Systematic failure debug and defective pattern extraction for FPGA product yield improvement

Cinti Chen*, Joe W. Zhao*, Ping Zhang**, Raymond Y. Xu**
*Xilinx, Inc, 2100 Logic Drive, San Jose, CA 95124, U.S.A.
**Anchor Semiconductor, Inc, 5403 Betsy Ross Drive, Santa Clara, CA 95054, U.S.A.

ABSTRACT

In this paper, we have presented an effective yield improvement methodology that can help both manufacturing foundries, fabless and fab-lite companies to identify systematic failures. It uses the physical addresses of failing bits from wafer sort results to overlay to inline wafer defect inspection locations. The inline defect patterns or the design patterns where overlay results showed matches were extracted and grouped by feature similarity or cell names. The potentially problematic design patterns can be obtained and used for design debug and process improvement.

Keywords: overlay defect inspection pattern extraction FPGA yield improvement

INTRODUCTION

As technology nodes shrinking from 65nm to 40/45nm now to 28/32nm, semiconductor industries face tremendous challenges in process development, product design and debug. The fabless or fab-lite companies struggle in meeting the demands of higher transistor density, lower power, faster speed in shorter design cycles and less mask revisions. At the same time, manufacturing fabs or foundries combat process, device and reliability issues with new integration schemes, new equipments and materials, tighter process controls, more sensitive inline monitoring and more sophisticated test vehicles at lowest costs.

Finding systematic failures in designs and OPC/process related marginalities are critical for semiconductor product yield improvement. However, several obstacles have often delayed in finding the root cause of failures. Many Fab's test vehicles can't simulate the design and layout environment of the real products. In addition, PFA (physical failure analysis) has its own limitations such as the cutting angle, the uncertainty of the defect locations, the experiences, skills of the operators and the statistically insufficient sample size. Many times PFA work misses finding true defects. Micro-probing by using bitmaps can make debug process time and resources consuming.

In this paper, we introduced a new tool to overcome these obstacles. It has been used to achieve shorter failure analysis and yield learning cycle on FPGA (Field Programmable Gate Array) products.

FPGAs are semiconductor devices with logic elements and interconnects that provide a unique system-on-chip platform for system vendors in countless markets. Designers use various building blocks and high density memory blocks for DSP (Digital Signal Processing), embedded functions, image processing, computing, and software acceleration applications. The programmability, flexibility and time-to-market advantages are the main attractions of FPGA devices. Furthermore, because the design and wafer cost increase of ASIC (Application-Specific Integrated Circuit) products overweight the benefits as technology nodes advanced to 40/45nm and beyond, FPGAs are replacing ASICs, logic blocks, DSPs, and even general-purpose processors in many applications.

However, several characteristics of FPGA products have made it difficult to achieve fast product ramp-up and short design cycles in the beginning of product introduction when the process and design issues are many while delivery timeline is urgent. First, various design blocks on FPGA have different design rules, different pattern densities and often different design sensitivities to process variability. Second, memory blocks with billions of SRAM (Static Random Access Memory) cells at minimum design rules dominate FPGA. Third, many FPGA products have larger chip sizes

than that of ASICs since FPGAs are all-in-one type of system-on-chip instead of targeted and limited functionalities and applications of ASICs. Fourth, FPGA users always demand fast-to-market product roll ups to gain competitive advantages.

Despite of these facts, FPGA companies have one of the highest profit margins among semiconductor companies. Constantly staying at far front of technology cutting edge, having less design fallouts and process excursions, quick and effective failure diagnosis, controlling costs by religiously meeting defect density road maps are the keys to their success.

The conventional way to do FPGA product debug is to throw different diagnostic patterns to purposely test the readout results. The generated physical failure addresses (bitmaps) can produce physical coordinates of failing memory cells and failing signatures hinting the failing mechanism. The physical address is only approximate and no hint of the exact layers which the defects might have resided. In many cases, PFA can't handle some failure signatures when failing addresses are many.

