Design/Process Learning from Production Test

David Abercrombie, Bernd Koenemann
Mentor Graphics Corporation.
Wilsonville, OR, San Jose, CA, USA
bernd_koenemann@mentor.com

Abstract

Modern Design-For-Test (DFT) practices not only simplify test generation but also make it much easier to diagnose problems uncovered in production test. In fact, many diagnostics steps can be automated enough to enable batch processing of large quantities of fail data captured during product ramp and volume production. Hidden in these fail data is very valuable information about the product design, manufacturing process, and interactions between the two. The embedded tutorial will provide an overview of some of the analysis methods that are being used and/or prototyped in the industry, as well as the underlying data sharing between the design and manufacturing areas that is required for and enabled by the analyses.

Introduction

Design and manufacturing used to exist in two more or less separate worlds. The "fastest path from RTL to GDSII" mantra of the Electronic Design Automation (EDA) industry neglects that silicon is not printed from GDSII and that production test generally stands between the design and shipping working products.

With modern sub-wavelength lithography, the path from GDSII to silicon includes increasingly complex and oftentimes finicky shapes enhancement algorithms for mask preparation. Even with all this effort, the final shapes ending up on the silicon are no longer entirely faithful reproductions of the GDSII. And, not only lateral shapes are subject to distortions. Copper interconnect structures, for example, are experience thickness variations caused by density-dependent effects in Chemical Mechanical Polishing (CMP). These and other emerging processing idiosyncrasies lead to sometimes undesirable parametric consequences in electrical behavior. How significant the parametric aberrations are, depends to a certain degree on how well the design structures anticipate the modifications and variations in manufacturing and how well the extraction/verification tools predict the modifications and variations.

In other words, modern design tools have to increasingly be familiar with and understand the manufacturing processes and how they interact with the design intent. Manufacturing, similarly, has to live with the fact that yield no longer is primarily dominated by catastrophic defects due to process issues like dirt or impurities, but also by parametric defects arising from design-process interactions. Yield ramp and yield management, consequently can no longer exist exclusively in the process domain, but must equally focus on design-related issues.

Electrical test at wafer or final sort is the place where the rubber of design specific parametric behavior first meets the road of silicon reality. It is production test where the statistical impact of certain design weaknesses first has an opportunity to manifest itself. Characterization test may be more thorough but does not test involve samples for meaningful statistical analysis. Analyzing large quantities of production test fails, hence, is the first chance to understand and subsequently alleviate the root causes for the fails. With modern diagnostic analysis techniques, manufacturing can be transformed from a "simple" pass/fail screening operation to an increasingly valuable data acquisition and design/process learning facility.

About Defects

Defects come in many forms and have many different root causes. Historically, manufacturing yield loss was largely dominated by random particle contamination (60% of the loss in mature 350nm processes, according to [1]). The particles interfere with the proper formation of silicon structures.

Figure 1 shows an example of how a particle on a bare wafer leads to malformed interconnects after CMP.

Figure 1: Particle-induced defect [2]

This type of defect can be characterized as random because particles settle on the silicon surface randomly. If a particle happens to hit an area with no circuit structures in close enough proximity, then the particle has no effect on the functionality of the final chip and may be considered “harmless”. On the other hand, if the particle of a certain size hits a so called “critical area” such that it overlaps with or sufficiently encroaches on nearby circuit structures, then the particle could cause malformations of the circuit elements. The yield impact of particles depends on the defect density (number of particles per unit area), their size distribution, and the critical area on the chip.

Figure 2 illustrates the critical area concept for particles that can lead to additional material.

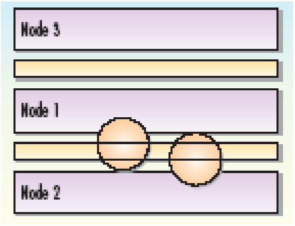


Figure 2: Critical area for additional material

Yield loss due to such randomly distributed particles or impurities sometimes is also referred to as defect-limited loss or area-based yield loss [1].

Another important characteristic of the particle defect shown in Figure 1 is that the particle is so called “visual defect” that can be seen by in-line wafer inspection equipment. Such equipment performs surface scans of selected wafers at selected processing steps in the fabricator. The surface scan images are optionally stored for future reference, and they are post-processed by image analysis software to identify defect locations and defect sizes. The locations (x, y, layer, etc.), sizes, and possibly other classifications are stored in a so called defect map. In-line inspection and defect maps are one vehicle used for continuously monitoring defect densities and defect distributions in a wafer processing facility.

Not all defects are random in nature and affect all wafer areas indiscriminately, but are more systematic in nature and may affect only certain design features or feature combinations. It is predicted that feature-based yield will become increasingly more significant than area-based yield loss for sub-wavelength technologies (see Figure 3).

