Functionality and performance improvements with Field-based OPC

Ben Painter, Kunal Taravade, Levi Barnes, Robert Lugg, Jeff Mayhew,
Glenn Newell*, Kevin Lucas**

Synopsys Inc., 2025 NW Cornelius Pass Rd., Hillsboro, OR 97124 USA
*Synopsys Inc., 700 East Middlefield Rd., Mountain View, CA 94043 USA
**Synopsys Inc., 1301 S. Mopac Expressway, Austin, TX 78746 USA

ABSTRACT

The upcoming 45nm and 32nm device generations will continue the familiar industry lithography trends of decreased production K1 factor, reduced focus error tolerances and increased pattern density. As previous experience has shown, small changes in the values of lithographic K1, focus tolerance and pattern density for the process-design space can lead to large required changes in OPC and RET solutions. Therefore, significant improvements in utility and speed are needed for these new device generations. In this paper we highlight significant new functionality and performance capabilities using existing Field-based OPC and RET methods. The use of dense grid calculations in Field-based methods is shown to provide a software platform for robust and fast implementation of new model-based RET techniques such as model-based assist feature placement and tuning. We present the performance and capability increases for model-based RET methods. Additionally, we have studied and present the performance of production 45nm generation field-based OPC and RET software across several different multiple-purpose hardware platforms. Significant improvements in runtime (for approximately the same hardware cost) are observed with new general purpose hardware platforms and with software optimization for this hardware.

Keywords: OPC, RET, Field-based OPC, model-based assist features, OPC hardware.

1. Introduction

The workhorse of model-based OPC is so-called "flash-based" simulation based on an efficient sparse convolution algorithm [1]. In OPC, the mask layout polygons are normally "dissected" into small movable segments, which are simulated at one or more image evaluation points to predict the error between the expected wafer pattern and the target (input) wafer pattern for that segment (i.e., the edge error). The simulation convolves spatial kernel functions with the physical mask layout to calculate the aerial image on the wafer at the segment. The aerial image is then translated into a prediction for the final wafer pattern edge. The segment location is then adjusted based on the error amount between the design target and the predicted pattern edge at the image evaluation point(s). All design segments are simulated and adjusted in local and global iterations to optimize the mask layout to converge upon the desired wafer pattern.

Field-based OPC [2] leverages many of the same methods of flash-based OPC. The fundamental difference is that field-based computes, in one operation, a uniform grid of simulation points, or pixels, for an entire working region (typically the region is a square area of between 40um^2 and 100um^2). The grid spacing is determined by the spatial bandwidth of the model. Applying the Nyquist criterion to establish grid spacing (about 20-25nm for 45nm node), the sample array completely describes the band-limited system, and any off-grid point can be accurately calculated with appropriate interpolation methods.

In flash-based OPC, each model evaluation point calls a flash-based simulation; and in field-based OPC, each model evaluation point interpolates the result from the pre-computed array of simulation pixels. The fundamental computation cost differences between the two methods is illustrated in figure 1a. With field-based, there is an up-front computational load for generating the simulation pixel array, but interpolation

Photomask and Next-Generation Lithography Mask Technology XIV, edited by Hidehiro Watanabe,
Proc. of SPIE Vol. 6607, 66072T, (2007) · 0277-786X/07/$18 · doi: 10.1117/12.729012

cost for each evaluation point is relatively low. The computational cost for flash-based is proportional to the number of evaluation points needed. There is thus a performance crossover point where field-based is more efficient for dense sample requirements and, conversely, flash-based is more efficient for sparse requirements. Figures 1b and 1c illustrate two other performance crossover parameters: complexity of the simulated pattern (number of defining vertices), and model ambit, respectively. The performance crossover point is at approximately the 45nm logic node, see figure 2 for results for a large # of 45nm layouts.

2. Platform and Software Comparisons

One important calculation type performed in field-based OPC is the FFT operation. In this section we compare FFT execution times on a variety of different hardware and software platforms in order to determine the best option. All comparisons are done on a price-normalized basis, comparing different systems of the same price, as obtained from a typical server OEM vendor. Configurations for the different systems, as well as the references from which these numbers were obtained, are as shown in the accompanying table.

We have graphed the time it takes to complete FFT operations on the various platforms normalized to the time taken to execute FFT in our OPC engine, Proteus®, on a Xeon® 5160 system with 16GB of RAM, see figure 3. FFT sizes are different for the different platforms shown, but each is compared to the FFT of the same size on the Proteus code to enable accurate final comparisons. The graph shows that the best price-performance is obtained on the Proteus FFT running on a Xeon™ 5160 (i.e., Woodcrest). The Proteus code has been optimized for the general purpose x86 architecture. This strategy allows one to extract the best cost of ownership from such a deployment, using the same hardware platform for both specialized operations such as the FFT, as well as for other routine tasks during OPC execution.

