# Self-Compensating Design for Focus Variation

Puneet Gupta[†] , Andrew B. Kahng[†] , Youngmin Kim[‡] , Dennis Sylvester[‡]
[†] Blaze DFM Inc.,Sunnyvale CA, [‡] EECS Department, University of Michigan at Ann Arbor
[†] {puneet, abk}@blaze-dfm.com, [‡] {kimyz, dennis}@eecs.umich.edu

## ABSTRACT

Process variations have become a bottleneck for predictable and high-yielding IC design and fabrication. Linewidth variation (ΔL) due to defocus in a chip is largely systematic after the layout is completed, i.e., dense lines "smile" through focus while isolated (iso) lines "frown". In this paper, we propose a design flow that allows explicit compensation of focus variation, either within a cell (*self-compensated cells*) or across cells in a critical path (*self-compensated design*). Assuming that *iso* and *dense* variants are available for each library cell, we achieve designs that are more robust to focus variation. Design with a self-compensated cell library incurs ~11-12% area penalty while compensating for focus variation. Across-cell optimization with a mix of *dense* and *iso* cell variants incurs ~6-8% area overhead compared to the original cell library, while meeting timing constraints across a large range of focus variation (from 0 to 0.4um). A combination of original and iso cells provides an even better self-compensating design option, with only 1% area overhead. Circuit delay distributions are tighter with self-compensated cells and self-compensated design than with a conventional design methodology.

## Categories and Subject Descriptors

B.7.2 [**Design Aids**]: Layout, B.8.2. [**Performance and Reliability**]: Performance Analysis and Design Aids

**General Terms:** Algorithms, Performance, Design, Reliability

## Keywords

Variation, Layout, Focus, ACLV, Manufacturability, Compensation

## 1. INTRODUCTION

Within-die process variation has become one of the most important considerations in IC manufacturing, particularly as lithography moves into the deeply subwavelength regime. Variation can occur at the fabrication stage (intrinsic variation) or during circuit operation (dynamic variation) [1]. There are two major components to intrinsic variation: random and systematic [1],[2],[5]. Because of the strong layout dependency of the systematic component, estimation of systematic variation is impossible until layout information is available.

Effective channel length ($L_{eff}$) variation is one of the clearest determinants of IC performance [3]. Across chip linewidth variation (ACLV) control is critical to the timing and functionality of a design [4]. Various RETs (Resolution Enhancement Techniques) such as SRAF (Sub-Resolution Assistant Feature), OPC (Optical Proximity Correction), and PSM (Phase Shifting Mask) are commonly used to achieve this in current design-to-manufacturing flows [13],[14].

One of the major sources of $L_{eff}$ variation is focus. Such focus variations can occur, for example, due to changes in wafer flatness or lens imperfections. Traditional corner-case timing analysis flows are very pessimistic in worst-casing focus impact on critical dimensions. This is because layout pitch and focus have very systematic interactions, as shown by so-called Bossung plots (e.g., Figure 1). Recent work [6] notes that comprehending systematic through-pitch and through-focus variations in design can reduce timing uncertainty by up to 30%.


*DAC 2005*, June 13-17, 2005, Anaheim, California, USA.


## 2. COMPENSATING FOCUS VARIATION

Systematic variation can be mitigated to some extent by performing OPC and inserting assist features, but it cannot completely be removed for various reasons (modeling errors, algorithmic inaccuracies, etc.). The remaining linewidth variation due to layout is significant even after the use of complex RET techniques, with isolated and dense lines retaining opposite behavior under varying defocus [6]. Thus, there is a possibility of compensating for systematic variation in the design itself. This compensation can be achieved in two ways:

- *Self-compensated cell layout.* This is a correct-by-construction methodology that relies on within-cell compensation of focus variation. For example, variation can be compensated in series-connected NMOS, if one device becomes thinner (thus, faster) under defocus, and the other device becomes fatter (thus, slower). This can be achieved by making one device "iso" and the other device "dense".
- *Self-compensated physical design.* This refers to compensation across cells (e.g., in a critical path). Consider two cells G1 and G2 that lie on the critical path G1→G2. Focus variation, if not corrected can cause variation in delay of the critical path and potential timing failures or parametric yield loss. However, if G1 is explicitly made "iso" while G2 is made "dense", then focus variation can be compensated.

