Yield Loss – It's Not All Process Defects

Copy article that ran on the webzine Semiconductor Online, November 2007

Using test management software, a systematic approach to improving test yields has been developed, which involves: assuring the integrity of test data, the generation of test process rules, and detecting and eradicating errors in the hardware and software of test cells.

Test has become a very critical process in the competitive world of deep submicron device manufacturing, where extremely small particulates can cause random fails. Systematic fails are on the increase as well. Therefore, test is no longer just thought of as a back-end process used to sort good devices from bad devices. Test is now a driving force in the closed-loop IC improvement cycle (Fig. 1) because the test phase is where all available data regarding the behavior of the device can be brought together to isolate root-cause failure modes and locations. The success of this loop is the key to rapid yield and reliability learning, which is so important to time-to-profit.

Generally speaking, those in the semiconductor industry are familiar with major yield detractors that occur in the various fabrication process steps, as well as in design marginalities (the other major source of yield hits). However, the test process itself can also be a yield detractor. The negative impact of test can go undetected no matter where you are on the yield curve. Many IDMs and fabless companies do not realize that they are losing up to 5% of good product in their test operations. With today's evolution away from vertically integrated IDMs, this yield loss can become not only more hidden but also a more significant drag on profitability. Loss of product does not occur on just one or two products, but across the board. In the past, this loss of good product was thought to be extremely low and considered "collateral damage." In other words, losing some good parts was thought to be unavoidable, considering the volumes, product mix, various test recipes/rules, retest/lots on hold, and number of insertions and handling. But today there is no excuse for such yield loss. Although subtle issues at test can cause false rejections, yield loss is controllable. State-of-the-art test management software can help control yield loss by providing a clear picture of all aspects of the test floor, can track exactly what is happening to each device, and can make real-time decisions that are driven by accurate data.

<u>Yield loss detection at all sites</u> This is not to minimize the importance of test in accelerating yield and reliability learning. Closed-loop learning has been successfully practiced for decades. However, given modern IC technology and end markets, learning to improve yield and reliability must happen more quickly with more accurate results. In today's fab process, random and systematic defects are increasing in frequency and are much more subtle and difficult to detect. Therefore, test results alone are inadequate to isolate some of these failure modes. Data must be used from the entire enterprise to get to

the root-cause failure mechanism to provide corrective action to the fab. The difficulty increases with today's business model of having chip design, fabrication, test and failure analysis partitioned across different companies in different regions of the world. The new geographically scattered fabless model makes it more critical to have an integrated network that allows quick, precise data management and communication.

IDMs and fabless companies need to see yield/reliability learning as the key to effective, comprehensive test management. What's more, test management must be viewed as a strategic initiative to collect, manage and leverage the massive amounts of data from the fab, electrical test, various test/burn-in insertions, data logging, diagnostics and failure analysis, as well as final assembly and distribution. It cannot be done piecemeal; it has to be a coordinated automated process with both real-time and off-line capabilities. Below we will look at the specific situations where these data exchanges occur. Most important for the test manager is having real-time accurate data that can be trusted. This is not easily accomplished because typically, software applications are developed at different times by different teams internally and externally. The resulting differences in design techniques, architectures, technology, and data formats are compounded by the "let's keep modifying what we have" syndrome -- cheaper in the short term, but more costly in the long term. Obtaining better yield and reliability can be a daunting task given existing infrastructure, various software tools in use, multiple sources of data with format differences and customized applications throughout the operation. However, the good news is that there are opportunities to make substantial gains in the short term while re-constructing the enterprise for longer-term strategic benefits. Specifically, several points of yield are available by implementing a comprehensive test management system across an enterprise.

The opportunities lie on each of the test floors -- by revealing process errors that were previously inaccessible and pinpointing the root-cause failure so it may be addressed quickly. Test centers, staffed by specialized engineers, have been continuously improved, becoming more effective, efficient and robust. In addition, test managers continue to fight the on-going battle to lower the cost of test. Nonetheless, everyday world-class test floors are rejecting several points of good yield available from the fab. It may seem hard to believe, but that's because this yield loss is not obvious. Just as fab defects are subtler these days, so are the process errors in test operations.

A close look at three areas that are key to running a world-class test center reveals the process errors that are the source of false product rejection. These three critical areas are:
1. Level of data integrity throughout the test sector.
2. Generation and management of test process rules.
3. Subtle errors in the hardware/software of the various test cells.