Details of the configurations compared:

1. 2-socket Dual Core Xeon® 5160 (Woodcrest) with 16GB RAM; Various FFT sizes
2. Cell BE Accelerator Card with 5GB RAM plugged into a single socket Xeon 5160 board with 10 GB RAM; FFT size = 4Kx4K [3]
3. Virtex-5 FPGA card loaded with Xilinx FFT (Radix-4) plugged into a 2-socket Dual Core Xeon® 5160 board with 16GB RAM; Reference:- Xilinx DS260 Datasheet, Feb 15, 2007, FFT size = 2K (1D FFT)
4. Clearspeed card (2 CPUs) plugged into a 2-socket Dual Core Xeon® 5160 board with 16GB RAM; FFT = 2Kx2K

Other variables:

1. Utilization of compute resources for custom vs. general purpose hardware
2. Power costs of custom vs. general purpose hardware

3. Hardware Comparisons

We tested the turn around time (TAT) performance of Proteus OPC on several general purpose hardware platforms using flash-based OPC for different 65nm and 45nm testcases. Sample results are shown in figures 4 and 5.

The following configurations were tested:
DualCore 3.0GHz Xeon® 5160 (Woodcrest) system
DualCore 2.4GHz Opteron® Rev E system
DualCore 2.4GHz Opteron® Rev F system
SingleCore 2.6GHz Opteron® system
SingleCore 2.8GHz Opteron® system
SingleCore 3.4GHz Xeon® system

Flash-based Proteus showed a 1.4X to 2X TAT reduction on Woodcrest systems vs. the Opteron® systems depending on vertex density and test case profile.

We also tested the performance of Proteus OPC on the different general purpose hardware platforms for field-based and flash-based OPC for a 45nm testcase. The results for TAT are shown in figure 6. This was a testcase where field-based OPC was faster than flash-based OPC (see section #1 for more discussion). Figure 6 shows that the relative flash vs. field-based TAT is approximately equal across several different general purpose platforms. Figure 6 also clearly shows that the significant TAT benefits of the DualCore Woodcrest hardware are applicable to both field and flash-based OPC. In addition, recently released QuadCore general purpose platforms should provide further significant price performance benefits.

4. Model-based Sub-Resolution Assist Features

Figure 7 shows a sample via layout which is difficult for rule-based assist feature (AF) placement schemes and would benefit from model-based AF placement. Model-based AF placement is particularly well suited to field-based modeling. To place AFs, a model is created which defines the benefit of placing an AF at any given position on the mask. Figure 8 shows an example of model output. In this figure, blue and green (darker) areas represent beneficial locations, while red and orange (lighter) areas are less beneficial or even detrimental. The task, then, is to place assist features on the mask in such a way as to maximize benefit, as defined by this model, while obeying mask and printability constraints. This requires evaluating the model at a large number of points, both in order to determine proper initial placement and to make decisions about AF orientations and relations to neighboring AFs. Figures 9 through 12 show examples of the model results and final AF placements that might be found from different imaging processes.

In a typical 50 micron square section of the pattern, the model is sampled more than 5 million times in the process of placing the assist features. All of these samples occur on a single, static image; that is, no perturbations need to be made to the mask. This type of evaluation is well-suited to field-based model evaluations. The relative benefit of field-based over flash-based model-based AF processing is further improved by the fact that there is an increasing # of vertices within the ambit of the model at smaller process nodes. This makes flash-based model evaluations more expensive, while field-based model evaluations are not significantly affected by pattern density. These effects combine to produce advantages of up to 30x in processing time (see Figure 13) for model-based assist feature placement using field models versus flash models.

5. Summary

In this poster we have highlighted significant new functionality and performance capabilities using existing Field-based OPC and RET methods. We have presented the performance of production 45nm generation field-based OPC and RET software across several different hardware platforms. Significant improvements in runtime (for approximately the same hardware cost) are observed with new general purpose hardware platforms and with our software optimization for the general purpose hardware. The use of dense grid calculations in Field-based methods offers a solid software platform for robust and fast implementation of new model-based RET techniques such as model-based assist feature placement and tuning. We presented significant performance and capability increases for model-based RET using field-based methods.