Just like the common goal of faster product ramp-up requires fabless company to team up with manufacturing foundries as partners, integrating inline wafer defect inspection data, wafer sort yield, test pattern failures into the design circuit layout and physical verification are critical to successful product development and fast yield ramping. Some efforts have been directed to this area in the last decade. For example, Design Based Binning (DBB) is to use both defect inspection images and design GDS clips to group patterns that are failing multiple times [1]. But this approach is limited to patterning related failures. Another team has developed methodology to filter and group those frequently appeared defects by using rule based or pattern based algorithms [2]. This is also limited to lithography patterning related hot spots. It can't help product debug when systematic failures or process excursions occur.

In this paper, we have extended our work on defect to bit failure overlay methodology to extracting defect patterns where systematic issues have caused sort failures. The extracted defective patterns of the failing memory cells then are grouped by similarity or by cell names and used for design and process debug.

DEFECT TO BIT OVERLAY METHOD

Since many of the pattern extraction and grouping are done on real sort failures, we start with defect to bit overlay method. The defect to bit failure overlay method involves overlaying two sets of data plus the design layout file: one is the sort test failures with the physical failure addresses and signatures of memory cells, another is inline defect inspection results with information like layer ID, defect ID, defect class, defect x and y coordinates, die x and y coordinates, etc. This overlaying process can be done by a design layout viewer or being processed through a script.

Table 1 is an example of a standard defect to bit failure overlay result table. The defect caught after spacer etch was matched to a single bit failure at sort test. The delta_x is the distance between the x coordinates of this failing bit's physical address and the defect ID #12. Both delta_x and delta_y are less than 1um suggesting a good match well within the defect to bit overlay matching criteria.

In the initial stage of setting up the overlay method, a confirmation by either inline defect inspection pictures or x-section SEM is needed to make sure that the data conversion and overlay is correct. Figure 1(a) showed inline defect inspection image of this spacer particle as shown in Table 1 and Figure 1(b) the design GDS clip with defect mark at this location. The defect marker indicated the approximate location and size of this defect.

Table 1. A defect to bit overlay result table indicating a defect caught after spacer etch is matched to a single bit failure at sort test.

Lot	WF	Signature	Die	Layer	Def ID	CLASS	def_x	def_y	bit_X	bit_Y	delta x	delta y
N999	7	Sb	45_46	SPEtch	12	particle	-1719.6	272.7	-1719.8	271.9	0.2	0.8

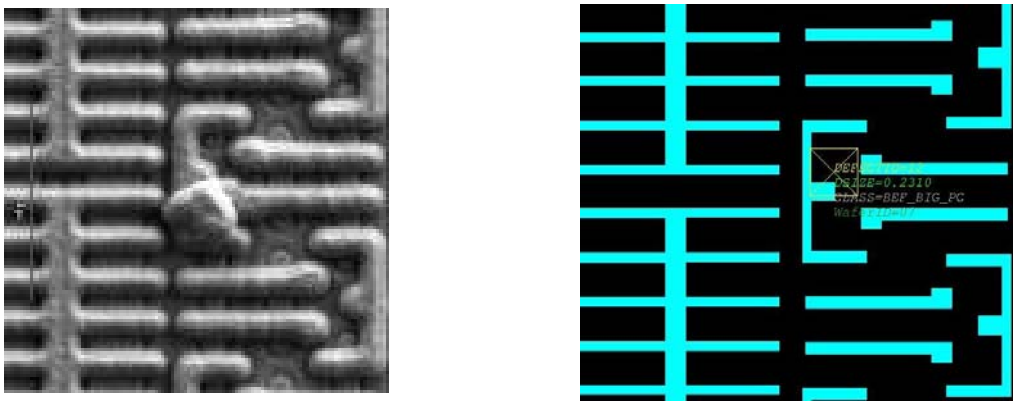

Figure 1: inline inspection SEM review image (a) and its corresponding design GDS clip (b).

PATTERN EXTRACTION METHOD

When memory and logic cell failure addresses and/or inline inspection defect locations are loaded into a design layout viewer, we can extract and group the design patterns where the killer defects or bit failures are located. The extraction and grouping can be based on similarity of the patterns or the cell names. This way, killer defects with highest frequency and the defective cells with the most occurrences are identified and investigated. This is valuable if systematic issue in process exists. We can also use defect pattern extraction to identify some specific process sensitive design layout. Hotspot resist model images can be converted into polygons or traced back to design features and used for defective pattern search. And this method is not limited to lithography pattern defect groupings as most other methods are done. Any systematic defects that can be detected by inline inspection are doable by this method.

