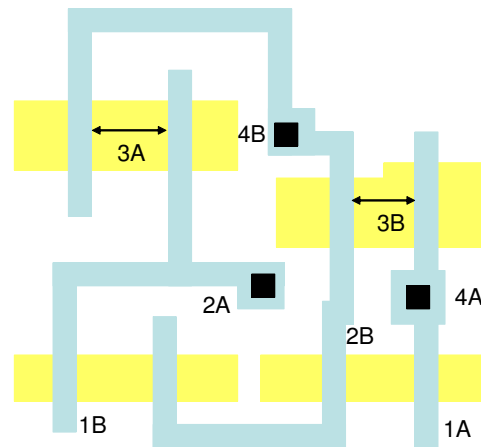


Figure 10. Example layout 1

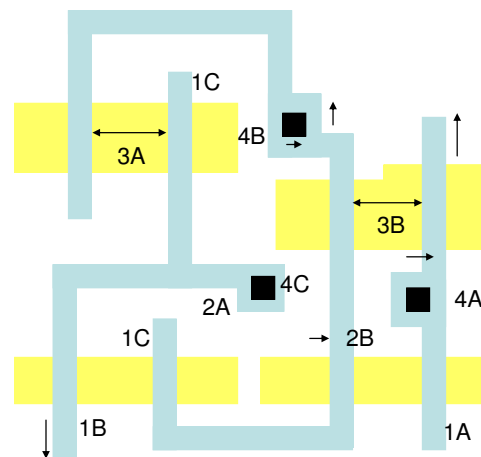


Figure 11. Example layout 2

Building a fractional compliance metric such as the one described here is a straightforward process using the scripting capability of modern DRC tools. Similar scripting methods have been shown in [9] to calculate critical area. The challenge is tuning it to give appropriate weight to various rules. Additionally, there are some DFM requirements that cannot readily be expressed in either rule or metric format, and these still require hand analysis.

	weight	layout 1		layout 2		ideal layout	
		raw	weighted	raw	weighted	raw	weighted
Rule 1	0.4	3.5	1.4	4.5	1.8	6	2.4
Rule 2	0.3	-4	-1.2	0	0	0	0
Rule 3	0.2	2.5	0.5	3	0.6	4	0.8
Rule 4	0.1	1	0.1	2.5	0.25	3	0.3
Total	1		0.8		2.65		3.5

Table 3. Metrics for example layouts

6 Extensions to Other IP

The discussion above concentrated mainly on standard cells, and while the DFM requirements for other IP are similar, there are some differences which will be outlined in this section.

6.1 I/O and Analog

Usually, I/O cells and analog blocks are able to accommodate all DFM recommendations, so the tradeoffs discussed above do not apply.

6.2 Memory

The basic methods shown in previous sections can be applied to many of the leaf cells in a memory. Bit cells are an exception, because they are usually optimized for manufacturability and yield by the foundry very early in the manufacturing process cycle. Some attributes of the bit cells force a level of “pitch matching” in other cells (e.g. sense amplifiers, word line drivers, prechargers etc.) that limit the degrees of freedom available for DFM optimization.

Memories also typically have redundancy available as an option to correct yield limiters. This changes some of the possible tradeoffs. As a simple example, suppose that the bit cells in a memory have an anticipated fail rate of X , and that these will be corrected by row redundancy. A DFM issue in a word line driver now has the option of being fixed by adding area to the driver cell, or by using the row redundancy. If the fail rate associated with this issue is a small fraction of X , then the cost of fixing it with redundancy is essentially zero. In general, critical area improvements can be traded off this way, while performance limitations cannot be, due to significantly higher expected failure rates.

6.3 Processors

If a processor is assembled from a set of optimized components (cells and memories), then a key portion of the work is already complete. In addition to the base work, applying robust design practices can improve manufacturability and yield by ensuring sufficient margin (thus allowing a certain level of immunity against minor timing variability and defects). For hard cores, careful attention to DFM practices during routing (via doubling, wire spreading, etc.) is also helpful.

7 Future work and Conclusions

We have tuned our metric function to reflect DFM requirements for 65nm and 90nm standard cell and memory IP across several processes. We are extending it to explicitly include lithography simulation in the

values. Also, the tools are currently run in batch mode as a post-processing step, rather than interactively during layout. As DFM tools mature, we expect improved flow and performance.

CMOS processes exhibit significant variability in their manufacturing parameters. Explicitly incorporating variability into the methods described in this paper presents further challenges.

8 References

- [1] H.T. Heineken, J. Khare, and M. d’Abreu, “Manufacturability Analysis of Standard Cell Libraries”, *Proc Custom Int. Circ. Conf*, pp. 321-324, May 1998.
- [2] C. Guardiani, N. Dragone, and P. McNamara, “Proactive Design for Manufacturing (DFM) for Nanometer SoC Designs”, *Proc Custom Int. Circ. Conf*, pp. 309-316, 2004.
- [3] A. Nardi and A. Sangiovanni-Vincentelli, “Synthesis for Manufacturability: a Sanity Check”, *Proc. Des. Aut. & Test Europe*, pp. 796-801, February 2004.
- [4] C. Bittlestone et al, “Architecting ASIC Libraries and Flows in Nanometer Era”, *Proc. Design Automation Conf.*, pp. 776-781, June 2003.
- [5] L. Pileggi et al, “Exploring Regular Fabrics to Optimize the Performance-Cost Trade-Off”, *Proc. Design Automation Conf.*, pp., 782-787, 2003.
- [6] P. Li, P. Nag, and W. Maly, “Cost Based Tradeoff Analysis of Standard Cell Designs”, *Proc. SLIP 2000*, pp. 129-135, 2000.
- [7] F. J. Ferguson and J. P. Shen, “Extraction and Simulation of Realistic Faults Using Inductive Fault Analysis”, *Proc. International Test Conference*, pp. 475-484, 1988.
- [8] R. Daasch et al, “Neighbor Selection for Variance Reduction in IDDQ and Other Parametric Data”, *Proc. International Test Conference*, pp. 92-100, 2001.
- [9] W. Pleskacz, C. Ouyang, and W. Maly, “A DRC Based Algorithm for Extraction of Critical Areas for Opens in Large VLSI Circuits”, *IEEE Trans. CAD*, Vol. 18, No. 2, Feb. 1999.
- [10] B.P. Wong et al, *Nano-CMOS Circuit and Physical Design*, Wiley 2005.