SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 2, No. 1, May 2005, 43 - 55

Advances in VLSI Testing at MultiGb per Second Rates

Dragan Topisirović[1]

Abstract: Today's high performance manufacturing of digital systems requires VLSI testing at speeds of multigigabits per second (multiGbps). Testing at Gbps needs high transfer rates among channels and functional units, and requires readdressing of data format and communication within a serial mode. This implies that a physical phenomena-jitter, is becoming very essential to tester operation. This establishes functional and design shift, which in turn dictates a corresponding shift in test and DFT (Design for Testability) methods. We, here, review various approaches and discuss the tradeoffs in testing actual devices.

For industry, volume-production stage and testing of multigigahertz have economic challenges. A particular solution based on the conventional ATE (Automated Test Equipment) resources, that will be discussed, allows for accurate testing of ICs with many channels and this systems can test ICs at 2.5 Gbps over 144 cannels, with extensions planned that will have test rates exceeding 5 Gbps.

Yield improvement requires understanding failures and identifying potential sources of yield loss. This text focuses on diagnosing of random logic circuits and classifying faults. An interesting scan-based diagnosis flow, which leverages the ATPG (Automatic Test Pattern Generator) patterns originally generated for fault coverage, will be described. This flow shows an adequate link between the design automation tools and the testers, and a correlation between the ATPG patterns and the tester failure reports.

Keywords: Built in Self Test, VLSI testing, Design for Testability, Stuck-at fault, MultiGbps

1 Introduction

Currently produced digital system's being of exceptionally high performance demand a testing of VLSI circuit at rates of Gb per second. In recent years, we are witnessing significantly fast growth of new techniques for testing of VLSI circuits and systems that give high quality and fast testing times.

[1]Faculty of Electrical Engineering, A. Medvedeva 14, 18 000 Niš, Serbia and Montenegro,
E-mail: centar@medianis.net

High density, core-based ICs have significant popularity, although complexity of these chips can slow down development and increase cost rather than enable high performance and profit margins in manufacturing. Today's economy and the rising role of new technologies, and expending costs for development of new products, are forcing the electronic industry to re-examine the existing approaches to design and test. For new product, the development of new technological environments promises to provide productivity increases and fastest time to market, while keeping costs under control. Although, testing and debugging these devices represents very difficult problems, the new economy and modern industry recognizes that testing costs are escalating faster than other costs related to the development phase.

1.1 Testing at Gbps Rates

Testing at Gbps rates is necessary to overcome between traditional techniques, which rely extensively an ATE, and the technology improvements in ICs and their high clock rate. This requires radical changes in the organization of the test as well as innovative and practical solutions to the support equipment. These changes have a profound impact on many aspects of existing test techniques. For example, allowing high transfer rates among channels and functional units, such as in the I/O definition of a SoC, requires readdressing the implication of data format and communication within a serial mode. This contains feature into a shell, that physical phenomena, such a jitter, are becoming very relevant to tester operation. It is today focus of all of these issues that makes multigigahertz testing a challenging problem in today's test technology.

1.2 Specifications of Testing

We pay attention into two aspects of testing:

- The first part, which includes "Testing Gbps Interfaces without a Gigahertz tester" and these relations represent new approaches and frameworks that enable testing of multigigahertz digital devices with or without a modified ATE. There are novel testing problems - called the source synchronous interface. The proposed technique relies heavily on DFT (Design for Testability) and in particular use a new methodology called AC I/O loop-back. This technique represents a significant improvement over a simple I/O loop-back arrangement. This technique allows the measurements of multiple functional parameters inclusive of AC timing specifications. Own example represents application of AC I/O loop-back and supporting DFT circuitry for the Processor Intel Pentium 4, showing that their technique can efficiently correlate different stress measurements at the physical layer within a self-test framework. A combination of timing stress and voltage stress generate diagrams with no need for a high-speed tester.

- The second part, "Multiplexing ATE Channels for Production testing at 2.5 Gbps", analyze testing at multigigahertz using a different technique, namely to multiplex ATE channels for production testing. Several features of current-generation ATE-timing calibration, modularity, temperature effects for sampling logic, and the large number of high channels - all shows the need for multiplexing. There are two variants for testing using new multiplexer circuit to accelerate the speed up to 2.5 Gbps. First variant uses differential pair signals in a arrangement with embedded ATE circuitry to support accurate timing calibration albeit jitter makes it prone to timing errors. The second variant reduces the negative influence of jitter on test operations. This type of design is expected to ensure high Gbps rates in future systems.

