Compression Mode Diagnosis Enables High Volume Monitoring Diagnosis Flow

Andreas Leininger and Peter Muhmenthaler

Infineon Technologies AG
Balanstrasse 73
81541 Munich, Germany

Wu-Tung Cheng, Nagesh Tamarapalli,
Wu Yang, and Hans Tsai

Mentor Graphics Corporation
8005 S.W. Boeckman Road
Wilsonville, OR 97070, USA

Abstract

Diagnosis of scan test fail data plays a crucial role in enhancing ramp up of new CMOS technology generations. To enable faster feedback it is preferable to establish a monitoring diagnosis methodology on the production test floor. This paper reports results of a study on using test time optimized compressed scan technology and associated new algorithms for fault diagnosis. Data is based on a system-on-a-chip (SoC) product that is manufactured using Infineon Technologies' 130 nm process. A comparison with uncompressed scan test and diagnosis shows feasibility of implementing a monitoring diagnosis flow with compressed scan test serving the high throughput test flow.

1 Introduction

The ongoing development in CMOS technology that is forecast by ITRS roadmap [1] points to very complex manufacturing processes in the years ahead. Highly sophisticated lithography techniques and new materials are challenging the controlled ramp-up of the new process generations below 100 nm feature size [2]. In addition, products being manufactured at these process nodes are incorporating continuously increasing complexity with respect to on-chip memory, mixed signal and RF functionality and especially logic. Further complicating the scenario, such products are primarily SoCs for highly price sensitive consumer applications that need to be manufactured at high process yield.

While there is a continuous effort to achieve high fault coverage test solutions at acceptable manufacturing test cost, there is an emerging set of requirements to go beyond efficient go/no-go testing and provide detailed digital test responses. Embedded Deterministic Test (EDT) technology [3, 4, 5] has been widely used to achieve a low-cost manufacturing test without any concessions. However, due to the need for faster and more reliable yield ramp-up when introducing new CMOS technologies there is also an increasing demand to analyze failing devices from the production test floor and obtain more information about them. Thus there is a strong motivation to develop a test flow, which offers far more than pass/fail data and first failing patterns as is usually done in the current process for product engineering.

Without losing the benefit of a given compression technique we propose to establish a monitoring diagnosis flow as shown in Figure 1. The basic assumption is that fail data from compression mode scan test is captured by production automatic test equipment (ATE) and passed over to an offline diagnosis procedure where possible suspects of given fail data can be computed.

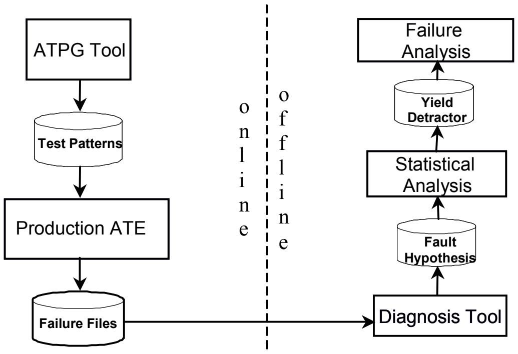

Figure 1: Monitoring diagnosis flow on manufacturing test floor

Devices incorporating embedded compression technology typically contain a mode to bypass the compression logic and provide access to scan chains such that they can be operated in uncompressed scan mode. However, note that it is not acceptable to employ bypass mode to facilitate diagnosis as this would cause inordinate increase in test time and thus reduces the throughput of the test floor by more than a magnitude. Furthermore, it would necessitate ATE upgrades because of the enormous amount of test data and fail response data that needs to be

0-7803-9039-3/$20.00 © 2005 IEEE

transferred from the test floor into the diagnosis database. Hence, diagnosis of all failing chips in the production environment is feasible only with the fail response data obtained in compression mode.

The goal of a monitoring diagnosis such as the one illustrated in Figure 1 is to perform statistical analysis for a given product. From such an analysis significant benefits like identification of systematic yield detractors and sensitive design features can be derived much faster than in the past because the statistical sample is no longer critical.

Design-for-Manufacturing and post layout applications need feedback from the manufacturing process to define and verify the actions for achieving stable high yield. In this context structural test methods which offer the possibility of automated fault diagnosis of defective integrated circuits are evolving as additional source of information for yield learning [6, 7, 8].

Uncompressed scan mode test and fault diagnosis techniques have long been successfully applied in the process of failure analysis [9]. Scan-based diagnosis is much simpler compared to analysis approaches using functional vectors as it does not require detailed design knowledge, and it quickly identifies a list of suspects or potentially defective nets. Based on these suspects, detailed physical analysis for determining the root cause can occur.