In this paper we compare and contrast the two approaches put forth above. We also study a hybrid flow that augments the original library (e.g., with insertion of iso-variant instances) to achieve design robustness. Section 3 describes the construction of a cell library that consists of each version of cells, and Section 4 describes self-compensating design. We present experimental results in Section 5, and Section 6 provides conclusions.

## 3. ISO/DENSE/SELF-COMPENSATED CELLS

### 3.1 CD Measurement

To analyze iso/dense/self-compensated behavior with defocus, we use a five-line pattern and sweep the space between the three center lines from 180nm to 480nm. SRAF (scattering bar) insertion and OPC are performed on these patterns using Calibre [7]. The average linewidth of the center line is then measured for each pattern.

Figure 1 shows the variation in this critical dimension (CD) for different space values at different defocus values. In our study 0.0um indicates in-focus and 0.4um is the worst-case defocus level.

Based on Figure 1, we generate a look-up table (LUT) using the function *CD = f(LS, RS, F)*, where LS is the left space, RS is the right space, and F is defocus. This allows us to obtain the exact degree to which specific patterns act isolated, dense, or self-compensated, and also to predict CD given defocus and spacings. The tolerance of the self-compensated devices is set at 4nm since the 3σ for the gate CD

control is 4nm in 130nm technology [8]. Thus, if linewidths are 4nm larger than nominal at 0.4um defocus, we assume those patterns are "dense"; similarly, if linewidths are 4nm smaller than nominal, we classify the patterns as "iso". Finally, if the CD variation is less than 4nm at 0.4um defocus, we consider the pattern "self-compensated". The first scattering bar insertion point is at a spacing of 420nm, therefore, the "most-iso" pattern has a spacing of roughly 400nm. At 420nm spacing and above, the pattern reverts to "dense" behavior as a result of scattering bar insertion. At the "most-dense" spacing (i.e., 180nm on each side), the linewidth increases 13% from nominal and in the "most-iso" case (i.e., 400nm on each side), it decreases 11% from nominal at the 0.4um defocus point.

The optimal scattering bar placement and width depend on numerous factors such as wavelength (λ), numerical aperture (NA), illumination type, and others [9],[10]. We use an optical model of 248nm wavelength and annular illumination with NA=0.7 lens.

## 3.2 Edge Devices

Special consideration is required for *edge devices*, i.e., devices that are closest to the cell boundary. We identify two different cases of edge devices: Case 1 has no neighboring devices on either side (e.g., INVX1), while Case 2 has no neighboring device on exactly one side (e.g., left-most or right-most devices in cells except INVX1 and INVX2). To investigate the edge effect in Case 1, we first sweep the spacing from 180nm to 1um symmetrically on both sides. For Case 2, we fix one side at 180nm or 380nm since most edge devices in TSMC 130nm standard cells have one of these two spaces on one side (i.e., without contact or with contact between poly lines). The spacing on the other side is swept up to 2um. Figure 2 shows linewidth vs. spacing in both Case 1 and 2. As can be seen from the left graph in Figure 2, linewidth is insensitive to focus after two SBs are inserted on each side of the poly line. The right graph in Figure 2 shows the Case 2 edge effect. When two adjacent poly lines are 1.2um apart (i.e., 2 SBs are inserted at each side), the linewidth does not vary much even if the spacing becomes larger. Since the distance from edge devices to the cell boundary for all cells is over 600nm in this technology, we assume that all edge devices in Case 2 follow the behavior seen in the plot.

## 3.3 Library Generation

The LUT that is constructed from Figure 1 gives CD and also the spacing between poly lines of iso/dense/self-compensated cells. Layout area is then estimated to quantify the area penalty and to extract the parasitic capacitance (i.e. AD, AS, PD, and PS) for these three versions of cells.

Figure 3 shows a representative layout of a large 2-input NAND gate in 130nm technology. Iso/dense/self-compensated devices co-exist in the original cell. Changing inter-device spacings allows us to generate iso/dense/self-compensated versions of cells. The area increase of iso versions of cells is 17% since they require more space to make all devices appear isolated. The area of self-compensated cells increases about 10% and there is a 3% average area increase in the dense versions.[1]

[1] Increased spacing can sometimes result in additional scattering bars, making devices behave dense.