1.3 Automated Test Equipment, Economics of Test

The economics of test, especially in a case of test equipment need in particular, has received significant attention from many vendors and ATE manufacturers, customers of ATE and the research community at large. ATE is shown in Fig. 1. Increasing cost of ATE, increases the price of the product. Features, such as multisite organisation, architecture modularization, and the increased presence of inexpensive testers such as those included in BIST techniques (BIST - Built In Self Test, Fig. 2) are some of the significant developments of recent years. A combination of BIST and ATE represent a possible alternative to speeding up test application time.

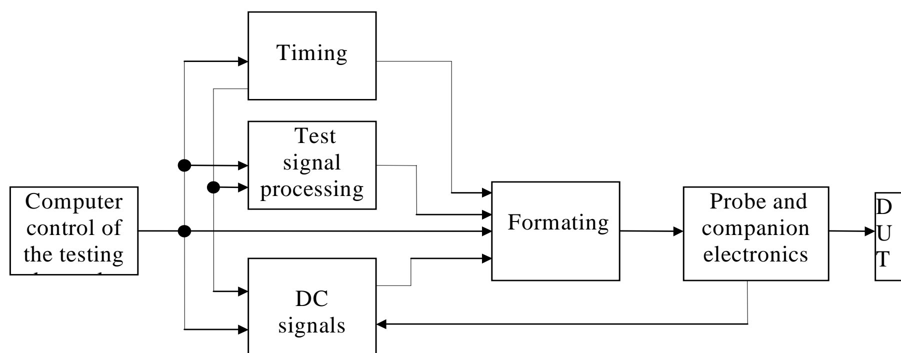

Fig. 1 - *Architecture of an electronic tester, ATE.*

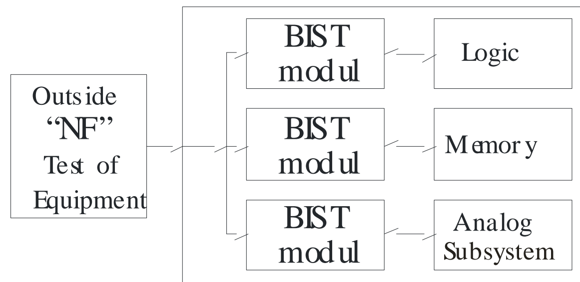

Fig. 2 - *Architecture of chip with built in self-test.*

1.4 Equipment for Testing

General block scheme of ATE, represented in Fig.1 [1-2]. The tester contains the following components:

- the computer system used for "testing programming", the electronic subsystem enabling the synchronisation, waveform generation, timing, formating;
- probe and companion electronics; and
- computer control of testing.

Today, many producers of such testing devices exist. Depending on configuration one tester of high performance may cost a couple of milions dollar and more [3]. High quality probe and catcher, cost up to half milion dollar [3, 4]. If we include costs working premises, electrical installation and working staff, it is easy to come to a conclusion why testing is exspensive business.

All of ATE must provide the following:

1. Condition and impulse:
    - power supply and ground;
    - output, incoming signals;
    - to adapt signal on site of reset impuls and consumer.
2. Measurements:
    - impedance on input pin,
    - threshold of logical level input digital signal
    - generation voltage on input pins
    - time of establishment at front and back edge of signals
    - propagation of delay
    - speed working
    - output signals

3. Extraction:
    - Adequate DC features
    - Adequate AC features
    - Functioning properly of logical function
    - Exact speed of work
    - Correct characteristic of signals.

Corresponding elctronics, that is a board for interface with DUT, (Device Interface Board, DIB), represents electrical interface between ATE and DUT. There are various form and size of DIB, but their common functions are to provide reliable and uncomplicated separable electrical interface between DUT and electrical instrument of the testers.

VLSI testers, that are available on the market, may satisfy different needs. In contrast to PCs, testers have no standard architecture. Every producer of testing equipment is trying to apply some unique performance in order to compete with rival producers. Different producers of testing equipment build their own S/W platform for testing. Moreover, test routines developed for one type of tester frequently are very difficulty to translate for use on other testers because of H/W optimization to testers. Fortunately, the majority of automatic testers have many common functions and features.