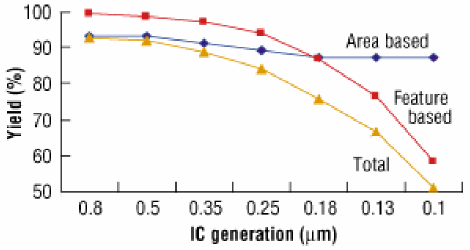


Figure 3: Predicted evolution of area-based versus feature-based yields [1]

The extreme depth versus width aspect ratio of via structures in modern technologies makes it difficult to reliably establish contact between the interconnect layers and resistive opens in via structures have become a notorious yield issue.

Sub-wavelength lithography effects, even with Resolution Enhancement Techniques like OPC, are another example for a potential systemic defect source. Sub-wavelength lithography effects impair the fidelity with which the intended layout features expressed in the GDSII can be printed onto the silicon. Figure 4 shows an overlay some intended layout features (solid shapes) and the predicted silicon image (white outlines) printed from a RET-enhanced mask.

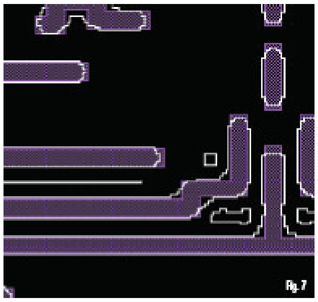


Figure 4: Comparison of intended layout features and predicted silicon image (white outlines) [3]

Despite a high effort expended in RET to preserve the features as faithfully as possible, the predicted actual silicon shapes are not an exact match. The formation of side-lobes in the actual silicon changes the parasitic coupling capacitances and increases the likelihood of shorts between some adjacent features. Hence, both the parametric electrical behavior and the defect-sensitivity of the circuit are impacted in a systematic fashion (i.e., only for certain combinations of layout features). In addition, gate density and new fabrication techniques like dual damascene copper interconnect produce a variety of more or less subtle defect types, such as antennae effects, metal erosion, stress voids, resistive vias and micro-bridges. A couple of examples are depicted in Figure 4a.

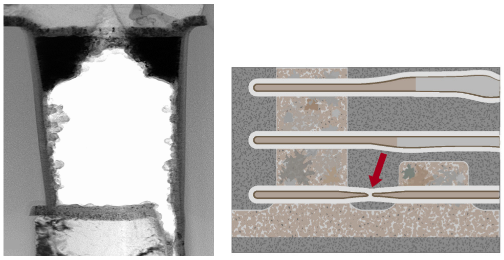


Fig. 4a. Nanometer test challenges: Small geometries and new materials create new defect types, such as resistive vias (left) and opens (right).

In addition to visible physical defect manifestations, other circuit failures can arise from electrical interactions and effects. Cross-talk between neighboring interconnect lines or noise coupling between circuits are examples for such defects. Accurately predicting circuit behavior during design is getting increasingly complex and difficult. The mismatch between silicon and layout shapes, as illustrated in Figure 4, is only one of many effects that can render parasitics extracted from the design data inaccurate and lead to unpredictable marginalities. Although systemic in nature, such effects can conspire with random inter- and intra-die process variations and signal integrity issues to create seemingly random parametric problems. To make matters worse, the parametric defects and some non-parametric physical processing problems are entirely invisible to in-line inspection tools traditionally used for defect monitoring. The International Technology Roadmap for Semiconductors (ITRS 2003) predicts that parametric and non-visual random defects will become increasingly significant and cumbersome for advanced process nodes [4].

Traditional Defect Learning: Defect Maps and Test Structures

The already mentioned surface scan equipment is one method to more or less directly find out about certain defects in the process. However the surface scan alone cannot predict well enough which of the observed defects will actually alter the electrical behavior in a way that leads to circuit malfunctions. Specially designed, easy to test and diagnose test structures are one method by which semiconductor manufacturers learn about which kind and how many defects actually affect the electrical properties of manufactured circuits. Such test structures are routinely manufactured and analyzed during all stages of process development and volume production to quantitatively monitor electrically relevant defect distributions.

Test structures often are designed for the detection and analysis of specific defect types in specific process modules. Figure 5, for example, illustrates a single-layer serpentine structure useful for interline bridging defect analysis.

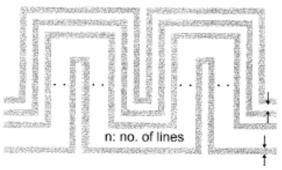


Figure 5: Single-layer serpentine test structure for bridging defects [5]

Regular and simple test structures like the one shown in Figure 5 greatly simplify the characterization of specific defect distributions, but lack the comprehensiveness of the wide range of circuit structures and layout features used for actual product designs. Hence, more sophisticated test chips combine a wide variety of test structures and may even be tailored to more closely represent the circuit and layout features of particular product designs. The Characterization Vehicles described in [1] are examples of highly sophisticated defect monitors.

Defect Learning: Product Designs

Product design substructures and complete product designs can also be useful for defect learning, particularly if the production test methodology is appropriately enhanced for that purpose. To that effect, the productions tests must be able to detect the presence of relevant defect populations and produce sufficient diagnostic data about the fail mechanisms. The diagnostic data must be logged and subsequently analyzed to determine their root causes. This analysis entails characterizing and localizing the likely root cause area (that is, finding the signals in the electrical circuit schematic that are most likely associated with the root cause), and then physically finding the defect (that is, if necessary, de-processing the indicated area to visually find the defect). Detailed Electrical and Physical Failure Analysis (EFA/PFA), tend to involve expensive lab equipment and manual effort by highly skilled Failure Analysis (FA) or silicon debug specialists.