1. Pattern extraction of inline defects

Inline defect inspection can produce thousands of defects especially in the early phase of the recipe setup when yield issues are many. During defect inspection, some defect images are captured, classified and reviewed. To increase the throughput, the number of the wafers inspected is limited, the numbers of the images captured are fixed and the defects can be mis-classified. Some defects were caught mainly because of recipe set up sensitivity. A huge amount of defects landed on empty space or dummy area where most small defects will not post harm. Finding true yield killers demands a lot of efforts in recipe fine tuning, defects classification and SEM reviews. What the pattern extraction method can do is to help yield engineers to be more effective in identifying hotspots and containing it before process marginality develops into some yield disasters.

Grouping defects by the similarity and criticality of the design patterns where they are landed on or the cell names where many defects frequent can save engineering time and increase the throughput of inspection steps. So here we start with pattern extraction and grouping of inline defects based on their pattern similarities.

Figure 2(a) and 2(b) are inline inspection wafer view and composite chip view respectively from one wafer at poly inspection. The chip view indicated that most of defects are concentrated in two columns. It suggests that either inspection recipe is too sensitive to the design layout in these two columns, or process related systematic defects occurred in these two columns. Therefore defect pattern extraction and cell extractions were performed as the first step of investigation.

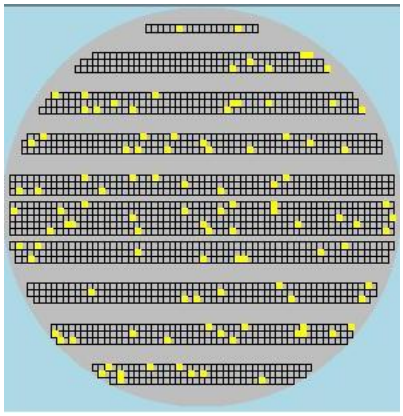

Figure 2(a): wafer view.

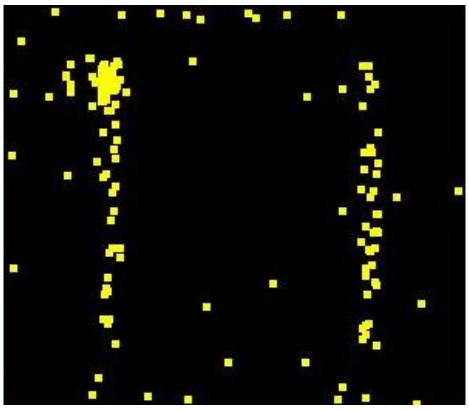

Figure 2(b): composite chip view.

Figure 3 shows the clip of pattern extraction from all the defect patterns where the defects reside on this layer. Majority of the defects in these two columns in Figure 2(b) have this design pattern. Inline inspection clips at one of these defect locations showed some residuals seen on diffusion (active) area (Figure 4 (a)). The tilted x-SEM picture confirmed residual nitride films on top of the diffusion area (Figure 4(b)). This failure pattern extraction directs yield improvement efforts to improving STI CMP marginality at narrow diffusion width in all area of the design layout.