2 Diagnosis Using ATPG

ATPG tools provide several advantages for testing scan-based designs. The advantage is high fault coverage with minimal human effort and the benefit is automated diagnosis. Traditional diagnosis requires users to write more functi-onal vectors to isolate failures, and today's scan-based designs, ATPG-based diagnosis tools are aware of internal states and can use these data to locate defects precisely.

When scan based patterns fail on the tester, a failure log stores the failure information for diagnosis. This log contains failing patterns, outputs and scan shift cycles. The failure file format is a property for each ATPG tool vendor, although the IEEE Std. 1450.1 working group is trying to standardize this format.

The diagnosis tool tries to match the observed behavior on the tester with the fault model simulation and analysis. The diagnosis results, or callouts, include a list of pins that best explain the observed behaviour. Then we can map the logical pins to topological locations in the layout.

One way to increase confidence is to regenerate additional patterns to characterize defects with more accuracy. ATPG tools have N-defect ability which allows generating a pattern set so that the ATPG tool detects the fault set N times. The callouts from diagnosis can serve as a fault set for an N-detect

ATPG. We can run the additional patterns on the defective part and further analyze the new failure log. If the diagnosis is consistent, the probability of an accurate diagnosis is higher.

2.1 Collecting ATPG test data using manufacturing ATE in production

A scan-based diagnosis flow requires the EDA-ATE link to be a real bi-directional connection, going from the EDA-DFT environment to the ATE, and vice versa. This last reverse path is critical. The requirement is to collect the data indicating the location of the mismatch on the ATE. It is important to ensure a complete, coherent correlation between the original ATPG pattern set and the tester failure report.

A set of tools and flow, which are implemented on different manufacturing ATE consist of:

- creating the failure database;
- setting up the ATE error memory to collect as many failures as possible;
- setting the test conditions;
- running the test and collecting sequence failures; and
- translating the ATE cycle-based mismatch into a scan-cell-based format.

The goal is to collect ATPG test failure data during the production phase and thus allow full traceability between any tested die and its coreresponding failure data collection. Two main issues are in this ATPG failure data collection:

- the current de facto standard, the Standard Test data Format (originally developed by Teradyne), is inconvenient for storing ATE failures;
- the failure data collection process presents overhead in the testing time;

ATE platform is specific and depends on the time required to continue the run rather than stopping at the first failure. The other contributor to the overhead is the time required to save the failure data to disk.

2.2 Fault–simulation-based diagnosis

The efficiency of logic diagnosis depends on the target fault models. In this case, the diagnostic algorithms acts on the assumption that, on a per-pattern basis, many realistic defects behave as stuck-at faults. A set of stuck-at fault candidates can represent many of these defects. Diagnostic analysis begins with all failing patterns unexplained. For each failing pattern, a backward path-tracing procedure based on the good mashine logic values derives a list of potential fault candidates consistent with the failing measures. The next step is the fault list pruning. We consider that a set of fault candidates explains a failing pattern if all failing measures exactly match with all simulation failures. The output of logic diagnosis is a set of defects, each explained by a set of failing patterns.

Two techniques are used to increase the accuracy and precision of logic diagnosis as well as ability of the diagnostic tool to analyze complex and multiple defects using stuck-at and transition fault simulation for fault list pruning. The first tecnique correlates the behaviour of same predefined defect types, or ***basic types***.

The basic types are:

- stuck-at (S);
- transition (T);
- bridging (B); and
- net (N).

To classify a fault candidate as a stuck-at or transition fault, the original stuck-at or transition fault should explain some failing patterns and pass all passing patterns. Transition faults require a certain transition on the fault site for all failing patterns. Classifying a fault as a bridging fault requires that the representative stuck-at fault explain a subset of the failing patterns and is a potential aggresor for the remaining failing patterns. Classifying a fault candidate as a net fault requires that the final diagnosis report includes at least one additional stuck-at fault candidate as a different fan-out branch of the same stem. This approach maximizes the diagnostic tool's ability to simultaneously derive fault candidates for all basic defect types while independently minimizing the number of potential fault candidates for each defect type. Fig. 3., shows an example format for the diagnostic report. In this case, we can conclude with a high level of confidence that a net-type defect exists in the stem of fan-out branches A and C.