To enable the high volume monitoring flow illustrated in Figure 1, the key building block, “Diagnosis Tool” has to support diagnosis in compression mode. The objective of the experiments conducted for this paper is to determine the feasibility of diagnosis with failing responses for EDT patterns and the effectiveness of such a technique. The desired ideal outcome would be to reach the same failure analysis quality using diagnosis with EDT patterns as achieved using the diagnosis with uncompressed patterns. However, note that statistical processing of large amount of fail data provides tolerance to some reduction in diagnosis resolution.

The paper is organized as follows. Section 2 gives a brief overview of EDT technology and presents a methodology for diagnosis of failures captured with EDT patterns. Section 3 presents a brief discussion about deriving bypass patterns equivalent to those of EDT patterns and the bypass mode diagnosis used for comparison. The experimental setup and criteria used for comparison is described in Section 4. Results are presented in Section 5. Finally, Section 6 and Section 7 present conclusions and future work.

2 EDT ATPG and Diagnosis Overview

Many compression techniques have recently been proposed to address the ever-increasing cost of test for the current nanometer designs [10, 11, 12, 13]. The Embedded Deterministic Test (EDT) technique proposed in [3] provides dramatic reduction in scan test data volume and scan test time and has been successfully used. In this section a brief overview of EDT technology is presented. Following this, diagnosis in EDT mode that is vital to high volume monitoring diagnosis flow is presented.

2.1 EDT ATPG

The EDT technology is composed of two complementary parts – hardware that is embedded on chip and new deterministic ATPG software that generates highly compressed patterns, which utilize the embedded hardware.

The EDT hardware is inserted along the scan paths, as shown in Figure 2, and does not require any modifications to the functional logic. It consists of a *decompressor* that maps the few external scan channels into a large number of internal scan chains, and a *selective XOR-tree based compactor* that compacts the large number of internal scan chains into few external scan channels. The compactor is comprised of a number of XOR-based linear spatial trees driven by scan chain outputs. The scan chains in the design are partitioned into multiple groups where each group is connected through selection logic to a separate spatial XOR-tree feeding an output scan channel. Finally, optional logic to bypass the decompressor and compactor and provide direct access to internal scan chains may also be included.

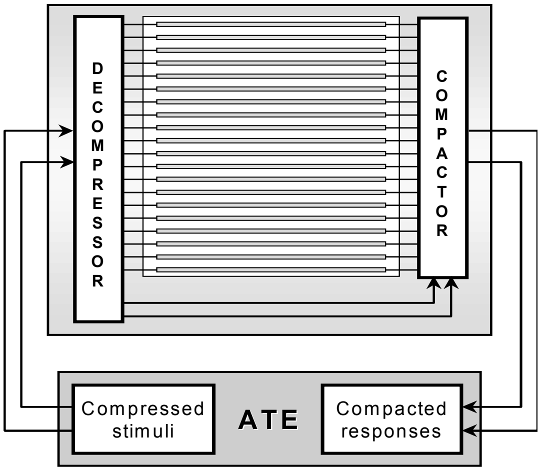

Figure 2: EDT architecture

The ATPG algorithms in EDT utilize the on-chip hardware to generate highly compressed test patterns. It is widely observed that deterministic test patterns typically have only between 1 to 5 percent of scan cells specified. Such partially specified test stimuli are called test cubes. In EDT, unlike the uncompressed ATPG, unspecified scan cells are not immediately randomly filled. Instead, the test cubes are compressed while ensuring that the values for specified scan cells are left unchanged. EDT algorithms also propagate the fault effects and the responses captured in the large number of internal scan chains through the selective compactor. Note that scan chain selection is enabled as necessary to eliminate masking of fault effects due to Xs and to avoid fault aliasing. Thus the captured responses undergo transformation as they are propagated from large number of internal scan chains into few channel outputs.

To the ATE, the compressed patterns generated by EDT appear no different than the uncompressed ATPG patterns except that they are highly compressed. In every clock cycle, compressed test data is supplied on the few external channel inputs and the decompressor expands the test data and feeds the large number of internal scan chains. On the output side, the compactor compacts the response from the many internal scan chain outputs and generates a response. This compacted response is compared by the ATE to the expected response. On detecting a mismatch in a cycle, the ATE can log failure information. In addition, the cycle-by-cycle comparison permits handling of production test anomalies, such as those that cause failure of all devices for certain cycles. These abnormal cycles can be ignored on the production test floor with EDT much like with the uncompressed ATPG.