Memory Diagnostics and Bitmapping

Their dense, yet regular structure that can be very sensitive to defects and at the same time simplifies diagnostics has made memories an industry favorite for defect monitoring and learning. Stand-alone DRAM products for a long time were the production vehicles of choice for yield ramp and process monitoring. With logic products (e.g., micro-processors) nowadays often leading the introduction of new technologies, the emphasis has shifted to embedded memories.

In direct access test methods the Automatic Test Equipment (ATE) can directly access the boundary of the embedded memory macros such that the memory test resources of the ATE can be utilized. These ATE resources may include sophisticated programmable pattern generation hardware as well as real-time fail data logging and diagnostic features. In many practical applications memory Built-In Self-Test (BIST) is replacing direct access memory testing. Memory diagnosis involves exciting the defects such that they cause the test to fail and collecting detailed fail data during test in a process called bit-mapping.

If BIST is used, the BIST methodology has to provide enough flexibility and sophistication to match the defect detection capability of ATE. As new process nodes can create new and sometimes unpredictable failure modes, the current trend in advanced BIST development is to improve the timing capabilities (to detect subtle timing fails), richness of test algorithms (to detect known subtle fail modes), and programmability (to allow for adjusting the test to emerging fail scenarios) of the on-chip BIST engines. In addition, more emphasis is given to data logging features for extracting detailed bit-level fail data from the chip. Where applicable, some fail data processing has to be moved entirely on-chip. For example, repairable memories with redundant spare rows/columns for test cost reasons tend to require real-time fail data analysis during productions test to determine how to program the spare address re-mapping data (e.g., fuse block). Neither does typical logic ATE have the real-time redundancy analysis features nor is the data-logging from BIST fast enough for real-time processing. Hence redundancy allocation is moved on-chip. Data logging for failure analysis, by contrast, does not have to be entirely real-time, and some practical trade-offs between data-logging bandwidth and on-chip hardware complexity/overhead can be made. Figure 5a shows some hardware components that make up a modern micro-coded memory BIST engine with on-chip 2-dimensional (both, spare rows and columns) for embedded DRAMs.

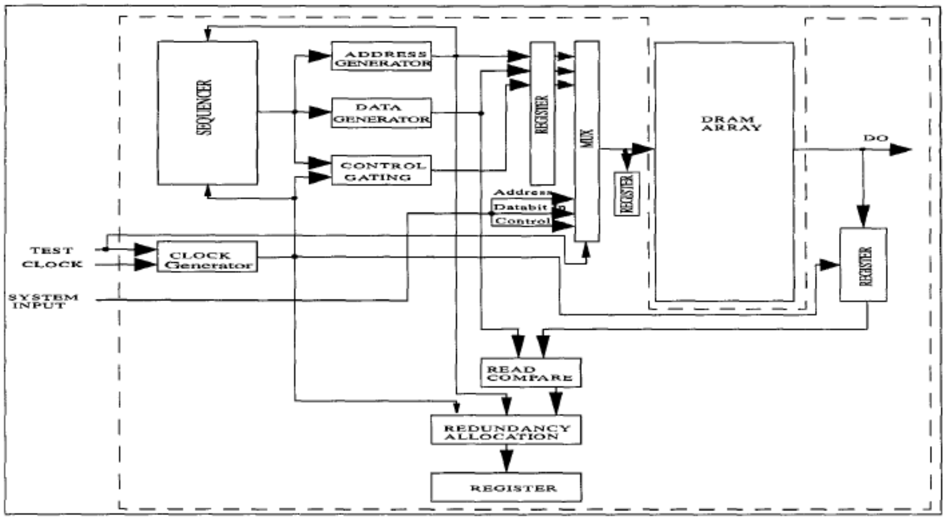
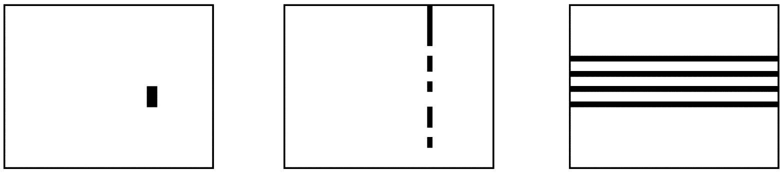


Figure 5a: High Level Diagram of BIST Engine of Embedded DRAMs with 2-Dimensional Redundancy [11]
(ed. note: will insert better picture)

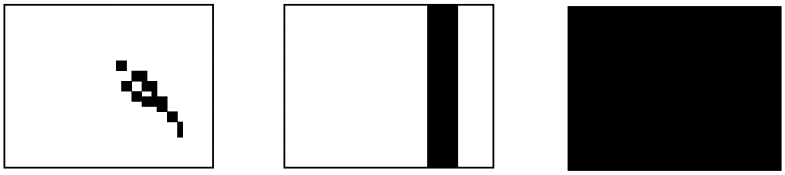
Regardless of whether external test equipment with direct memory access or BIST is used, the test algorithms address the memory in terms of the logic address space of the memory. The initial fail data, hence, consist of logic word/bit errors. To be more useful for defect learning, the logic word/bit data have to be translated into physical row/column data. This involves understanding the physical memory architecture and any address/bit scrambling that is implemented. The bit-map results can be displayed in logical and/or physical array form for each failing chip or for a full wafer. The pattern of failing bits in the memory array can be an indicator for the type of defect to look for. A defect in a single cell will in general create a different fail pattern than, say a bit/word-line or address decoder problem. Figure 5b shows some physical fail bit map patterns related to different fail modes.