The second technique is based on the iterative nature of diagnosis and focuses on increasing accuracy for multiple defects. The diagnostic algorithm is a multiphase procedure that is used to derive the high confidence defects during the first pass. After this pass, the diagnostic algorithm updates the failing measures for all unexplained failing patterns based on the already-extracted defects. All passes after the first one use less-restrictive constraints for faultslist pruning. The goal is to extract additional information from the unexplained failing patterns, which might explain some multiple defects or complex defects that don't behave as stuck-at faults. An analysis based on cones af logic within the circuit and backward path tracing is used to distinguish unrelated failing measures.

2.3 Using diagnosis results

The first effort involves finding a tradeoff between the equipment test time and the diagnostic tool's accuracy and precision.

Defect 1
Explained failing patterns:3, 7, 9 and 13
Fault 1: <defect types: ST*N, polarity, pin A, cell type>
Fault 2: <defect types: ***T****, polarity, pin B, cell type>

Defect 2
Explained failing patterns: 4 and 9
Fault 1: <defect Types: S**N polarity, pin C, cell type>

Fig. 3 - *Tipical diagnostic report, indicating fault candidates (pins) and different fault types: stuck-at (S), transition (T), bridging (B) and net (N).*

As Fig. 3 shows, for each defect, the typical diagnosis report highlights a list of fault candidates (pins), the corresponding cell types, and the associated behaviour explainning a set of test patterns. This approach for extracting a list of the potential cells responsible for the failures has two main goals [5]:

- Indentification of the existing sources of design marginality;
- The critical process steps for the design.

The existence of an essential defect in each lot can cause a small yield loss. The key is to unscramble the essential process defect from the repetitive failure mechanisms caused by design or layout marginalities that are not easily detectable otherwise.

Classifying electrical defects requires performing several postprocessing analyses of the diagnosis report. This postprocessing weights the statistics with respect to several parametars, such as library cell area and number of instances in the design. Finally, the power supply subsystem is measuring the supply current, the so called I_{DDQ} which in some testing techniques has a decisive role. We also use I_{DDQ} measurement to help classify defects. I_{DDQ} is used often in diagnostical purpose too. This test flow uses ATPG vectors to take I_{DDQ} measurements on qualified strobe points. The digital test of I_{DDQ} is the DFT method intendend to uncover an elegant (catastrophic, more exactly parametrics) defects in digital circuits. It observes behaviour of CMOS circuits to stationary discipline and measurements of very small current between power supply and grounding, [1], [6]. Any change of I_{DDQ} value from the expected one shows at defect to resistance that can but must not be catastrophic, as shown in Fig. 4.

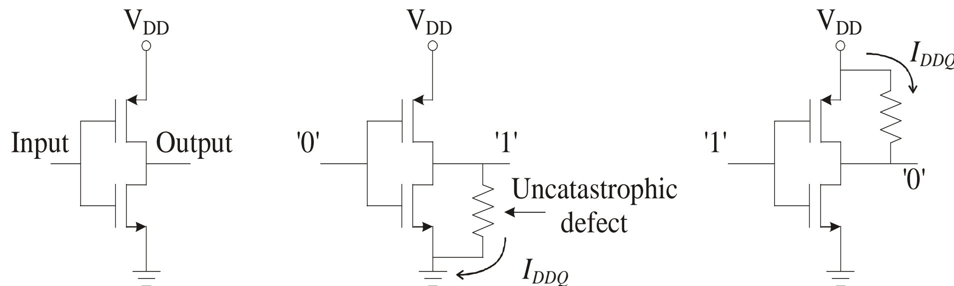

Fig. 4 - *Leakage of I_{DDQ} in a CMOS inverter. a) fault free circuitt, b) leakage between output and ground node, and c) leakage between output and supply node.*

2.4 Correlation with optical inspection data

To avoid hours of work in analyzing all failures, we developed a methodology to correlate electrical failures to particular process steps [5]. In this step, we used methodology that dedicated inline defectivity inspections during fabrication. Comparing the wafer imagines with the layout, we can use a third-party tool to identify abnormalities.