2.2 EDT Diagnosis

Automated fault diagnosis with uncompressed ATPG patterns has widely been applied with much success. However, with compressed ATPG such as EDT automated fault diagnosis needs to be enhanced. This is due to the fact that with compression, data that is applied to IC and the response that is captured undergo a transformation. In other words, what is observed on the ATE scan output channels is a compacted version of the data captured in the scan elements of the IC. Thus automated fault diagnosis has to be able to take fail log corresponding to the compacted response stream and determine the suspects inside the logic cones.

Recently, a new diagnosis methodology that can be applied in the context of any compactor called *compactor independent direct diagnosis* has been proposed [14]. This methodology is used for implementing EDT diagnosis. With EDT, test responses for each pattern are captured into scan cells and shifted through the compactor on a cycle-by-cycle basis and compared on ATE. Therefore each bit P_i of the compacted response can be expressed as a function of a set of values that are captured into scan cells before compaction. This function, called "transformation function", is denoted as Φ_i such that $P_i = \Phi_i(C_i)$, where C_i is the values of the set of scan cells that are compacted together to obtain P_i. Note that both Φ_i and C_i are determined by compactor architecture. In the case of EDT, Φ_i is XOR and C_i is the set of scan cells that are located in the same shift out cycle in the group of scan chains connected through selection logic to a scan output channel.

An example circuit consisting two scan chains with EDT compactor is shown in Figure 3. In this example, there are 6 scan cells numbered from 0 to 5. For scan cell *i*, its fan-in logic cone is denoted as LC_i. Note that different logic cones may interact, as illustrated in the figure for LC_0, LC_1, and LC_2. This is especially true when multiple capture cycles are used for scan test patterns. After compaction, three bits P_0, P_1, and P_2 are observed in sequence for each test pattern.

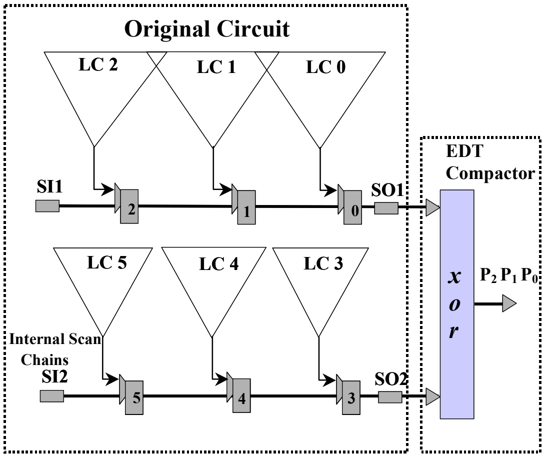

Figure 3: An Example Design with EDT Compactor

In a scenario with interacting logic cones, faults in the logic common across different cones can propagate to multiple scan cells in the same scan out cycle. In the EDT diagnosis approach, the observed fault response P_i is expressed in terms of the candidate logic cones. This direct relation enables diagnosis even when an error propagates to multiple

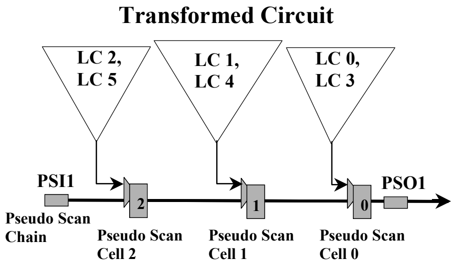

Figure 4: Circuit transformation for direct diagnosis – identifcation of suspect candidates

scan cells within a cycle. To obtain such a relation, a conceptual circuit transformation is performed to convert the original circuit into a circuit containing pseudo-scan chains, the number of which is equal to the number of EDT compactor outputs. The number of pseudo-scan cells in each of these pseudo-scan chains is equal to the number of shift cycles in the original circuit. The pseudo-scan cells are driven by the union of logic cones that have an impact on their value. Figure 4 illustrates the transformed circuit corresponding to the original circuit in Figure 3. The three bits P_0, P_1, and P_2 of the response stream after compaction can be expressed as follows:

$$P_0 = xor\ (\{LC0, LC3\})$$

$$P_1 = xor\ (\{ LC1, LC4\})$$

$$P_2 = xor\ (\{ LC2, LC5\})$$

Suppose a failure is observed at P_1, the *pseudo-scan cell 1*. Based on the transformed circuit in Figure 4, the possible faulty logic must be in union of logic cones LC_1 and LC_4. Thus the initial suspect candidate list is obtained from the union of all logic cones corresponding to failing pseudo-scan cell. Note that in this step of determining the initial candidates, the exact compaction function is not considered. This guarantees that all possible faulty logic sources are in the suspect list. In cases where multiple pseudo-scan cells fail, the suspect list is obtained by performing intersection of the logic cone union corresponding to each failing pseudo-scan cell.