References:

[1] Fast proximity correction with zone sampling. John P. Stirniman, Michael L. Rieger.. Proc. of SPIE Vol. 2197, 1994.
[2] Migrating from Traditional OPC to Field-based OPC for 45nm Node Production. Rick Farnbach, Josh Tuttle, et. al. Proc. of SPIE Vol. 6520, 2007.
[3] Potential of the Cell Processor for Scientific Computing. Williams, et. al. Proceedings of the 3rd conference on Computing frontiers, pp 9-20, 2006.

Figure 1a: Relative Flash to Field-based computation as a function of model evaluation density.

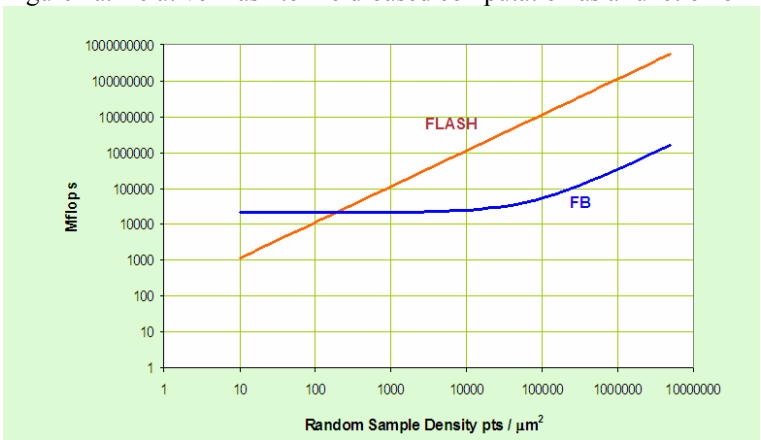
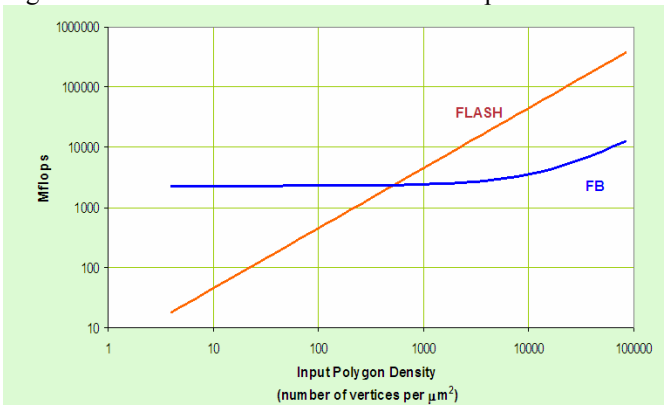


Figure 1b: Relative Flash to Field-based computation as a function of input layout complexity



.Figure 1c: Relative Flash to Field-based computation as a function of model ambit.

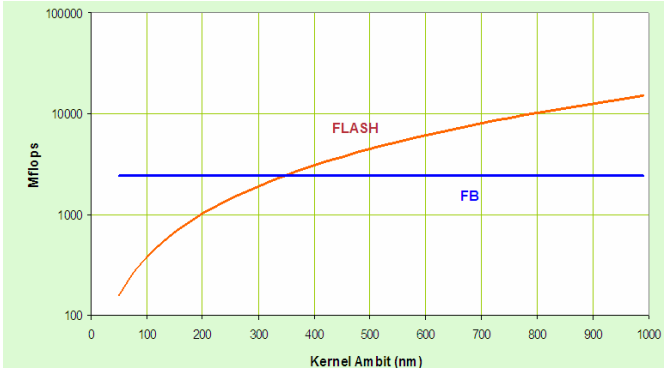
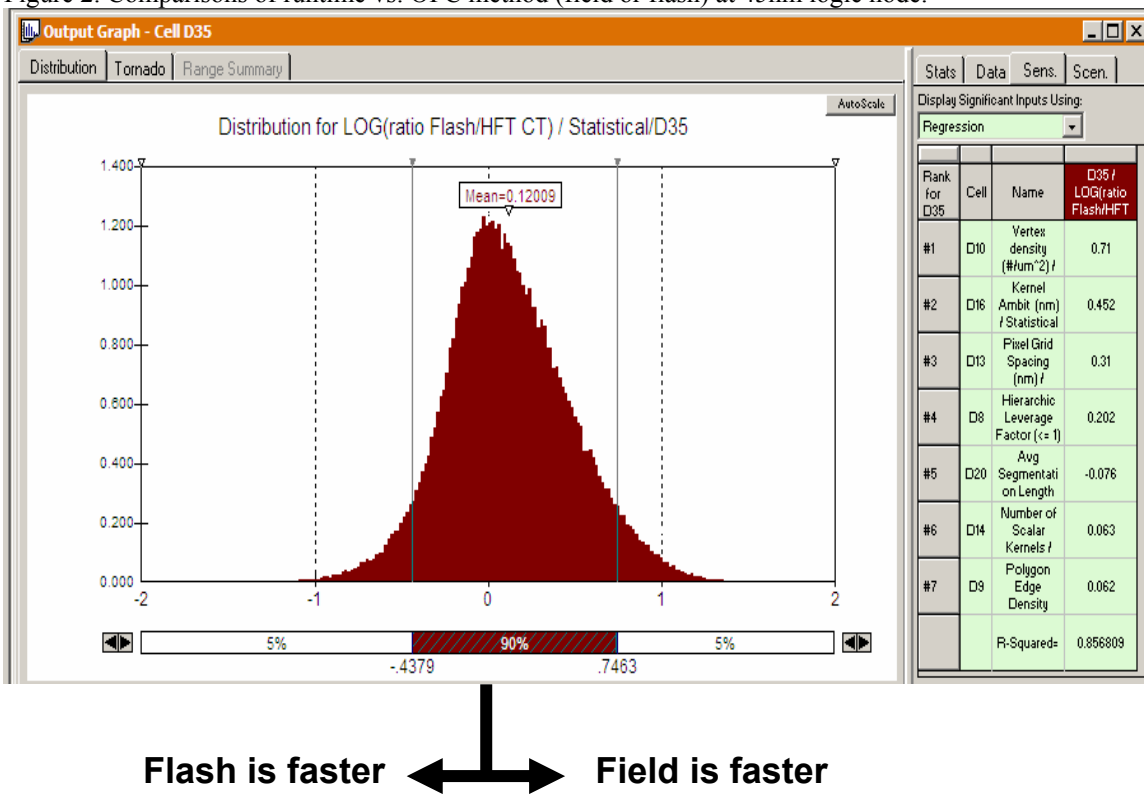