The remainder of this article identifies the hazards and where improvements need to be made.

Can you be sure of your data's integrity?

Data integrity starts with having a solid hand-shaking process for transfer of all data. A look at the big picture of data flow and communications on the test floor shows that this is not as easy as one might think (Fig. 2). There are so many software applications (Table 1) active on the test floor and involved with interfacing to the rest of the enterprise that the opportunities to lose or corrupt data are mind-boggling.

Since the programs supporting these operations and data exchanges are often developed independently by different sources either internally or through third parties, the data formats can be unique for many of the applications. Often, there is no standardized handshake process for the data movement. For instance, sending data to failure analysis may require someone checking to see if the data arrived without corruption. Moreover, this process is repeated many times, multiplied by having thousands of active product part numbers. If all these data exchanges were within one company at one location, the situation would be complicated enough. But today, operations are spread around the globe with constantly changing locations and elements within each location. Fabrication may occur at one or more locations; wafer test may or may not be done at the fab; chips are shipped to one or more assembly houses; testing may be done at the assembly house or a different location; and finally the product is sent to one or more distribution centers for delivery to multiple customer locations (Fig. 3). Along the way the test results, data logging, diagnostics, failure analysis and quality monitoring must still happen, each with their own set of risks. For example, different test equipment will use different data references, which will require additional data transformations. All of these factors significantly escalate the risk of data integrity issues.

To avoid these compounded risks, your test management solution should provide 100% data integrity no matter how many times and no matter where the data transfer occurs. When you are making decisions that will affect yield, you must have confidence that the analysis is based on accurate data. At this point you may be thinking, "I don't hear about data inaccuracies being an issue." Unfortunately for your operation, this is probably true. As the saying goes, "you only get what you measure and only find what you look for." If your test team knew where good product was rejected due to inaccurate data, they would fix it. The reality is that exposure to data integrity issues needs to be explored by every company. The scope and breadth of the data management operations (Table 1 implemented across Fig. 3) demand tighter control. We are not talking about the obvious, such as consistent missing data, but about more subtle cases where data exchanges drop bits or certain conditions where the data is intermittent. This may cause a "hang-up" or "freeze-up." All too many times the solution is basically a "re-set" or "skip it and continue" because it isn't obvious where the real problem occurs. The solution lies in the software, which needs to monitor and flag errors in data exchanges and data management. The software must execute in a fully automated mode so it can immediately take corrective action upon pinpointing the problem.

Test rules play key role

With all of these handoffs, it's very easy to see how good devices are rejected when false information surfaces due to data corruption. A fundamental area where this can easily happen is in the creation and execution of device-specific test rules. The rules provide real-time control of the test process and include company-wide rules as well as specific ones tailored for individual products. The rules can be developed according to the specifics of the device, the particular test insertion and the test program. As part of a new product introduction process, these rules are designed and developed via a rule set generator. Some of the rules are used in real time at each test insertion, and some are used off-line after test is completed.

An important role of these rules generated over a period of time is to provide control of the test process. Some rules enable test time reduction, some improve quality via statistical process control, and some cover reliability screens as well as disposition of the product. Creating rule sets depends on the test engineer's collaboration with the process engineer to use data from the fab – for example, from lithography, defect monitors and electrical test. This data can be overlaid with test results from all insertions. Some of this can be used in real time or data can be used from other devices, wafers and/or lots for off-line analysis.

The key point to remember is that these rules are running silently in the background and make automatic decisions that affect yield in real time. Test rules are absolutely necessary to run an efficient test area with hundreds of different products. They not only exert control over the test process but can also leverage knowledge for yield and reliability learning.

However there are numerous rule pitfalls waiting to have a negative impact on yield. Different engineers create the rules at different points in time. Many are carried forward from one product family to another and combined with new ones. There can be a lack of discipline; rules are not created with a common template so they can be validated and checked as a set to make sure there are no conflicts between rules. They must also be executed in the proper priority. Many companies rely on manual intervention for rule validation and problem resolution, which is slow, error prone and inconsistent as it depends on the engineer's experience, thoroughness, and skill set (for example, fluency with scripting languages). In addition, thousands of part numbers pass through a test center with a variety of routings and different technical and business requirements, adding to the complexity.