Next, fault simulation is applied on the transformed circuit as illustrated in Figure 5, where each suspect candidate is injected in the transformed circuit and simulated. The simulation results are compared to tester observed failures. A fault *f* is considered a real suspect if the simulation results match with the tester observed failures, as illustrated in the figure for the case when pseudo-scan cell P_1 is faulty. Note that in this step, unlike the previous one, the compaction function is taken into account.

The diagnosis methodology can be summarized into the following steps.

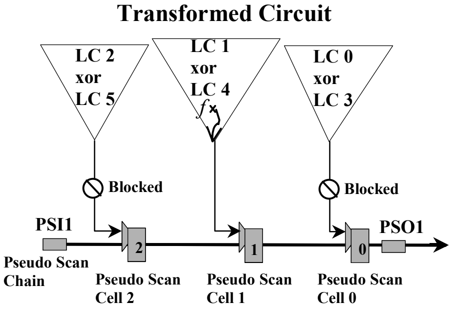

Figure 5: Fault simulation to determine real suspects

1. The first step is to obtain a transformation function for each response bit of a given compactor.
2. The original circuit is conceptually transformed into a circuit containing as many pseudo-scan chains as the compactor outputs. The number of pseudo-scan cells in each of these pseudo-scan chains is equal to the number of shift cycles in the original circuit. The pseudo-scan cells are driven by the union of logic cones that have an impact on their value.
3. The failure file generated by the tester is read in and the failures are assumed to be captured at pseudo-scan cells. Direct mapping relation is built from the failing pseudo-scan cells all the way into the transformed circuit to locate the logic cone(s) that constitutes an initial suspect fault list.
4. For each suspect, fault simulation is performed using the transformed circuit. The simulation results are then compared with the failure data collected from tester. Suspects are ranked based on how well the simulated failures and tester failures match. All suspect(s) based on the ranking scores are finally reported.

The EDT diagnosis methodology is capable of handling even cases where multiple scan cells might be faulty in single shift cycle. In addition, a significant advantage of the proposed diagnosis methodology is the fact it permits diagnosis with production test failures. Thus it is not necessary to bypass the compactor and apply a different test than the production test set for the purpose of diagnosis.

3 Uncompressed ATPG and Diagnosis Overview

The EDT logic can optionally include bypass logic to provide direct access to internal scan chains. In the bypass mode, the decompressor and compactor are bypassed and the short internal scan chains are concatenated to form as many long scan chains as the number of channels. Figure 6 shows the bypass mode scan architecture for the circuit incorporating EDT illustrated in Figure 2.

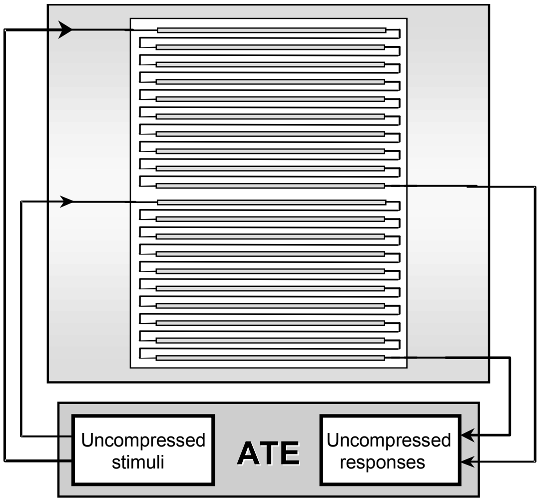

Figure 6: Bypass architecture

The bypass mode can be used in situations where direct access to internal state of the circuit is required, for example in design debug. It may also be used to generate test patterns using the uncompressed ATPG algorithms. However, for the purposes of comparing the resolution of diagnosis with uncompressed versus EDT patterns, it is necessary to ensure that the pattern set used in these two modes are the same. This is achieved in this study by deriving from EDT patterns uncompressed patterns that are equivalent. Such uncompressed patterns applied in the bypass mode have the same scan load values as the EDT patterns at the end of the shift cycles. That is once completed loaded, values in the scan cells match exactly for EDT and uncompressed patterns. In addition, observation of the scan cells is also maintained the same for both EDT and uncompressed patterns.

Once equivalent patterns are derived, diagnosis in bypass mode is the same as diagnosis with uncompressed ATPG patterns. Such a diagnosis technique has been successfully applied for a while and many algorithms exist for this purpose.

4 Experimental setup

In the next subsections, first details about the diagnosis tool employed for this study are described. Following that, details about the SoC product used for experiments are discussed. Finally criteria used for comparing diagnosis results with uncompressed and EDT patterns are presented.