Figure 2. Comparisons of runtime vs. OPC method (field or flash) at 45nm logic node.



Flash is faster ⟷ **Field is faster**

Figure 3: FFT calculation time normalized by price for different hardware solutions. The Proteus FFT on the Woodcrest general purpose hardware is seen to provide the best price/performance ratio.

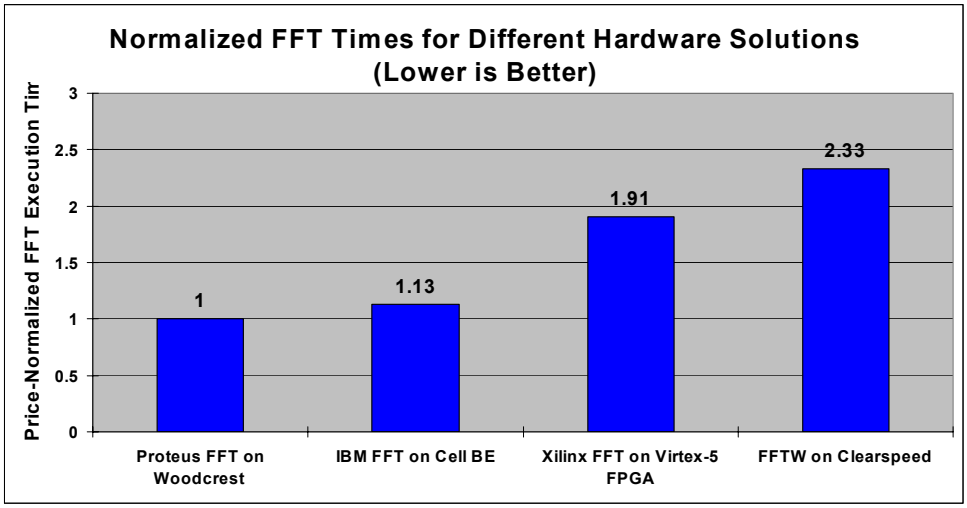


Figure 4. Proteus turnaround time as a function of general purpose hardware platform for OPC case #1. The Woodcrest platform can be seen to offer significant performance benefit vs. other options.