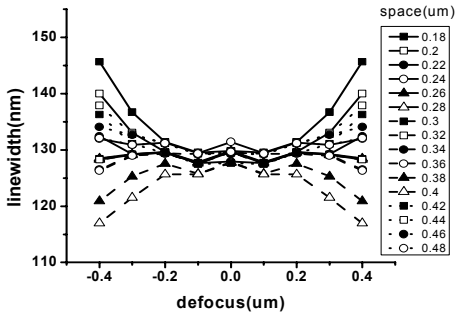


**Figure 1. Linewidth variation with defocus level (nominal linewidth = 130nm).**

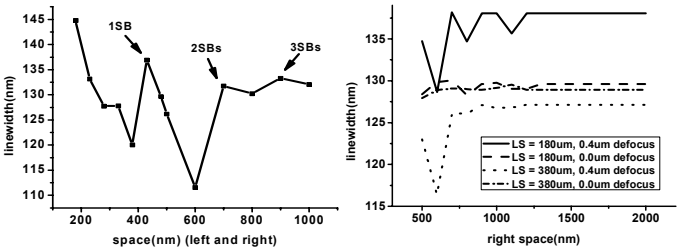


**Figure 2. Linewidth variation at 0.4um defocus in case 1(left), case 2(right). The arrows indicate scattering bar (SB) insertion points in the left plot.**

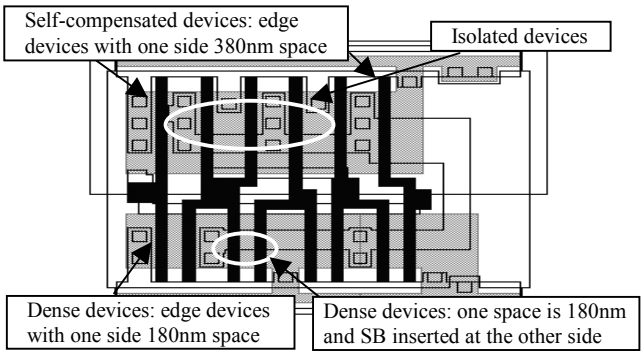


**Figure 3. Sample cell layout (NAND2X6) showing isolated, dense, and self-compensated devices.**

We consider 21 frequently used cells (INV: x1, x2, x6, x8, x12, NAND2(3) and NOR2(3): x1, x2, x4, x6). The LUT provides the exact isoness/denseness/self-compensatedness of devices at various spacings and two defocus points (0.0 and 0.4um). The CDs of edge devices follow the observations described in Section 3.2. Therefore we arrive at 4 different libraries that are characterized at both zero-focus and 0.4um defocus. HSPICE (using Autochar [11]) is employed to generate detailed .lib models.

# 4. SELF-COMPENSATING DESIGN

## 4.1 Self-Compensated Cells

Within a cell, self-compensated devices are constructed by modifying the cell layout to have explicitly iso and dense spacings. Starting from the original TSMC 0.13um standard-cell layouts, we generate new versions of cells that are "self-compensated" by adjusting the spacing between poly lines. As can be seen from Figure 1, the self-compensating space range is 260nm to 340nm each side. However, in our study self-compensation is achieved by having iso spacing on one side and dense spacing on the other. This allows for smaller cell layout areas with the same linewidth-focus behavior vs. using self-compensated spacings on both sides.

## 4.2 Optimization (self-compensated physical design)

### *4.2.1 Dense with Iso Design*

Another option is to generate optimized circuits using both *dense* and *iso* cells to meet timing at all focus points. This problem can be solved as a sizing problem. Since dense cells are slower (at worst-case focus) and smaller while iso cells are faster and bigger, we start with the circuit initially synthesized with dense cells, then swap in iso versions to meet timing at the worst-case defocus level.