4.1 Characteristics of the Diagnosis Tool

For this study, a diagnosis tool that supports uncompressed as well as EDT patterns was used. This tool utilizes a location-based approach to identify the suspects. In addition, it also classifies the suspects and assigns a score for each suspect indicating how well the suspect behavior matches with the observed failures on the tester. The general flow of this approach is as follows. First, for each failing pattern, it performs critical path tracing starting from the ATE observed failures and identifies initial list of suspect fault locations. At each of these suspect locations a fault is injected and fault simulated to see whether the failure propagates to all observable points with same behavior as observed on the ATE. If indeed the failures observed with simulation match those observed on the ATE, the failing pattern is explained by that suspect fault location. After determining the list of suspects based on all failing patterns, a heuristic method is used to find a set of minimal faults that can explain all the failing patterns.

The diagnostic tool partitions all fail data into several independent *symptoms*. For each symptom, it lists the suspects that explain the failing patterns in that symptom. In addition, new diagnostic algorithms are added to identify defect type for all suspects. Defect type identification classifies the suspects into stuck-at, 2-line bridge, 3-line bridge, open, or indeterminate depending on how well the behavior of the suspects matches with these defects.

Besides identifying suspect types, the new algorithm gives a score to each suspect based on a formula that calculates the similarity of simulated behavior versus what was observed on the ATE. The score is normalized from 1 to 100 with the higher the better. Based on the scores, all suspects of each symptom are ranked. The tool has an option to list the percentage of suspects of each symptom. By default, it only lists the suspects with score higher than 80 or ranked top three of each symptom.

4.2 Characteristics of the Selected SoC

The experiment was performed on a standard SoC product incorporating EDT with bypass mode. Table 1 illustrates the characteristics of the selected product that was manufactured in 130 nm technology. GC_{total} is the total gate count of the device whereas GC_{EDT} is the gate count of the EDT logic alone and in parenthesis the percentage of the total gate count is given. *Scan Cells* is the total number of scan flip-flops, whereas, *Non-scan Cells* is the total number of non-scan flip-flops and non-scan latches. *X Sources* denotes the number of circuit nodes that are considered as sources of observable unknown values by the ATPG tool. *Channels* describes the number of external scan channels, and *Chains* is the number of internal scan chains within the device. L_{EDT} and L_{BYP} are the length of the longest internal scan chain for EDT and bypass modes, respectively.

The design contains about 5% non-scan elements such as non-scan flip-flops and latches. In addition, it also includes a flash memory and a significant number of SRAMs. All internal clock trees were balanced for the scan test and only one external test clock is used.

GC_{total}	1,314,177
GC_{EDT}	8,691 (0.68%)
Scan Cells	64,914
Non-scan Cells	3,037
X-Sources	2,817
Channels	36
Chains	357
L_{EDT}	208
L_{Byp}	2,008
Compression Target (L_{EDT}/ L_{Byp})	9.65X

Table 1: Characteristics of the design

For this design 1800 EDT patterns with a stuck-at coverage of 95% were generated. No special patterns like sequential patterns for SRAM interface testing were used and there was no effort spent in getting high test coverage. An identical set of 1800 uncompressed patterns for bypass mode with exactly the same load and unload values as in the EDT mode were derived from the EDT patterns. Both pattern sets were applied to 70 failing devices sampled from production test failures. Fail data was collected on a tester for the following four different scenarios.

1. Failure file with up to 10K failure cycles using uncompressed patterns.
2. Failure file with up to 256 failure cycles using uncompressed patterns.
3. Failure file with up to 10K failure cycles using EDT patterns.
4. Failure file with up to 256 failure cycles using EDT patterns.

Note that while the failure file with 10K fail cycles may result in better diagnostic resolution and is more suitable for failure analysis, the time penalty incurred in logging that many cycles on the tester may not be acceptable for all test equipment in the context of a high volume monitoring diagnosis flow. Due to that reason failure files with fail cycles limited to 256 were also collected. The objective is to compare the diagnosis result for these scenarios and understand the consequent change in resolution.

4.3 Uncompressed Diagnosis and EDT Diagnosis Comparison Criteria

Since for monitoring diagnosis and failure analysis purposes net information is more relevant, the following comparison metrics are based on net information.

4.3.1 Symptom Match Rules

The following criteria were used to determine if a symptom of uncompressed diagnosis matches with a symptom of EDT mode diagnosis and vice versa.

Match: The top 3 suspects of the compared symptoms are the same. Note that difference in the suspect order is allowed.

No match: The top 3 suspects of either symptom do not show up in the other symptom.