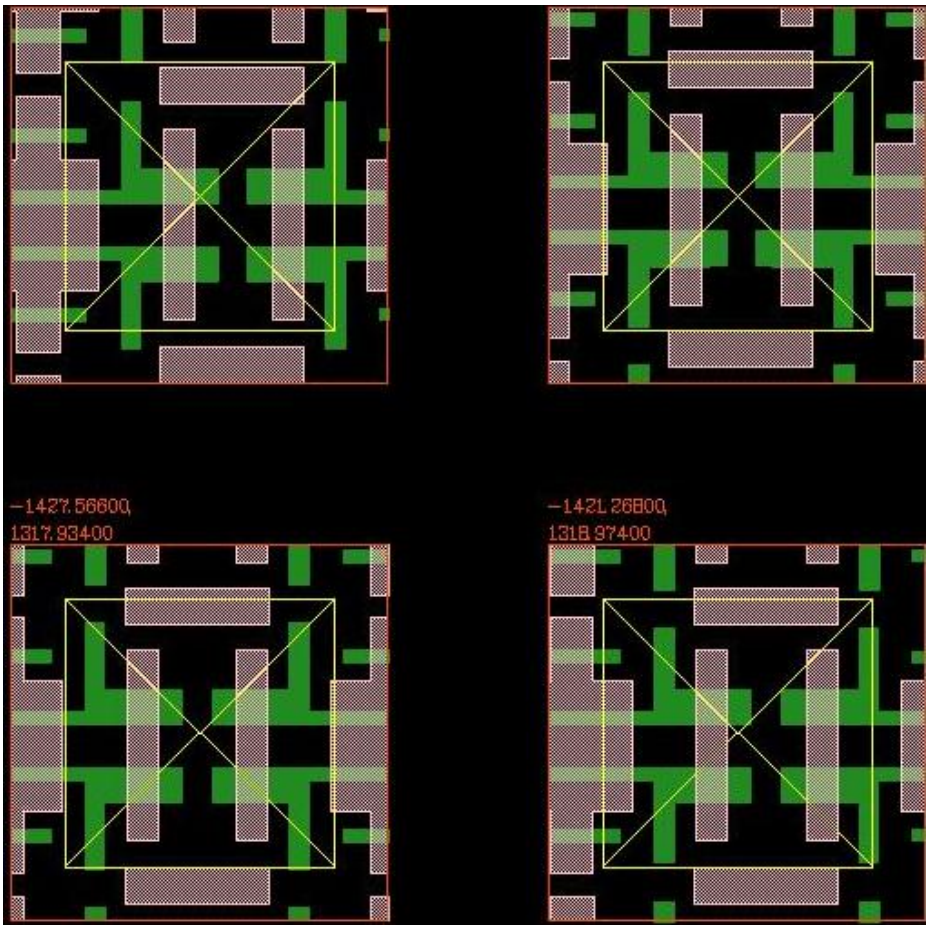

Figure 3: Some defect pattern extraction examples of poly and diffusion layers.

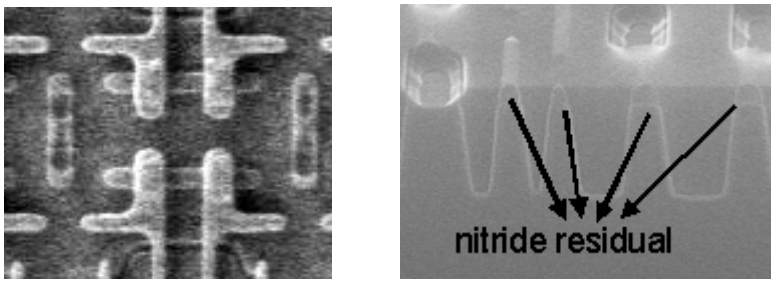

Figure 4: Inline defect top down SEM image showing defects on active area (a) and the tilted cross section SEM showing nitride residuals (b).

2. Pattern extraction of bit failure signatures

The extraction and grouping of the design patterns where the failing bits are matched to inline defect locations can lead us directly to possible systematic failures. This can significantly shorten the debug time since it will point to the most likely design layers or process steps.

In this case showed in Figure 5, we've seen much higher average failing percentage on some lots in certain bit failures than what we've normally seen (at <2%). Some wafers failed this bin signature as high as 24%. Using defect to bit failure overlay method, we picked up many bit failures that were matched to metal layer defects (Table 2). The specific failing bin is b32 and the specific failing signature is "v-d" meaning the failing bits are dual bits laid out vertically.