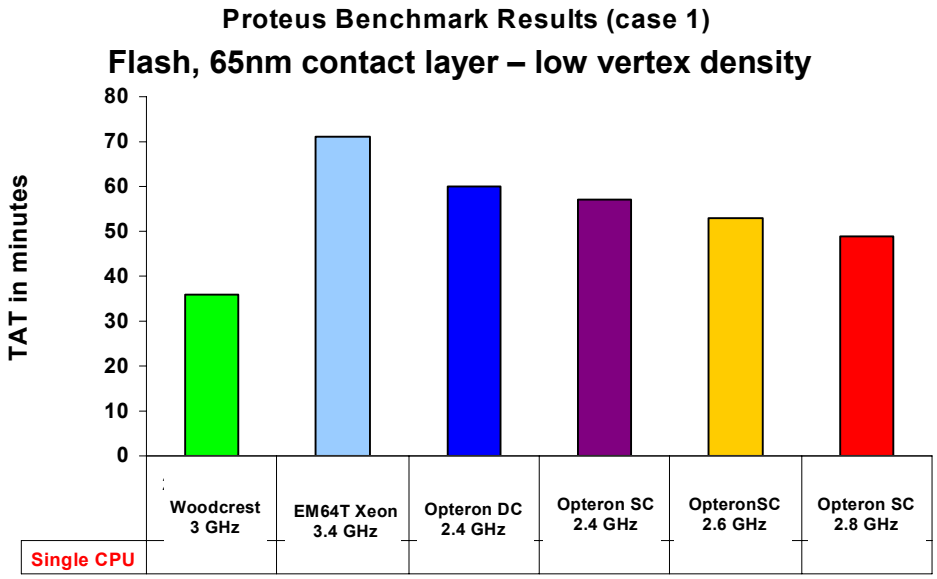


Figure 5. Proteus turnaround time as a function of general purpose hardware platform for OPC case #2. The Woodcrest platform can be seen to offer significant performance benefit vs. other options.

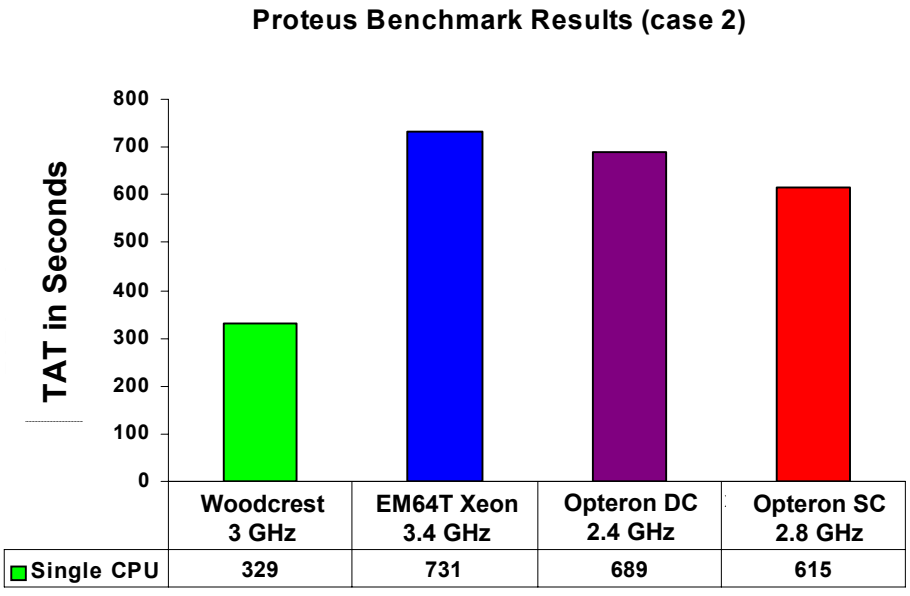


Figure 6. Proteus OPC turnaround time as a function of hardware platform. The Woodcrest platform can be seen to offer significant performance benefit vs. other options for both field and flash.