Partial match: All other situations not listed in match and no match categories result in partially matched symptoms.

4.3.2 Device Diagnosis Match Rules

Based on the symptom match rules described above, the following criteria were used to determine if uncompressed diagnosis and EDT diagnosis results match for a particular device.

Perfect match: All symptoms of uncompressed and EDT diagnosis match.

Good match: The uncompressed and EDT diagnosis are considered good match if the match-ratio is equal to or greater than 50%. The match-ratio is defined as total number of matched symptoms plus partial match symptoms with 50% credit divided by greater number of symptoms from either uncompressed or EDT diagnosis.

Bad match: The uncompressed and EDT diagnosis are considered bad match if the match-ratio is less than 50%.

No match: None of the symptoms from the uncompressed and EDT diagnosis match.

5 Results

In this section results from the following five different comparisons of uncompressed diagnosis and EDT diagnosis are presented.

1. Uncompressed diagnosis vs. EDT diagnosis with failures files containing up to 10K fail cycles
2. Uncompressed diagnosis vs. EDT diagnosis with failures files containing up to 256 fail cycles
3. Uncompressed diagnosis with failures files containing up to 10K fail cycles vs. 256 cycles
4. EDT diagnosis with failures files containing up to 10K fail cycles vs. 256 cycles
5. Uncompressed diagnosis with failure files containing up to 10K fail cycles vs. EDT diagnosis with failures files containing up to 256 fail cycles

5.1 Uncompressed Diagnosis (10K failure cycles) vs. EDT Diagnosis (10K failing cycles)

Table 2 below illustrates the comparison results for uncompressed vs. EDT diagnosis with 10K failure cycles.

As can be seen from the table, 97.1% of the dies have matched in either perfect or good manner. Of the 70 total dies, 25 of them had equivalent failure files for uncompressed and EDT scan patterns, while the remaining 45 dies had differences. Failure files contain information to be used by diagnosis such as which patterns and cycles had mismatches. The failure files for a die in uncompressed and EDT mode are considered equivalent so long as the mismatching patterns are the same even if mismatching cycles for these patterns are different as might be the case due to compaction. Notice that it is not uncommon for the failures to be different even when the same set of patterns is applied to a device multiple times. These differences may be the result of inherent variances in the equipment or due to defects such as opens, as illustrated in Figure 7, that cause unstable test results. Thus it is not surprising that the there are differences in the failure files between uncompressed and EDT patterns even though these patterns are equivalent.

Match	#Die	% Of all die
Perfect	53	75.7%
Good	15	21.4%
Bad	1	1.4%
No	1	1.4%

Table 2: Match statistics for 10K failure cycles

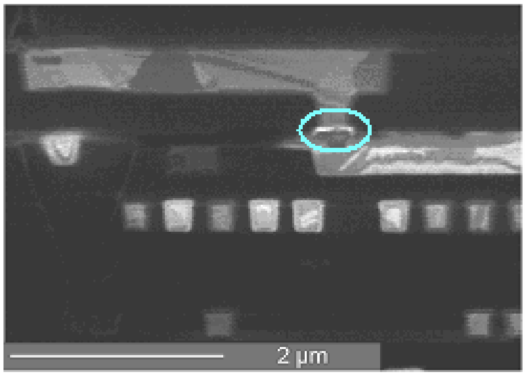

Figure 7: Example of a Void in Via5 identified based on a diagnosis result

Match	Failure File	#Die	% Of all die
Perfect	Match	25	35.7%
Perfect	Mismatch	28	40%
Good	Mismatch	15	21.4%
Bad	Mismatch	1	1.4%
No	Mismatch	1	1.4%

Table 3: Match statistics split according failure file correlation

It is interesting to note from Table 3 that diagnosis results with uncompressed and EDT patterns for all 25 dies with matching failures files have matched perfectly. In fact, in addition, results for 28 dies have also matched perfectly even though their failure files have differences for uncompressed vs. EDT patterns. The 15 dies that resulted in good match as well one die each of bad and no match are all for the dies with differences in failure files. The die that resulted in no match between diagnosis with uncompressed and EDT patterns had 2 and 3 failing patterns respectively, leading to a 30% percent difference of the failing data, impacting the diagnosis result.

5.2 Uncompressed Diagnosis (256 failure cycles) vs. EDT Diagnosis (256 failure cycles)

Table 4 illustrates the match results when the number of cycles in the failure files is limited to 256 with uncompressed and EDT patterns. The 53 dies that matched perfectly with 10K failure cycles stayed the same even when failure files was restricted to 256 cycles. In fact, the only change in result was one die had moved from good match to bad match category.