Vertical Pair: Bit Line Contact

Partial Column: Resistive Bit Line Short

Multi-Row: Address Decoder



Swatch: CMP Scratch

Entire Bit: Sense amp, I/O

Catastrophic: Timing Circuit

Figure 5b: Physical Memory Fail Patterns and Related Fail Modes [12]

If full physical information including the wafer map is available, then the fail bit maps can be translated from chip-level coordinates to wafer-level coordinates for comparison with defect maps captured by in-line inspection equipment. An example is shown if Figure 6.

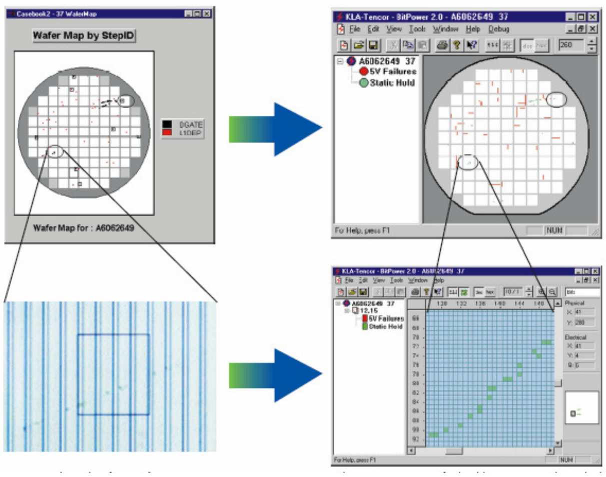


Figure 6: Correlating defect maps (left) and memory fail bit maps (right) at wafer (top) and intra-chip (bottom) levels [6]

This example shows that many different pieces of information contribute to efficient defect learning. For the example at hand, fail bit map data logs from the test equipment are needed together with complete logical/physical information (chip-level and wafer-layout), plus defect maps from wafer processing. Besides having access to the different data types, some logistical effort is involved to make sure that inspection and bitmapping can be performed on the same wafers. The example reveals a matching pattern in the defect map and the physically organized bit map, which likely indicates that individual bit cells are defective.

In addition to visualizing the bit and defect maps, modern bitmapping tools can accelerate subsequent detailed failure analysis by helping to quickly and automatically navigating FA and/or debug analysis equipment like microscopes, probe stations, or FIB machines to the areas of interest.

Logic Diagnostics and Bitmapping

As with memories, defect analysis from logic production test fails is only possible if the relevant defects are actually detected by the production test patterns. And, similarly, advanced logic test methodology developments try to improve the timing accuracy (to detect subtle timing fails), enhanced and more flexible fault models (to force the Automatic Test Pattern Generation, ATPG, tools to generate more stringent tests for relevant logic failure modes), and better fail data logging (to enable large-scale fail data collection during production test). One good example of the trend to enhance the tests for relevant fail mechanisms is to derive fault models from the physical design. Figure 6a shows a representative tool flow for extracting realistic bridging faults from the physical design.

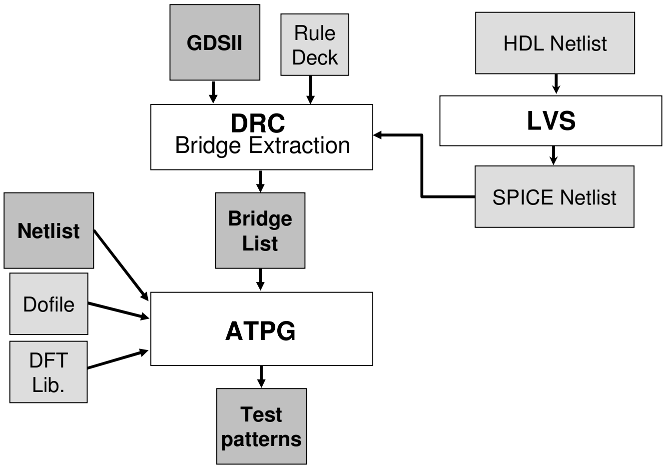


Figure 6a: Representative Flow for Bridging Fault Extraction from Layout

More stringent tests tend to require additional test vectors that can translate into longer test times and test costs. Additional emphasis, hence, is placed on test compression techniques that help reduce both, the test data volume and test times, to enable cost-effective testing on existing and/or new lower-cost logic ATE.