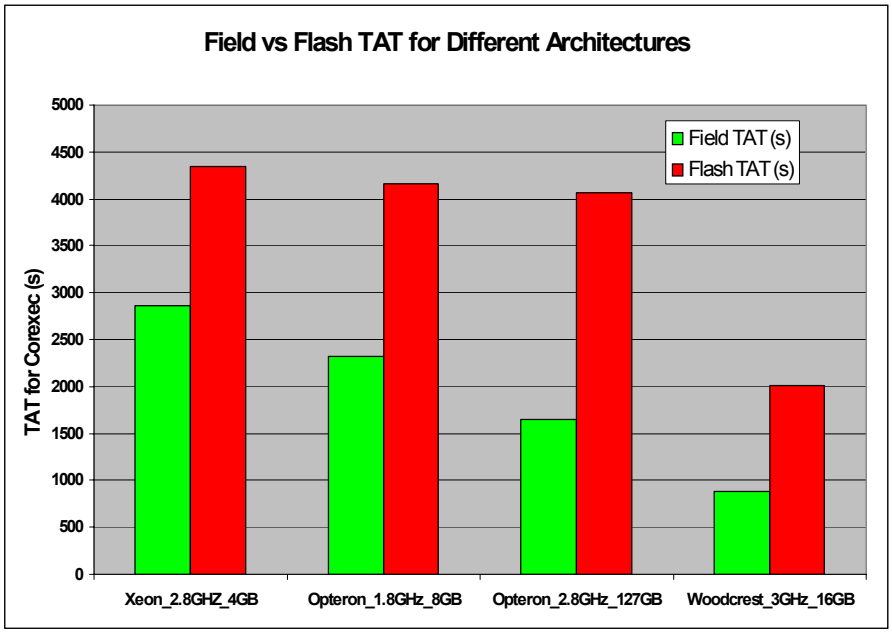


Figure 7. Example random logic via layout which is difficult for rule-based assist feature method to accurately place assist features in.

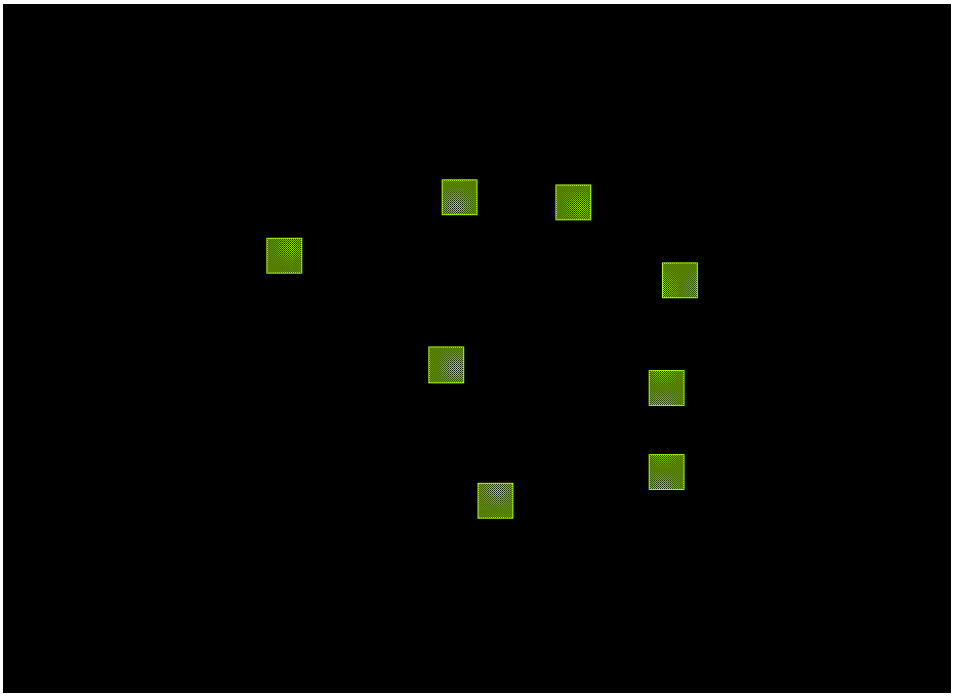


Figure 8. Model-based assist feature placement example showing the simulation of image intensity on the wafer for layout in figure 8. The ideal assist feature placement is seen at the high intensity sidelobes.