Match	#Die	% Of all die
Perfect	53	75.7%
Good	14	20%
Bad	2	2.9%
No	1	1.4%

Table 4: Match statistics for 256 failure cycles

It is interesting to note that 36 dies had their failure files truncated. The remaining 34 dies were not affected since for them the number of failure cycles is less than 256. Table 5 below shows the match statistics for the dies that had their failure files truncated.

Match	#Die
Perfect	22
Good	13
Bad	1

Table 5: Match statistics for the 36 dies with truncated failure files

5.3 Uncompressed Diagnosis (10K failure cycles) vs. Uncompressed Diagnosis (256 failure cycles)

This comparison illustrates the impact of truncation of failure file to more practical limit such as 256 cycles, as needed for high volume monitoring diagnosis, would have on the diagnosis result.

Match	#Die	% Failing pattern reduction (average)	% Of all die
Perfect (w/o truncation)	34	0%	48.6%
Perfect (with truncation)	24	57.4%	34.3%
Good (with truncation)	6	80.6%	8.6%
Bad (with truncation)	6	94.2%	8.5%

Table 6: Match statistics for 10K vs. 256 fail cycles

From Table 6 we can see that 91.5% of the dies has resulted in either perfect or good match even with 256 failure cycles. The third column in Table 6 supplies the average percentage reduction in the number of failing patterns recorded in the failure file due to the 256 cycle limit.

5.4 EDT Diagnosis (10K cycles) vs. EDT Diagnosis (256 cycles)

This comparison illustrates the impact truncation of failure file to more practical limit, such as 256 cycles, would have on the diagnosis result for EDT diagnosis. From Table 7 we can see that 91.3% of the dies have resulted in either perfect or good match even with 256 failure cycles. The comparison with section 5.3 shows that the impact of truncation is similar for EDT diagnosis and uncompressed diagnosis. In either EDT or uncompressed mode 6 dies are badly diagnosed.

Match	#Die	% Fail pattern reduction (average)	% Of all die
Perfect (w/o truncation)	34	0%	48.6%
Perfect (with truncation)	23	59.2%	32.7%
Good (with truncation)	7	73.8%	10%
Bad (with truncation)	6	94.1%	8.6%

Table 7: Match statistics for 10K vs. 256 fail cycles

5.5 Uncompressed Diagnosis (10K cycles) vs. EDT Diagnosis (256 cycles)

For completeness the final comparison is between the two extreme scenarios, i.e., uncompressed diagnosis with 10K fail cycles versus EDT mode diagnosis with failure file limited to 256 as targeted for high volume monitoring diagnosis flow.

Match	#Die	% Of all die
Perfect	50	71.4%
Good	10	14.3%
Bad	9	12.9%
No	1	1.4%

Table 8: Match statistics for uncompressed 10K fail cycles vs. EDT 256 fail cycles

Instead of the expected 8 badly diagnosed dies, 2 from the compression and 6 because of truncation, a total of 9 devices were diagnosed badly. The one additional device is because of the hard decision limit for good and bad devices which cause border crossing effects. As can be seen from Table 8, about 86% of the dies resulted in either perfect or good match. Especially the perfect diagnosed dies seem to be resistant to truncation and compression. In spite of the failure cycle restriction EDT diagnosis has matched with the best-case uncompressed diagnosis for a majority of the dies. Nevertheless the increase by a factor of 5 from 2 to 10 bad and no match results is important and must be subject of further improvements.

6 Conclusions

The results and analysis above have shown that EDT mode diagnosis performs well when compared to diagnosis with uncompressed patterns. This is indeed the case even when truncating the failure cycles from 10K to 256 as would be needed for a feasible high volume monitoring diagnosis flow.

For the case where EDT and uncompressed diagnosis were compared based on failure files with 10K fail cycles, 53 out of 70 dies (about 76%) had perfectly matching diagnosis results, while 15 out of 70 dies (21%) had good matching diagnosis results. In other words, for 68 out of 70 dies (about 97%) perfect or good matching behavior has been observed between EDT and uncompressed diagnosis. When the failure cycles are restricted to 256 cycles for EDT mode and the diagnosis results are compared with the extreme scenario for uncompressed diagnosis (with 10K failure cycles), 86% of the dies had either perfect or good matching diagnosis result. These results are encouraging and demonstrate the feasibility of a monitoring EDT diagnosis flow serving high volume test. Efforts are currently on to proceed with setting up a high volume diagnosis flow based on the EDT technology.