A comprehensive test management solution must have the flexibility to handle these variations and signal on a real-time basis when human intervention is required. In production, the rule sets that execute in real time must have real-time reaction and control. It is very easy for conflicting rules to go unnoticed, resulting in conditions that reject good product. This happens more often than test managers like to admit. The risk is that although the test operation seems to be running fine, for periods of months some good devices are dumped into the reject bin because rules are not adequately validated or viewed as a complete set for that part number. Some examples include:

- Retest of die exceeding allowable number of touchdowns at probe
- Excess number of devices rejected in Parts Average Testing
- Rejecting die because of bad contact on a probe card

The longer these problems go undetected, the more false rejects occur and the root cause proliferates until the failure mode becomes obvious. At this point the floor manager must figure out how long the problem existed and how much material needs to be retested. At wafer probe, retesting a couple of wafers is not unreasonable. But at high-volume final test, retesting is very time consuming and inefficient. The better alternative is real-time software automation that detects and flags the problem in its infancy.

The rule set needs to be seen as a living, evolving collection of software guidelines. A common occurrence is to put the test rules in place and then assume that nothing happens to them. Simulating and validating the rules prior to publishing them is necessary as a first step. During the production ramp-up, the rules also should be checked frequently to assure they are doing what was expected. Then as an on-going quality check, these rules must periodically be validated. Because a change can be made that on the surface would seem to be correct, but may conflict with another rule in the set causing a false rejection criteria, there needs to be an automated process to manage and control the rule sets along with an auto release process so they are copied to all locations that use them regardless of company ownership or geographical location.

Role of the test cell

Beyond data integrity and rules, the fundamental element in the test center is the tester itself, which is a complex piece of hardware. The tester hardware is another source of subtle errors that can falsely reject good product. For example, an instrument in the tester causes an intermittent failure where measurement bits get stuck (known as a “freeze”) and a measurement repeats itself with data that is not true. The answer from a bad chip can get stuck and can falsely cause a good chip to look like a fail. This random problem can go undetected unless the test management software is specifically looking for this pattern.

An advanced real-time statistical process control (SPC) capability can monitor for a parametric consecutive bin limit. This involves monitoring each tester and testing each product with a full suite of SPC capabilities. This monitoring is made easier with test management software that can track the entire tester fleet’s overall performance. By comparing the bins and measurements across all the testers processing the same product, problems with a specific tester can be isolated. Monitoring test equipment on all test floors, including outsourced facilities, provides a way to tell quickly if there is a problem with the hardware, program or process rules.

Customized hardware such as probe cards and contactors also require monitoring to assure no unnecessary yield degradation. All of this monitoring is meant to reduce the amount of product that gets scrapped before the problem becomes more severe. The comprehensive test management software has access to the tester data in real time and performs its own analysis automatically, it immediately provides the resulting information to the appropriate personnel via dashboard, email or pager notifications.

Conclusions

The overall goal of effective test management is reducing false rejections due to processes, procedures, data management and ATE errors. There are potential sources for these false rejections anywhere there are marginalities in the testing process. This approach helps guarantee that test processes are accurate, repeatable and reliable. It allows development of a sophisticated, automated expert system that is capable of real time decision-making across all operations, with built-in data mining and analysis that uses process and test knowledge to identify problems in real time.

This article has explored only a few, major areas that are sources of false product rejection. Unfortunately, there are others. In addition to the issue of yield as a test management challenge, are the issues of test time reduction, utilization, reliability and quality. Best-in-class test management solutions offer innovative, proprietary methods and algorithms that, when properly implemented, offer significant returns on investment.

John Bearden supports U.S. business development efforts for OptimalTest. Previously, he was with IBM for 33 years, holding various test engineering and management positions. Bearden holds a bachelor's degree in electrical engineering from Northeastern University (Boston) and an MBA from St. Michael's College (Colchester, Vt.). E-mail: john.bearden@optimaltest.com

Table 1. Test Data Exchanges

Test program creation/management	EDA data
Product routings	E-test/parametrics/defect monitors
Fab data (litho, etc.)	Rule set generators/test rules
Probe/pad/bump inspection	Central servers – test cell computers
Data transfer within test cells	Set-ups: recipes, probe patterns, etc.
Golden wafers data/history	Test/burn-in results – each insertion
On-tester diagnostics	Post-test diagnostics
Binning/data logging	Physical failure analysis data
Data-to-fusers/sorters/pickers	Lots-on-hold data/history
Burn-in operations	Reliability predictions/sorting
Quality monitors	Statistical post processing
Chip packaging/marking	Inter-company exchanges