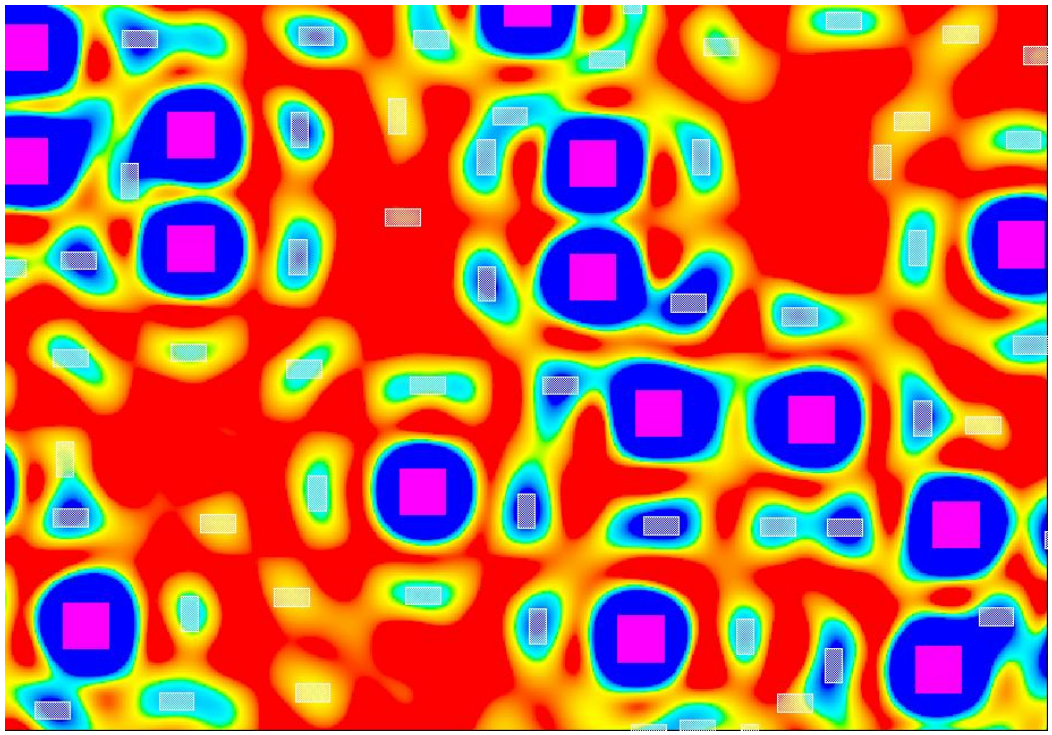


Figure 9. Simulation of image intensity on the wafer for the via layout in figure 7 assuming annular illumination. The ideal placement of assist features is at the darker spots and rings around the vias.

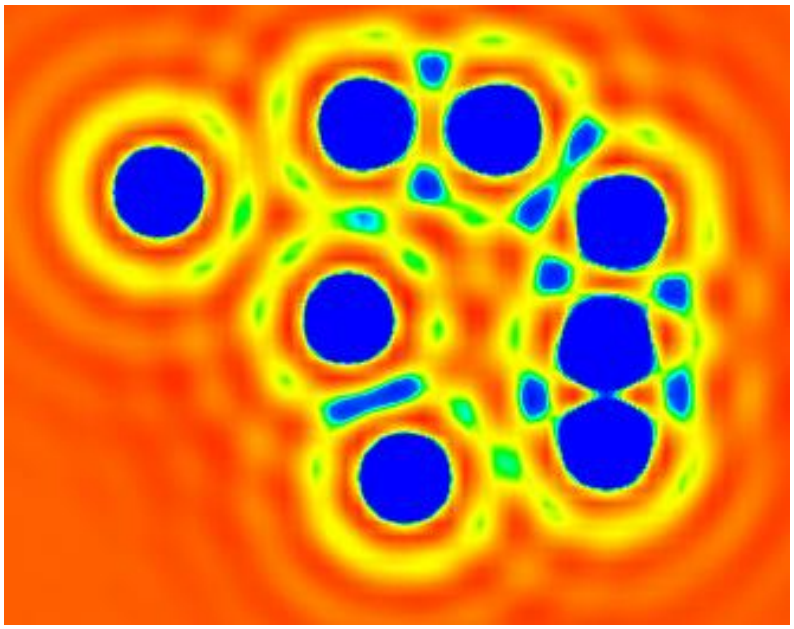


Figure 10. Output of model-based assist feature placement method for the annular illumination aerial image in figure 9. In this example, the assist features are limited to Manhattan geometries.

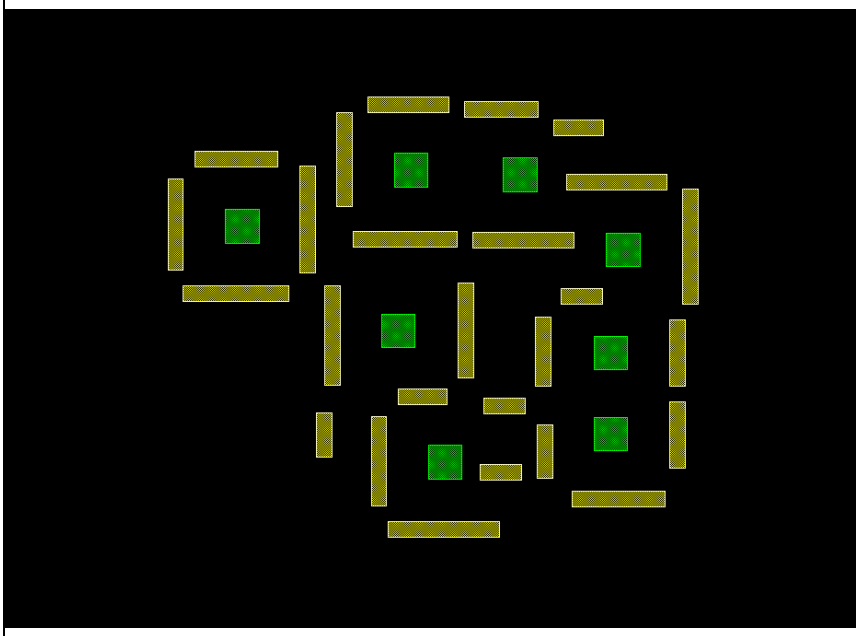


Figure 11. Simulation of image intensity on the wafer for the via layout in figure 7 assuming quadrupole illumination. The ideal placement of assist features is at the darker spots and lines outside the vias.