Initially, synthesis with the "dense" library results in the slowest timing with small area. The optimization of delay versus area is implemented using a sensitivity-based approach to minimize area penalty while instantiating "iso" counterparts of "dense" cells in the circuit to meet timing constraints. In our experiments, the required time at the primary outputs is set to be 1% higher than the worst-case delay with the original library at 0.0 defocus. Because the original library is a mixture of iso, dense, and self-compensated devices, at 0.0um defocus level some cells show better timing in the original library than in their iso versions. The sensitivity of gates with respect to a change from "dense" to "iso" variants can be defined as [15]:

$$Sensitivity = \frac{1}{\Delta A + K_1} \sum_{arcs} \frac{\Delta D}{slack_{arc} - S_{\min} + K_2} \qquad (1)$$

where $\Delta A$ is the change in area and $\Delta D$ is the change in delay due to swapping "dense" with "iso". $S_{min}$ is the worst slack in the circuit when synthesized using the "dense" library, and the arcs consist of all rise and fall transitions from each input to output of the gate. The term $slack_{arc}$ is the difference between arrival and required times of the timing arc, and $K_1$ and $K_2$ are small positive numbers to ensure stability of the equation. Pseudocode for the first phase of our optimization process is as follows:

```
While worst_slack is negative
{
   Calculate sensitivities of all gates in the circuit
   Sort sensitivities in non-increasing order
   Swap the "dense" version with "iso" cell based on the
order of sensitivities
   Calculate new_delay of circuit
   Update Worst_slack
}
```

As the pseudocode indicates, we first sort sensitivities in non-increasing order. The gate with maximum sensitivity is then swapped with its corresponding *iso* version. Incremental timing analysis updates the *worst_slack* value and new sensitivities are then calculated if the timing is not met. Since all gates are *dense* at first, the design may not meet timing at worst-case defocus. Changing from *dense* to *iso* will compensate for the focus along critical paths. The process iterates the swapping until timing constraints are met.

Even after the above optimization procedure (which ensures timing correctness at both best and worst focus conditions), the circuit may not meet timing constraints at intermediate values of focus since delay variation with focus may be highly non-linear and even non-monotone. Thus, the timing constraint should be checked across defocus levels. If the extreme defocus point is out of the permissible focus range or the maximum delay is less than the required time, no more steps are needed. However, if the maximum-delay defocus point is within the permissible focus range, a post-processing step is required to globally meet the timing constraint. At the extreme defocus point, we can apply the same sensitivity-based optimization process shown above to ensure that the optimized circuit meets timing throughout the expected defocus range.

**Table 1. Average delay variation across benchmarks at 0.4um defocus with various libraries at 3 timing constraints.**

| Cell versions | Average Δ delay (%) | | |
|---|---|---|---|
| | Min_delay | 10% slower | 20%slower |
| **original** | 2.5 | 4.1 | 4.1 |
| **iso** | -8.9 | -5.6 | -6.2 |
| **dense** | 14.8 | 14.9 | 15.2 |
| **self-compensated** | 0.5 | 0.5 | 1.0 |

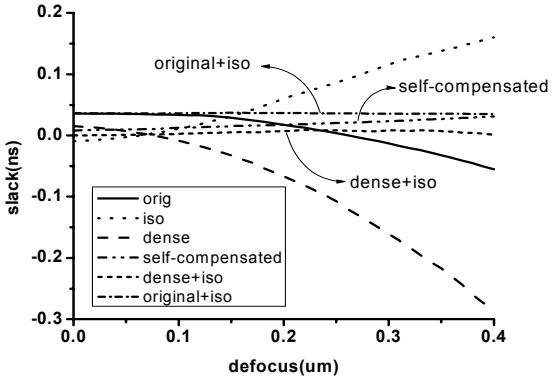


**Figure 4. Slack(ns) vs. defocus for testcase c3540 showing the effective compensation in self-compensating design options.**

### *4.2.2 Original + Iso Design*

The original library is a mixture of iso, dense, and self-compensated devices and the delay variation at 0.4um defocus of original cells is 2.5% to 4.1%. As a result, taking a design implemented with the original library and then optimizing it through the introduction of iso cells provides another option to reducing through-focus variability with small area overhead. This approach is particularly suitable for near-term technologies that already have libraries available – in this case users could choose to supplement this pre-existing library rather than creating two new (iso and dense) libraries. This latter option, corresponding to the strategy in Section 4.2.1 is more applicable in exploratory technologies such as 45nm today for which library design has not yet commenced. Starting from circuits synthesized with the original library at 0.0um defocus, "iso" cells are instantiated to meet timing at the worst case defocus level. The same sensitivity-based optimization process described in Section 4.2.1 can be applied.