Correlation involves translating the (x,y) coordinates of the callouts found with the diagnosis tool and overlaying these results with inline-inspection data maps stored in a dedicated database. Inline inspection data is available for only a limited number of wafers, but it is usually representative of the entire lot. Overlaying process-and electrical-defect data helps identify any process-defects causing the electrical failure. Every step, in which the defects found through optical inspection overlap the callouts from the diagnosis tool to within a certain tolerance, we declare the defect as a HIT. With high probability, this indicates, that the physical defect is at the callout location and is the actual cause of the ATE failure. In other words, we have found the «killer defect».

2.5 Understanding test-mode functional marginalities

Several factors will further contribute to mutual understanding. We discuss yield losses determined by marginalities in the functionality of the chip uner test. These types of factors often influence yield in various ways and we associated yield variation with process variation. If there are parameters present outside an acceptable range, that affects yield.

The key of this analysis is to understed systematic marginalities that might unpredictably affect the yield. The proposed flow leverages well-known techniques such as SHMOO plots, which can be used to assert the behaviour of a chip with respect to a given test pattern set when test condition such as power supply voltage, temperature and timing are varied. Usually, shmoo plots are represented using 2D or 3D charts. Each test result is reported with green and

red boxes to identify passes and failures of the given pattern set [7]. Fig. 5 shows an example. This methodology uses DFT -aware shmoo plots, in which we vary parameters determining test conditions according to the DFT solutions in place. DFT metholology today is very popular and paymentable application concept to projecting for testability. The testability understanding feature of circuit can be tested. Design For Testability makes automatization of testing possible.We assume that mismatches obtained through these variations may point out the critical regions on the die that are more marginal or loss robust. We can guess that yield killing factors stem from a lack of robustness in a specific topological region of the product design.

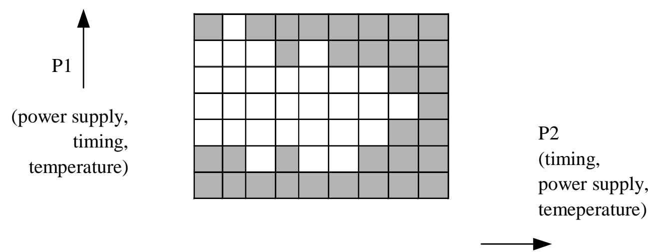

Fig. 5 - *Example shmoo plot for two test patterns, P1 and P2, with example parameters. The gray squares represent the last failures, and the white squares mean the test in for that specific condition passes (that is, there are no mismattches).*

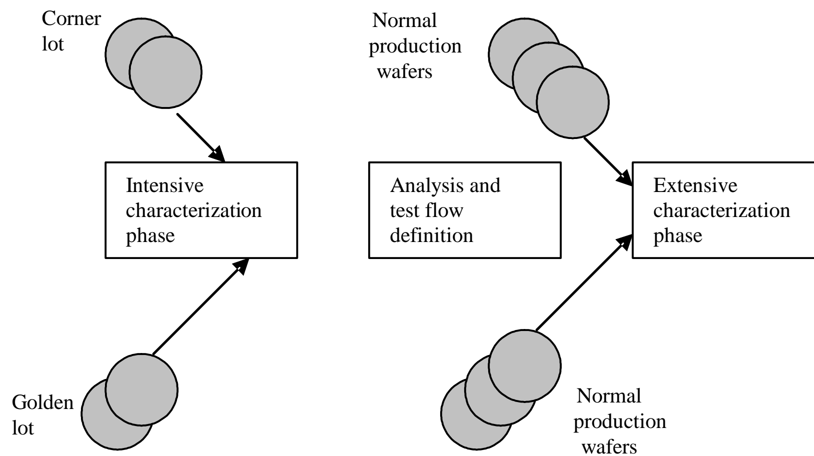

Fig. 6 - *Proposed characterization flow. The stacks of circles represent silicon wafers. A corner lot is a set of wafers processed through the intentional spreading of one or more process parameters. A golden production lot is a set of wafers processed under nominal conditions.*

Fig. 6 shows that this flow has two different phases:

- The first phase identifies the best flow and conditions for the diagnosis This phase is *Intensive*; this phase exhaustively exploits test condition and collects and processes all data.
- The second phase targets the production. This phase is *Extensive*; this phase involves carefully managing test time overhead and exploring test conditions on a statistical basis.