Logic design structures can be quite different from memory design structures and be subject to different failure mechanisms and design sensitivities. Embedded memories furthermore may not use all metal levels and therefore may not be able to expose defect sensitivities in all layers of the interconnect stack. On the other hand, logic is much less regular than memories are and therefore diagnosing logic fails can be more difficult. As a compromise between regularity and logic, Field Programmable Gate Arrays (FPGAs) have become increasingly popular yield learning vehicles. FPGA testing and diagnostics are very specialized skills that are primarily available to and used by FPGA companies. Very few commercially available tools exist.

However, some automated logic diagnostic tools are available for scan-based logic designs and they are used as part of increasingly sophisticated debug, failure analysis, and yield learning systems for logic products. To enable automated diagnostics, fail data logging must be implemented on the test floor. The fail sets must identify failing scan cells and chip outputs for some number of failing test vectors. Since fail data logging can take time, it typically is implemented at a sample level and may not be performed for all chips in production test. Furthermore, existing logic ATE tends may have limited data logging memory, meaning that it is not viable to collect complete fail data for all failing tests. The raw map of the failing scan-cells/outputs can to some extent help to grossly localize areas of concern, but is not always enough. As illustrated in Figure 7, the root cause location could be quite remote from the failing scan cell or primary output. It is the objective of logic diagnostic tools to track down these “remote” locations.

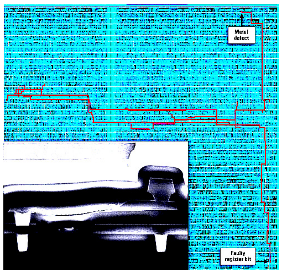


Figure 7: Example of distance between root cause location (upper right) and failing scan cell (lower right) [7]

The automated logic diagnostic tools require access to the fail sets logged by the test equipment, a gate level netlist of the design under test, the original test vectors from Automatic Pattern Generation (ATPG), and a mapping between the tester fail data (typically in tester cycle/tester pin format) and corresponding test vectors in ATPG-format.

There are two approaches to diagnosis. Some tools use a so called static fault dictionary that is pre-computed once up-front from fault simulation. For each fault, the dictionary records the first failing test vector and the expected scan-cell/output fails if the fault is present. The second approach, sometimes called post-test simulation creates a new, reduced fault dictionary targeted to each fails set under analysis. First, a netlist back-trace from the failing scan cells/output determines a subset of logic gates that could be responsible for the observed fails. Logic analysis can further reduce a set of potential faults in the extracted subset that are logically consistent with the values captured at the tester. Diagnostic fault simulation is performed for the thus reduced subset of candidate faults.

Regardless of whether a dictionary or post-test simulation approach is used, the diagnostic software tries to find the best possible matches between fail patterns predicted by the simulator and the fail patterns observed by the tester. Additional analyses (e.g., bridging fault analysis) may be performed if the model faults do not create a good enough match to the reality of observed fails.

The ideal result is a small set of logic nets/pins that are most likely associated with the root cause location for the observed test fails. Even narrowing the candidate list down to a single net/pin in the gate level model by itself does not provide enough resolution to immediately find the responsible defect or defects. The call out list from the logic diagnostic tools, hence, represents a starting point for subsequent more detailed electrical and physical analysis. Like in memory bitmapping, the results from the logic domain must be translated into the electrical and physical domains. The FA/debug lab specialist may use an array of analysis equipment to further narrow down the search space to a more manageable area (e.g., transistor, net segment, via, etc.). Typical equipment used at this point includes probing equipment that can acquire additional internal circuit switching/state information (e.g., voltage contrast e-beam, laser voltage probe, time resolved photon emission, microprobe), FIB machines (e.g., to drill holes for microprobes and/or perform circuit edits), thermal and other imaging (e.g., to visualize hot spots), as well as a variety of environmental control units (e.g., thermal control unit) or equipment for controlled modification of circuit behavior (e.g., light-induced or temperature-induced voltage alteration). The final objective in the case of physical defects is to produce an image of the defect (e.g., SEM image as illustrated in Figure 7).

Like for memories, there can be value in correlating locations and layers from in-line defect maps with callouts from the logic diagnostic tools. This so called logic bitmapping is only possible if the logic nets can be linked to layout shapes and the chip-level layout coordinate system can be translated to the wafer level coordinates. If a defect is found that is on or reasonable close (overlay registration and equipment accuracy limits have to be allowed for) to a suspected layout shape on some layer, then the area/layers for subsequent analysis can be considerable narrowed down. In some cases it may not even be necessary to perform physical failure analysis because the defect type and the electrical fail syndrome are enough to for the FA engineer to understand the problem.

Figure 8 illustrates the logic bitmapping process.

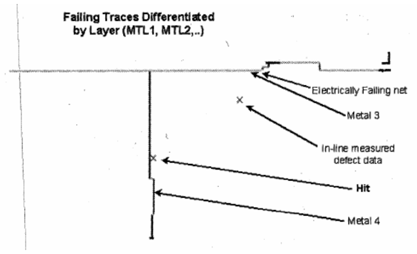


Figure 8: Logic bitmapping example showing overlay between electrically failing net and defect locations [8]

Statistical Approaches

The above defect learning examples primarily focus on understanding the defects that cause a particular chip to fail. While that is an important part of defect and yield learning, it is equally useful to extract significant trends by statistically analyzing many fail sets.