7 Future Work

Given the good experimental results the next step to implement a monitoring diagnosis flow will be to increase the sample size of diagnosed devices. The results obtained in this way will be statistically analysed e.g. by methods described in [8] regarding geometric concentrations of defects or increased fail rates of modules and library cells. Applying statistical analysis will result in identifying yield detractors. Subsequent physical failure analysis will give additional feedback on the localization quality of EDT diagnosis and will enable continuous improvement of the presented methods.

The main bottleneck of the volume diagnosis flow is the insufficient data capturing capabilities of present test equipment. The approach to reduce the amount of fail information, which is stored per device for subsequent fault diagnosis, leads to an unnecessary loss in diagnosis quality, as observed in the experiment; the overhead test time for failure logging remains important. To overcome this limitation is the task of the test system vendors. Future work on the diagnosis software platform has to cover significant reduction in runtime to achieve our challenging throughput goals. And as the primary target the localization capabilities of given diagnosis algorithms need to be enhanced by taking more sophisticated fault models into account.

8 Acknowledgement

The authors would like to thank Andras Kun for bringing in his expertise in test engineering and enabling the rapid execution of the experiment on standard ATE equipment.

9 References

1. International Technology Roadmap for Semiconductors, 2004
2. R. Madge, “New test paradigms for yield and manufacturability”, *Proc. ITC*, pp. 13-13, 2004
3. J. Rajski, M. Kassab, N. Mukherjee, N. Tamarapalli, J. Tyszer, and J. Qian, “Embedded deterministic test for low cost manufacturing

test", *IEEE Design & Test of Computers*, pp. 58-66, Vol, 20, Issue 5, Sept.-Oct. 2003

4. F. Poehl, M. Beck, R. Arnold, P. Muhmenthaler, N. Tamarapalli, M. Kassab, N. Mukherjee, J. Rajski, "Industrial Experience with Adoption of EDT for Low-Cost Test without Concessions", *Proc. ITC*, pp. 1211-1220, 2003
5. J. Rajski, J. Tyszer, M. Kassab, N. Mukherjee, "Test pattern compression for an integrated circuit test environment," USA patent, Serial No.6,327,687, December 4, 2001
6. D. Appello, A. Fudoli, K. Giarda, V. Tancorre, E. Gizdarski, and B. Matthew, "Understanding yield losses in logic circuits", *IEEE Design & Test of Computers*, pp. 208-215, Vol, 21, Issue 3, May-June 2004
7. C. Hora, R. Segers, S. Eichenberger, M. Lousberg, "On a statistical fault diagnosis approach enabling fast yield ramp-up, *Proc. of Euro. Test Workshop*, pp. 193-198, 2002
8. H.-P Erb, C. Burmer, A. Leininger: "Yield enhancement through fast statistical scan test analysis for digital logic", Advanced Semiconductor Manufacturing Conference (ASMC) 2005
9. C. Burmer, P. Egger: "Software aided Failure analysis using ATPG tool"; International Conference on the Physical and Failure Analysis (IPFA) Proceedings 2001
10. B. Koenemann, "LFSR-coded test patterns for scan designs", *Proc. Euro. Test Conf.*, pp. 237-242, 1991
11. F. Hsu, K. Butler, J. Patel, "A case study on the implementation of the Illinois scan architecture", *Proc. ITC*, pp. 538-547, 2001
12. I. Bayraktaroglu, A. Orailoglu, "Test volume and application time reduction", *Proc. DAC*, pp. 151-155, 2001
13. S. Mitra, K.-S. Kim, "X-compact: An efficient response compaction technique", *IEEE Trans. On CAD of ICs*, pp. 421-432, Vol. 23, Issue 3, March 2004
14. W.-T. Cheng, K.-H. Tsai, Y. Huang, N. Tamarapalli, and J. Rajski, "Compactor independent direct diagnosis", *Proc. of Asian Test Symp.*, pp. 15-17, 2004
15. J. Waicukauski and E. Lindbloom, "Failure Diagnosis of Structured Circuts", *IEEE Design & Test of Computers*, pp. 49-60, Vol. 6, Issue 4, 1989
16. T. Bartenstein, D. Heaberlin, L.Huisman and D. Sliwinski, "Diagnosing Combinational Logic Designs using the Single Location At-A-Time(SLAT) Paradigm", in Proc. Intl. Test Conf., 2001, pp. 287-296
17. D. B. Lavo, I. Hartanto, T.Larrabe, "Multiplets, Models, and the Search for Meaning: Improving Per-Test Fault Diagnosis," in Proc. Intl Test Conf., 2002, pp.250-259
18. Z. Wang, K.-H. Tsai, M. Marek Sadowska, and J. Rajski, "An Efficient and Effective Methodology on the Multiple Fault Diagnosis", in Proc. Intl .Test. Conf., 2003, pp. 329-338