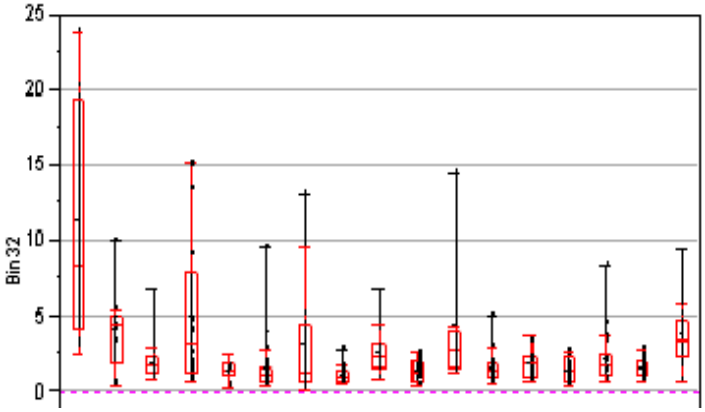

Figure 5: failing percentage of some bin signatures. X-axis is lot ID.

Table 2: part of the defect to bit failure overlay result table showing matches to certain bin32 failure signatures.

LotID	WaferID	Die	Bin	Failure	Layer	DefID	DSIZE	CLASS	Def_x	Def_y	bit_X	bit_Y	delta_x	delta_y
A999	16	-4_2	32	v_d	m1	257	1.8	0	913.3	-5132.5	913.3	-5131.9	0.0	-0.6
A999	16	0_8	32	v_d_Pd	m2	74	1.2	0	1242.1	887.4	1239.9	886.7	2.2	0.7
A999	16	1_-8	32	v_d	m3	20	1	0	5094.6	-5005.5	5095.0	-5007.1	-0.4	1.6

The overlay results were loaded into layout viewer and the defective patterns of these metal defects were extracted. Figure 6 showed some defect pattern extraction clips where the defects on metal 1 layer were matched to this type of bin

failing signature. From the layout viewer, we observed that the common characteristics of these patterns are small metal 1 islands with single or dual contacts and dual vias. Most of the metal islands have the minimum design rules. The inline image clips confirmed this similarity (Figure 7). Further investigations have shown metal open issues on metal trench etch at this type of metal islands.

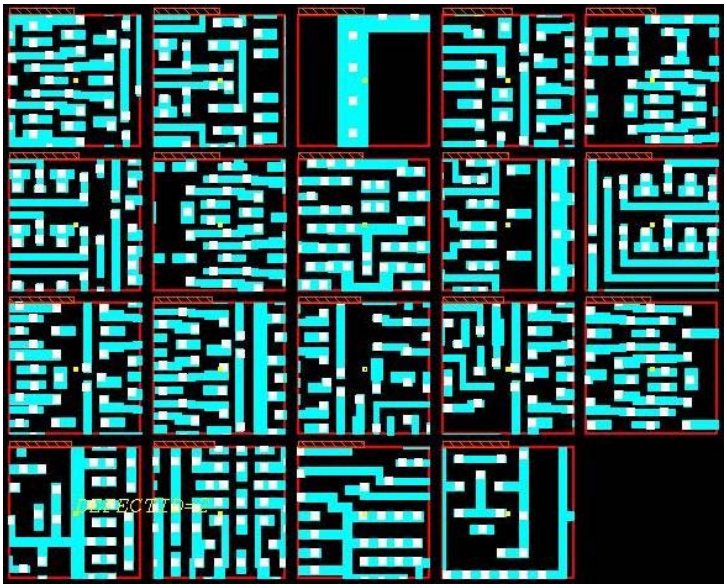

Figure 6: M1 pattern extraction of the matched defect to bin failing overlay results.

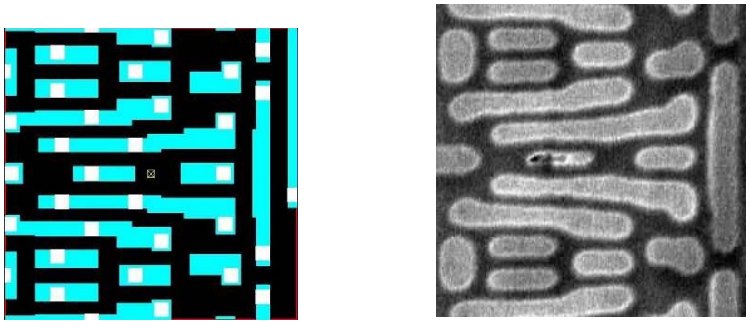

Figure 7: a design clip where a defect is matched to a bit failure (a) and the corresponding inline defect SEM image showing the defect (b).