In between these two flows are the analysis an the test flow definition.

The test failure data collected from these flows makes possible analyses that go beyond identifyng defects in the circuit's correct operation. The systematic capture of the device behaviour at the margins of its operating conditions and the consequent analysis with the other methods can identify marginalities that cause systematic failures.

2.6 Experimental results

The goal is to check the diagnostic tool's ability to locate the basic defect types and to minimize the number of initial fault candidates (potential locations) for consideration during diagnosis. The advantages of simulation over silicon-base experiments are numerous. Simulation's quickness and lower cost let us conduct many experiments to tune the algorithms. It used 10 full-scan industrial circuits and ran 1.000 experiments for each defect type. The diagnosis algorithm is accuracy for simple defect types (single and multiple stuck-at faults and single transition faults) was in the 98%. For more complex defect types such as bridge faults the accuracy was in the 90%. Thus, the algorithm initially satisfied the necessary conditions of having high accuracy for real physical defects when a good correlation existed between the selected fault model and the behaviour of real physical defects.

2.7 Efficiency diagnosis

There are some methods that explain effectiveness of diagnosis flow. One method includs considering number of times that the algorithm explained the failing patterns and one another method of a ranking, including cases in which the diagnostic tool detected a single candidate relatively more often than on failing devices producing multiple candidates.

The other method of the diagnosis flow's effectiveness is the statistical process. In this case we consider the largest possible failing population. We are also limited by the number of mismatches and we could log for each failing

sample. We did not expect any negative effects from considering cases only in which diagnosis returned a maximum of two callouts and this method has remained successful while keeping the solution manageable. This number depended on the test time common that production could tolerate and more importantly, the architectural limitation of production ATE in logging mismatches, **Table 1**.

Table 1

One example of efficiency results for diagnosis flow.

Category	No. of parts
Total for diagnosis	12.645
Those with	
-all failing patterns explained	12.050
-one diagnosed callout	4.830
-two diagnosed callouts	595
-partially complete diagnosis	7.220

This diagnostic process can achieve a reasonable success ratio in locating various manufacturing defect. For example, one stuck-at defect demonstrates as a short in the poly, and second transition defect effects only the circuit's speed performance. The systematic analysis of several production lots of wafers using proposed method and flow shows how to identify design marginalities. For example, narrow poly in the cell, which occured during etching negatively affected to circuit timing. The complete analysis flows included an extrapolation of parameters and remodelling to more accurately simulation of the affected circuitry. Resimulation allowed a further assessment of the failures. The main effect is the intrisic impossibility of pinpointing the yield-killing factors that eliminate yield. The key step is forecast for yield improvement quality. We are considering the need to correlate more data sources and one of the main goals we are focusing is using layout area checks to address design marginalities.

3 Conclusion

Research results shown are related to the problem of testing and diagnosis of digital electronic circuits operated at very high frequencies. Problems related to short transition times were discussed first. Then impact on testing technology was considered including the ATE performance. Accordingly, new design architectures were discussed enabling design for testability at GHz. Finally specific problems related to diagnosis of digital circuits were discussed and experience demonstrated.

4 References

[1] V.B. Litovski: CAD of electronic circuits, DIGP "Nova Jugoslavija", Vranje, 2000 (in Serbian).

[2] M. Baker: Demystifying Mixed-Signal Test Methods, Elsevier Science, USA, 2003.

[3] M. Burns, G. W. Roberts: An Introduction to Mixed-Signal IC Test and Measurement, Oxford University Press, New York, 2001.

[4] B. Davis: The Economics of Automatic Testings, McGraw-Hill Book Company, London, 1982.

[5] C. Hora et al.: An Effective Diagnosis Method to Support Yield Improvement, Proc. Int'l Test Conf. (ITC 03), IEEE, Press, 2002, pp. 260-269.

[6] E. Isern, J. Figueras: IDDQ Test and Diagnosis of CMOS Circuits, IEEE Design and Test of Computers, Vol. 12, No. 4, Winter 1995, pp. 60-67.

[7] K. Baker, J. van Beers: Shmoo Plotting: The Black Art of IC testing, IEEE Design & test, Vol. 14, No. 3, July-Sept. 1997, pp. 90-97.