Statistical analysis can be applied to data collected for test chips/defect monitors, memories, and product chips. The data analysis can target design-related issues and process-related problems. Large amounts of logistical and technical data are routinely collected during wafer processing. The data types include Work-In-Process (WIP), equipment status and parameters, in-line inspection, metrology, electrical test, and many more. Increasingly, these data are consolidated into data warehouses using modern relational database technology from where they can be accessed and processed by data mining, statistical analysis, and visualization/report generation utilities.

Figure 9 shows a diagram of the emerging e-diagnostics infrastructure for manufacturing data collection.

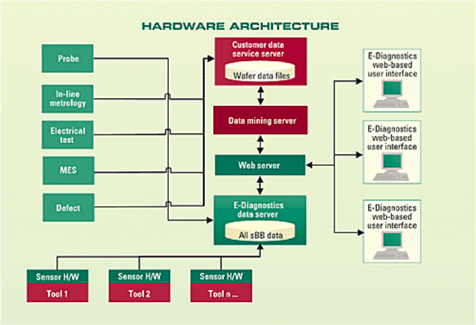


Figure 9: Manufacturing data collection infrastructure with e-diagnostics [9]

The analysis software tries to find correlations between yield problems and manufacturing events (e.g., equipment maintenance), equipment parameters, metrology data, defect data, and so on.

Finding correlations between yield and design characteristics can be accomplished by extending the statistical approach to the design-space. For example, automated logic diagnostics can run overnight for many fail sets logged during testing and then querying the results for design-related commonalities. For example, the failing nets/pins called out by logic diagnostics can be sorted by library cell type or cell instance (including relative location on the chip). The sorted results are can then be visualized in Pareto charts. If, for example, one particular cell type stands out, the some samples from that fail set population can be submitted for detailed diagnosis and failure analysis.

A graph showing a fail rate distribution constructed from logic diagnostics results for many fail sets and sorted by cell type is shown in Figure 10.

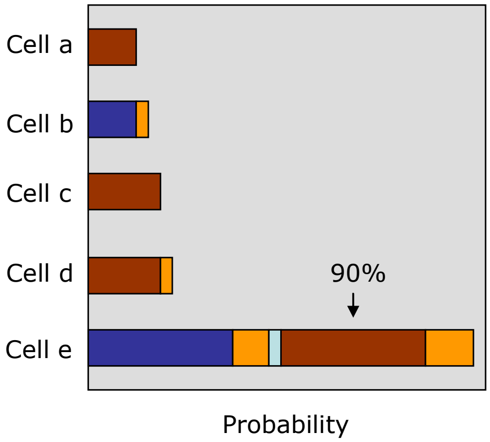


Figure 10: Frequency of logic test failure sorted by cell type/location [10]

This graph reveals that a particular cell affected more fails than any other cell type. Tracking cell locations pointed to the boundary between two vastly different layout-topographies, which happened to affected poly-etching at that transition.

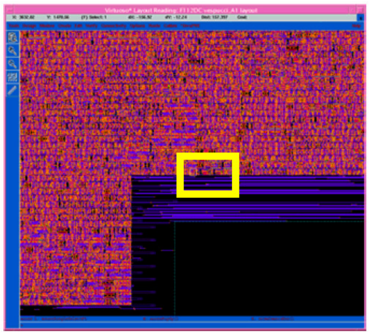


Figure 11: Yield sensitive chip area identified by sorting logic diagnostics results according to cell instance/location [10]

The Connection to Design For Manucturing (DFM)

The unstated power of DFM is hidden in expanding the resources leveraged towards yield improvement. By empowering the design community to contribute to activities that improve yield, there is hope in overcoming the potential loss in yield caused by the new technology sensitivities alluded to earlier. The key to empowering the design community is in identifying the features that the designers can change to improve the yield and characterizing the yield impact of each such feature for making trade-off decisions. Feedback of real fail probabilities from manufacturing is crucial for tuning the up-front DFM efforts for best return on investment.

The traditional FA based yield learning methodology can provide some indication of the features and mechanisms causing yield loss, but the low throughput of EFA/PFA labs limits the ability to quantify the statistical impact of the mechanisms or determine whether all the problems have been found. With the automated volume diagnosis methods outlined in the previous chapter it is possible to generate large, statistically meaningful datasets about the yield performance of individual nets within the design. New design feature analysis tools, such as Critical Area Analysis (CAA), generate detailed statistics about the layout. These types of tools are normally applied to the chip as a whole to predict yield, but we are applying them at the net level to create a statistical database for correlation to the net yield statistics generated by the compressed production SCAN diagnostics. The basic flow chart of the correlation process is shown in Figure 12.