3. Pattern search of bit failure patterns

During OPC verification, many hotspots are identified and cleared by changing design layout or design rules. However, occasional systematic process issues or design layout hotspots are overlooked for various reasons. So building a hotspot library and using the defective features for pattern search are beneficial for both foundries and design firms.

The pattern search starts with grabbing the defects classified as pattern related defects, such as pattern deform, partial pattern, missing pattern, local defocus, etc. These inline images or OPC hotspot images are loaded onto layout viewer and the design patterns where these defects are resided can be obtained. Then these patterns are added to a defect pattern

library and used to search all the defects caught during the process or all the design database of interests. Figure 8 is a clip of a defect that was matched to many dataline failure addresses in sort. After tracing the defect image to the design layout, we used this feature to search through the design database. Figure 9 showed similar design features that are susceptible to this broken poly pattern.

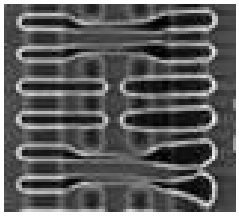

Figure 8: inline defect that is matched to many dataline failure addresses.

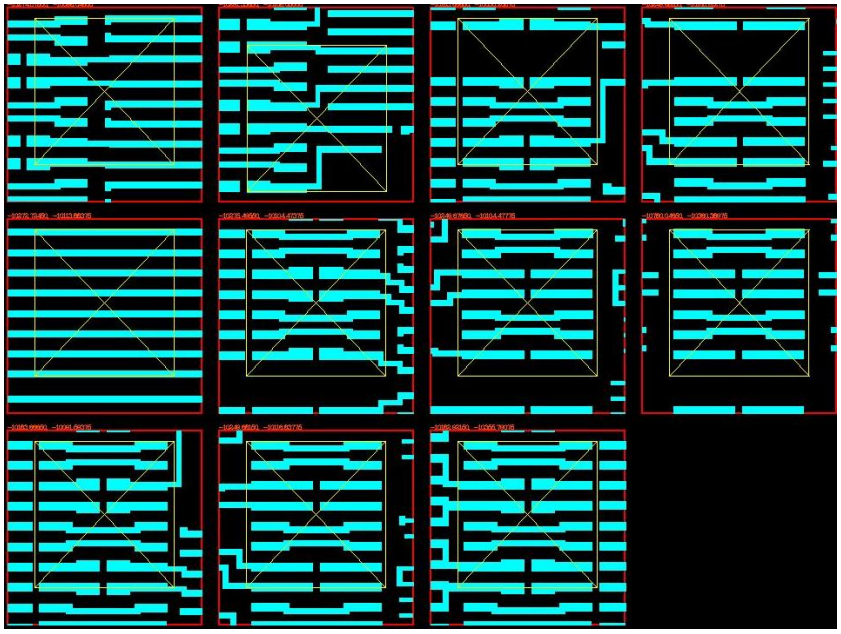

Figure 9: defective pattern search result in tile view

4. Pixel based similarity search of OPC hotspots

The algorithm of the pattern search is based on the similarity of all the pixels in the design clips. The extracted defects are filtered out if the similarity rankings don't surpass the threshold. For example, for all the defects that have greater than 70% of the pixels same as that of the defective pattern in the library, we define it as a candidate for this type of systematic or hotspot defect. The rest is ignored.

The design feature clips are the majority in the hotspot library, but we can also use OPC hotspot resist model as the input for defective pattern search. The OPC hotspots resist images are traced back to the corresponding design layout by image to polygon conversion. Then the corresponding polygon clip is converted into thousands of pixels and used for pixel based similarity comparison through the design database. Figure 10 is a clip of a hotspot that is in danger of short due to resist pinching. This OPC simulated resist profile is converted into spaces and features based on the contrast of the

image (Figure 11). The spaces and features are further converted into polygons for hotspot search. As showed in Figure 12, a search of this type of hotspot picked up many similar patterns.