If characterization or library generation effort is a concern, then the frequency count of iso gates actually used by the approach here enables us to filter the iso cells needed for effective optimization. In this way we find it is possible to achieve the timing constraint across all circuits studied with only a subset of all iso cells.

## 5. RESULTS

To quantify delay variation with defocus across the iso/dense/self-compensated libraries and using our optimization approaches, timing libraries for three different variants of each cell are generated as described in Section 3. ISCAS85 benchmark circuits are then synthesized at three different timing constraints (i.e. minimum, 10% and 20% slower than minimum timing) using Synopsys Design Compiler [12],[16].

Average delay variations of all benchmarks are shown in Table 1. As can be seen in Table 1, the dense cells at 0.4um defocus give 15% slower timing than the original library, and iso cells at 0.4um defocus give 5.6% to 9% faster timing than the original library at 0.0um defocus. The reduction of delay with the iso version at 0.4um defocus is slightly smaller than we might expect from the linewidth versus focus curves of Figure 1 because the iso cells have larger parasitics

which degrades the speed. The self-compensated cells in which the devices are modified to tolerate the defocus variation by canceling the iso-ness and dense-ness of the patterns shows minimum variation (less than 1%) at 0.4um defocus.

Looking more closely at how the various optimization choices affect timing across the entire defocus range of interest, Figure 4 shows the delay variation for c3540 from 0.0um to 0.4um defocus. As can be seen clearly from the curves, self-compensated cells, dense with iso optimization, and original with iso optimization all satisfy the timing requirement throughout the defocus range. The latter two approaches each benefit from the post-processing step that examines circuit delay at intermediate focus conditions – without this step, timing cannot be guaranteed through the defocus range.

Table 2 shows the area penalty of the self-compensating design options. We observe a 10 ~ 12% area penalty for a self-compensated cell based design compared to a 6 ~ 8% area penalty for the self-compensating design approach that uses a combination of dense and iso cell variants. Furthermore, the original + iso compensation scheme results in less than 1% area overhead while meeting timing.

In the dense and iso optimization of Section 4.2.1, approximately 32% of dense gates must be replaced to satisfy the timing constraint across defocus levels. However, in the original + iso option (Section 4.2.2), only 11% of the original instances needed to be replaced with their iso counterparts. These results clearly explain the very small area penalty seen in the original + iso optimization approach; only a small amount of the larger iso variants must be included.

Monte-Carlo simulation with 1000 trials is applied to investigate the impact of defocus variation on delay distribution. A normal distribution of focus with mean = 0.0um and $3\sigma$ = 0.4um is assumed. Figure 5 shows 1000 Monte-Carlo simulation results for the c3540 circuit. Self-compensated, dense + iso, and original + iso library options meet timing requirement at all randomly chosen defocus points In particular, the two optimization strategies suggested in Sections 4.2.1 and 4.2.2 demonstrate appreciably tighter distributions than the self-compensated cell-based approach.

## 6. CONCLUSION AND FUTURE WORK

A novel design technique to compensate for lithographic focus variation is proposed in this paper. Given self-compensated cells, which modify devices to cancel expected focus variation, designs that are much more robust to focus variation become possible. Compensated design for focus variation is achievable with small area penalties, assuming that dense and iso counterpart cells are also available. Since the original standard cells are a mixture of iso, dense, and self-compensated devices, we can also choose to add dedicated iso cells in order to meet timing at worst case defocus conditions. We observe that with dense and iso library options we can achieve a compensated design with 6-9% area overhead (compared to 10-12% in a self-compensated library based design) and by supplementing the original library with isolated variants, through-focus timing can be guaranteed with only 1% area penalty.

Our results are based on a 130nm technology – compensation in more advanced technologies such as 65nm is worth investigating as the impact is expected to be more. Also, our current sensitivity-based approach is used to optimize delay vs. area. While iso cells become faster under defocus conditions they also exhibit greatly enhanced leakage under defocus conditions since leakage grows exponentially when $L_{eff}$ becomes less than its nominal value. Therefore, joint optimization in the delay/area/leakage space is another compelling area of study.