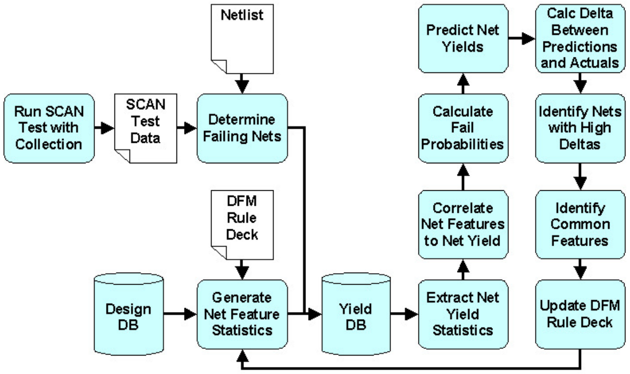


Figure 12: Flow chart of physical yield analysis flow

Table 1 shows an example of the type of data that can be created with this methodology. Each row represents the data for a given net within the design. The first group of columns for each net contains the statistics about the design features contained on those nets. These are calculated using the layout analysis tools. The "Actual Yield" column on the right hand side of the table represents the yield of each net calculated over a large statistical sample of production test data using compressed SCAN diagnostics.

Unit	#	mm	mm	mm	#		%
	Open Mechanisms			Bridging Mechs			
NetName	Single Vias	Length M1	Length M2	Min Space M1	Min Space M1-45		Actual Yield
N348342	2	3.72992	1.29073	0.70642	3		99.979%
N723000	5	2.41953	2.13749	0.55512	3		99.979%
N722774	4	4.54991	1.95610	0.52412	3		99.977%
N868686	2	4.81466	1.90007	0.26846	2		99.981%
N906825	3	0.82530	1.95039	0.44865	3		99.984%
N430501	2	0.24455	2.05877	0.72941	2		99.985%
N306671	2	3.05894	2.20033	0.25910	2		99.985%
N960836	3	3.19550	2.00096	0.24427	3		99.983%
N631146	2	3.50581	1.57913	0.48508	4		99.980%
N899470	4	1.33828	2.59918	0.39739	4		99.979%
N001955	2	4.36102	1.68437	0.41973	2		99.978%
N022249	2	4.39427	1.62443	0.66852	2		99.981%
N309851	3	2.48415	2.88285	0.72739	2		99.978%
N806446	3	3.83013	2.05903	0.63866	3		99.977%
N579008	2	4.16777	1.66266	0.55018	2		99.981%
N224536	4	5.04853	2.25012	0.55275	4		99.976%
N200866	5	4.38409	2.12106	0.45839	3		99.979%
N331473	2	2.59351	2.36975	0.50468	3		99.980%

Table 1. Net statistics database physical yield analysis.

By correlating each of the feature statistics to the actual yield across all the nets in the design, estimations of the failure rates of these features can be calculated. Figure 13 shows a couple of example graphs from this process. The slopes of linear fits of these graphs represent the approximate failure rate of each feature. The spread in the data is caused by the yield impact of all the other failure mechanisms that vary on each net.

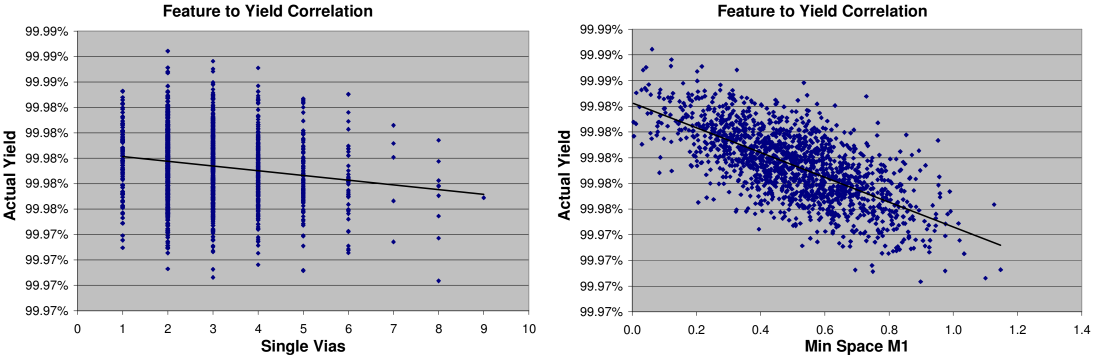


Figure 13: Correlation graphs of design features of nets to yield of nets

These slopes can be put back into the database (Table 2) as Fit Fail Probabilities and when multiplied by the design statistics for each net can be used to calculate a predicted yield for each net. The delta between the predicted yield and the actual yield represents the impact of design features not yet identified through the yield learning process.

Unit	#	mm	mm	mm	#		%	%	%
Fit Fail Prob	**0.0004%**	**0.0010%**	**0.0009%**	**0.0097%**	**0.0010%**		**77.94%**	**69.08%**	**8.86%**
	Open Mechanisms			Bridging Mechs					
NetName	Single Vias	Length M1	Length M2	Min Space M1	Min Space M1-45		Predicted Yield	Actual Yield	Unknown
N348342	2	3.72992	1.29073	0.70642	3		99.984%	99.979%	0.005%
N723000	5	2.41953	2.13749	0.55512	3		99.985%	99.979%	0.007%
N722774	4	4.54991	1.95610	0.52412	3		99.984%	99.977%	0.007%
N868686	2	4.81466	1.90007	0.26846	2		99.988%	99.981%	0.007%
N906825	3	0.82530	1.95039	0.44865	3		99.989%	99.984%	0.005%

Table 2. Failure probabilities and comparison of predicted yield with actual yield.