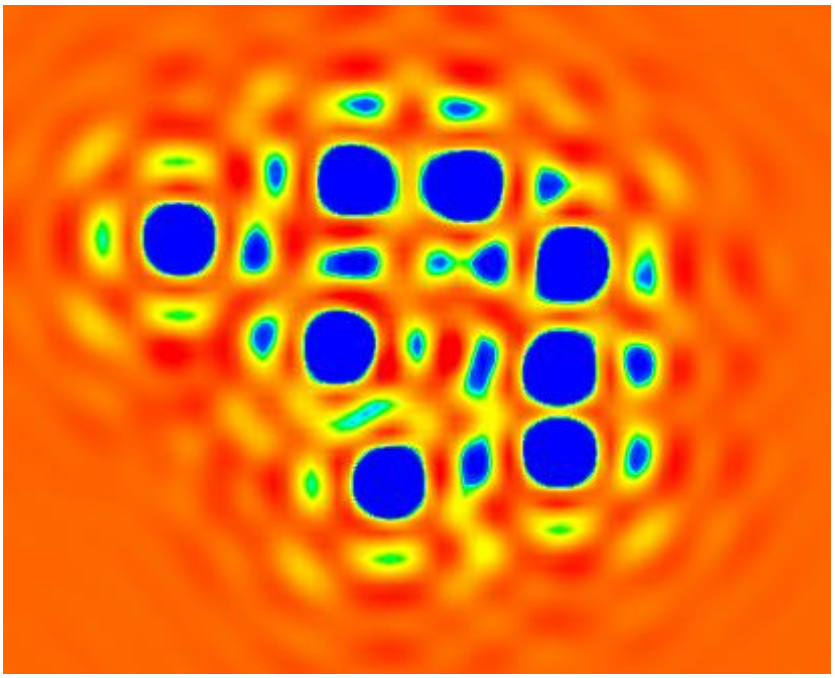


Figure 12. Output of model-based assist feature placement method for the annular illumination aerial image in figure 11. In this example, the assist features are limited to Manhattan geometries.

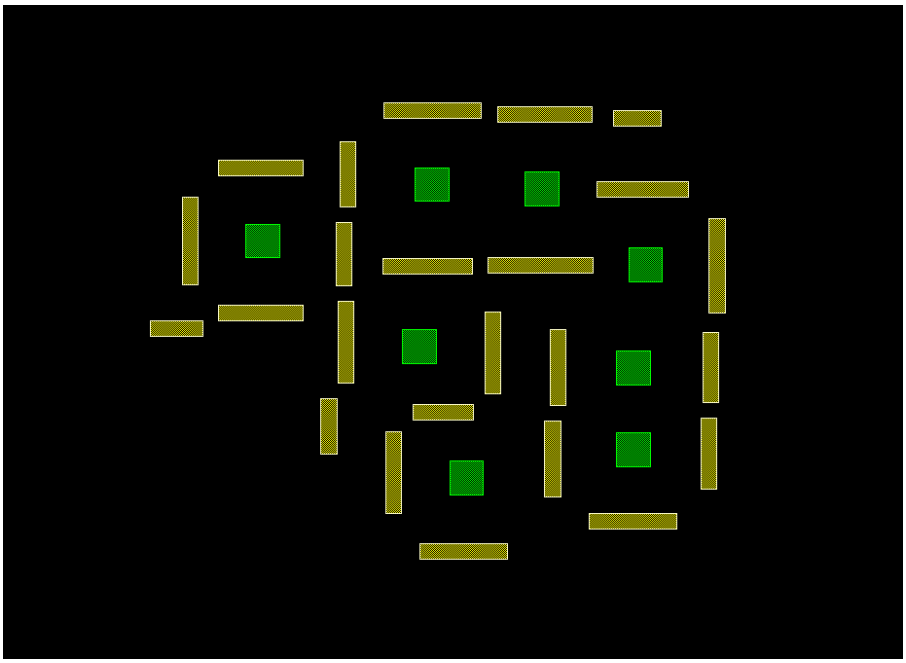


Figure 13. Model-based assist feature placement turnaround time as a function of technology node and image simulation methodology. Field-based methods provide large TAT benefits for advanced nodes.