**Table 2. Average area increase (%) from original layout with different design options.**

| design options | Min delay | 10% slower | 20% slower |
|---|---|---|---|
| dense | 2.93 | 2.36 | 3.14 |
| iso | 19.82 | 17.59 | 17.8 |
| self-compensated | 12.05 | 11.48 | 10.74 |
| optimization(dense + iso) | 6.06 | 6.73 | 7.65 |
| optimization (original + iso) | 0.38 | 0.74 | 0.73 |

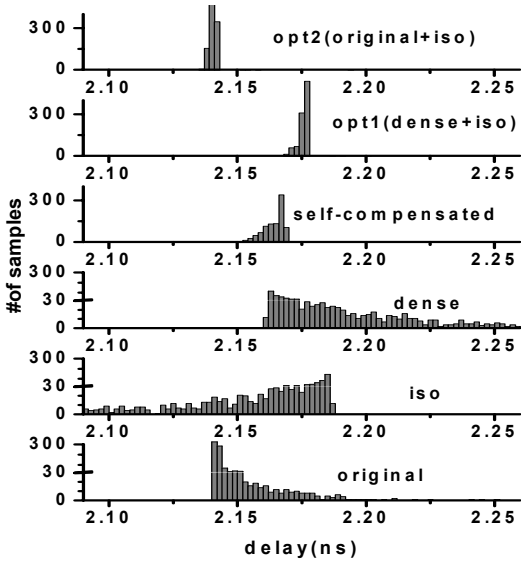


**Figure 5. Stacked histogram showing the distribution of delay for c3540 (req_time = 2.177ns), note that there are breaks in y-axis at the bottom three plots**

## 7. REFERENCES


[1] Y. Cao, *et al.*, "Design Sensitivities to Variability: Extrapolations and Assessments in Nanometer VLSI", *Proc. ASIC/SOC,* 2002, pp. 411-415.

[2] S. R. Nassif, "Design for Variability in DSM Technologies", *Proc. ISQED* 2000, pp. 451-454.

[3] S. R. Nassif, "Within-Chip Variability Analysis", *Proc. IEDM,* 1998, pp. 283-286.

[4] M. Orshansky, L. Milor, P. Chen, K. Keutzer, C. Hu, "Impact of Systematic Spatial Intra-Chip Gate Length Variability on Performance of High-Speed Digital Circuits", *ICCAD*, 2000, pp. 62-67.

[5] P. Gupta and A. B. Kahng, "Manufacturing-Aware Physical Design", *ICCAD,* 2003, pp. 681–687.

[6] P. Gupta and H. Fook-Luen, "Toward a systematic-variation aware timing methodology," *Proc. DAC*, 2004, pp. 321-326.

[7] Calibre version 2004.1_7.33, *http://www.mentor.com*.

[8] "International Technology Roadmap for Semiconductors 2003," *http://public.itrs.net/Files/2003ITRS/Home2003.htm.*

[9] T. Yorick, *et al*, "ArF imaging with off-axis illumination and subresolution assist bars: a compromise between mask constraints and lithographic process constraints," *Proc. SPIE*, 2002, vol. 4691, pp. 1522-1529.

[10] A. J. Lori, T. R. Michael, D. Jason, and J. Christiane, "Effect of scattering bar assist features in 193-nm lithography," *Proc. SPIE*, 2002, vol. 4691, pp. 861-870.

[11] Autochar - Automates the characterization of digital circuits, *http://directory.fsf.org/design/cad/autochar.html*.

[12] Design Compiler version V-2003.12, *http://www.synopsys.com*.

[13] A. B. Kahng and Y. C. Pati, "Subwavelength Lithography and its Potential Impact on Design and EDA", *Proc. DAC*, 1999, pp. 799-804.

[14] L. W. Liebmann, S. M. Mansfield, A. K. Wong, M. A. Lavin, W. C. Leipold, T.G. Dunham, "TCAD Development for Lithography Resolution Enhancement", *IBM J. RES. & DEV*, vol. 45, no. 5, 2001.

[15] S. Sirichotiyakul, *et al.*, "Stand-by power minimization through simultaneous threshold voltage selection and circuit sizing" *Proc. DAC*, 1999, pp. 436-441.

[16] F. Brglez and H. Fujiwara, "A neutral netlist of 10 combinational benchmark circuits and a target translator in Fortran", *Proc. ISCAS*, May 1989, pp. 695-698.