From this data we are now able to quantify the failure probabilities or yield impact of design layout features, predict the yield of any design element by their feature statistics, and quantify the remaining impact of design sensitivities yet to be discovered. The scarce and valuable EFA/PFA resources can then be focused more productively on problems that exhibit particularly high levels of "unknown" yield loss to identify the next feature to add to the table. At that point you simply generate the design statistics for that feature for every net, add that column to the left hand side of the table, and update the yield prediction to find how much of the unknown has been explained by the new find.

This process is similar to test chip based yield learning acceleration programs being offered today. In this scenario, companies design test chips with test structures containing large numbers of individual design feature occurrences (ex. via chains or metal combs). After running wafers through the fab with this mask set and testing the structures, the failure probabilities for the features on the test chip are calculated. These estimates are then combined with chip level design feature statistics to predict the yield impact.

- The new methodology can complement the other methods and can be very cost-effective:
- No cost of designing the test chip
- No cost of manufacturing a mask set
- No cost of running wafers
- No cost of special testing because production compressed SCAN test results for diagnostics are used
- No delay of waiting on the cycle time to run the test chip because we use production wafers
- If we discover a new feature we do not have to re-design, re-mask, re-manufacture, and re-test a new test chip to characterize its impact

- Reduces the risk that the test structures in the test chip may not accurately reflect the context causing the yield issues in a real product and therefore may not detect the issue.

Another advantage to this technique is in yield excursion analysis. Since there is a baseline yield impact per feature, if there is a sudden yield drop in production users can run the net yield diagnostics on bad lots and compare the correlation slopes of these lots to the baseline. The slope of the feature causing the increased yield loss will increase as compared to the baseline identifying the source. This enables containment of the yield excursion within hours of analysis as compared to days or weeks with the FA or test chip methodologies.

Summary

Understanding defects, both in the process and the design domains, is an important part of yield learning. The industry has developed and is using increasingly sophisticated and automated methods for defect learning. The methods range from dedicated easy-to-diagnose test structures, to detailed diagnosis and failure analysis of individual failing chips, all the way to statistical analysis of large amounts of manufacturing data and electrical test fails from many chips.

With design and manufacturing becoming more and more intertwined, the success of defect learning is increasingly dependent on bringing information from all domains of design and manufacturing together into an integrated analysis environment.

References

1. D. Ciplickas, et al., “A New Paradigm for Evaluating IC Yield Loss”, Solid State Technology, September 2001.
2. Skumanich and E. Ryabova, "Methodology for Targeted Defect Reduction and Inspection Optimization", IEEE/SEMI Advanced Semiconductor Manufacturing Conference, pp. 72-76, 2002.
3. M. Rencher and F. Schellenburg, “Why Interconnect and Lithography Modeling Impacts Yield”, EE Design (http://i.cmpnet.com/eedesign/2003/inside_eedesign6.pdf), pp. 23-27, 2003.
4. International Technology Roadmap for Semiconductors, “Yield Enhancement” Section, 2003
5. C. Hess, et al., ”Fast Extraction of Defect Size Distribution Using a Single Layer Short Flow NEST Structure”, IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING, VOL. 14, NO. 4, pp. 330-337, 2001
6. K. Bernstein, “Yield Enhancement with Bitmap Analysis”, KLA-Tencor, Company Magazine, pp 22-24, Autumn 1998
7. A Weber, “APC: 2004 Technology and Market Update”, Semiconductor International, 7/1/2004
8. H. Balachandran, et al., “Correlation of Logic Failures to a Suspect Process Step”, International Test Conference, pp. 459-466, 1999
9. V. Kot and M. Yedatore, “The Next Step in E-diagnostics: Mining the Tool Sensors”, Semiconductor International, 10/1/2003
10. D. Appello, et al., “Yield Analysis of Logic Circuits”, VLSI Test Symposium, pp. 103-108, 2004
11. P. Jakobsen et al., “Embedded DRAM built in self test and methodology for test insertion”, Proceedings ITC, pp. 975 – 984, 2001
12. From a presentation by R. Aitken, ARM/Artisan
13. R. Press, R. and J. Ferguson, “Enhancing Manufacturing Test and Yield in the Nanometer Era”, Semiconductor Manufacturing, pg 38-44, August 2004.
14. S. Cook, “Rethinking Test at 130 Nanometers and Below”, eedesign.com, 9/12/2003.

15. Lin et al. "High Frequency, At-Speed Scan Testing." IEEE Design & Test, Sept./Oct. 2003.

16. W-T. Cheng et al, “Compactor Independent Direct Diagnosis”, IEEE Asian Test Symposium, 2004.