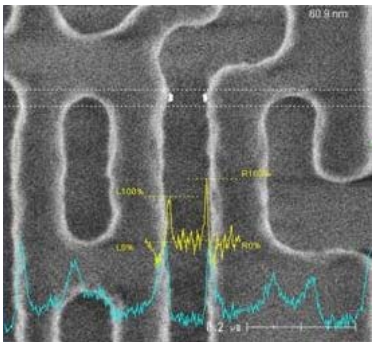

Figure 10: OPC resist profile of a hotspot

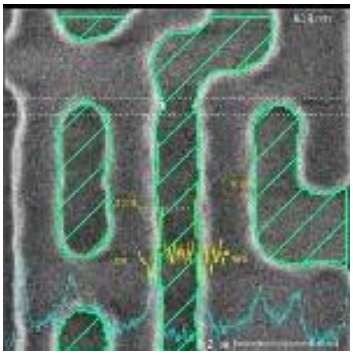

Figure 11: resist profile is converted into GDS features

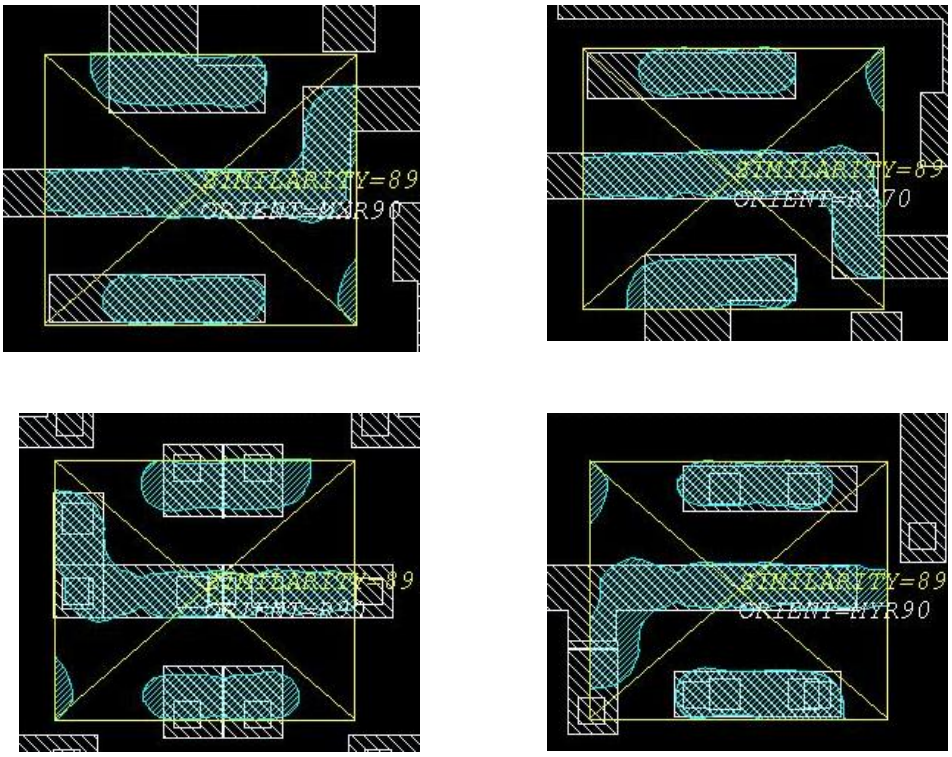

Figure 12: similar pattern search results using a hotspot SEM image as a template.

CONCLUSION

The defect to bit failure overlay method has been extended to defective pattern and hotspot exaction and grouping in order to identify systematic process or design layout failures. Extracting and grouping defects or yield killers by feature similarity or pixel based similarity are shown to be effective and advantageous than conventional debug methods.

ACKNOLEDGEMENT

The authors would like to thank the support and collaboration from Xilinx's foundry partners including Samsung, TSB and UMC.

REFERENCES

[1] Huang, C., Liu, H., Tzou, S.F., Park, A., Young, C., Chang, E., "Effective learning and feed Back to designers through design and wafer inspection integration", Porc. SPIE, vol. 6925, (2008).

[2] Kang, J.-H., Choi, J.-Y., Shim, Y.-A., Lee, H.-S., Su, B., Chan, W., Zhang, P., Wu, J. and Kim, K.-Y., "Combination of rule and pattern based lithography unfriendly pattern detection in OPC flow", Proc. SPIE, vol. 7122, (2008).