From Embedded Test to Embedded Diagnosis

Hans-Joachim Wunderlich
Universität Stuttgart
wu@informatik.uni-stuttgart.de

Abstract

Testing integrated circuits with millions of transistors puts strong requirements on test volume, test application time, test speed, and test resolution. To overcome these challenges, it is widely accepted to partition test resources between the automatic test equipment (ATE) and the circuit under test (CUT). These strategies may reach from simple test data compression/decompression schemes to implementing a complete built-in self-test. Very often these schemes come with reduced diagnostic resolution.

In this paper, an overview is given on techniques for embedding test into a circuit while still keeping diagnostic capabilities. Built-in diagnosis techniques may be used after manufacturing, for chip characterization and field return analysis, and even for rapid prototyping.

1. Introduction

The economic relevance of diagnosis of microelectronic circuits is rather often overlooked. While boards, multi-chip modules, and even systems in a package may be repaired, the repair of systems-on-a-chip is still the exception. But even without a repair option, diagnostic capabilities are essential in the lifecycle of a micro-electronic system.

A) Field: Maintenance and repair may form significant costs not only for the user but also for the chip provider, if warranty costs are included or if customer satisfaction is concerned. Self-repairable components will reduce these costs significantly, especially if the repair is remotely initiated. In any case, the defective module has to be located either for replacement or for repair.

Even if repair is not foreseen, fault and failure analysis on returns will help finding the root causes and avoiding future flaws. Identifying process and design weakness or environmental causes for failures are essential for product improvement and cost reduction.

B) Manufacturing: "Time-to-volume" and "time-to-market" are keys for the economic success of a product. For obtaining times as short as possible, it is not sufficient to get just the first silicon working. The design has to be manufacturable and has to fit to the process.

"Yield ramping" is the task to adapt both the product and the process and requires thorough root cause analysis of failures and outliers [Hora02a,b]. Moreover, not just single faulty chips but the entire volume manufactured has to be observed continuously in order to find process weakness and possible problems as early as possible. The large amount of data generated during manufacturing testing is subject of subtle statistical analysis from the area of data mining.

C) Chip characterization: The first silicon is subject of diagnosis and debug, of course. Comprehensive measurements are performed for validating functional and parametric specification compliance. In the case of failure, root causes have to be identified. An additional difficulty comes from the fact that we are facing design errors, design and process weaknesses and spot defects at the same time. Silicon debug has to identify logical and timing faults as well as their structural and geometric location. Leakage current, hot spots, or threshold voltages have to be measured and regions have to be identified where they may be out of range. Since in nanometer technology we have to expect an increasing variation of device parameters, there will be no sharp separation of devices within or out of the specification [Bork03]. Functionality depends on the robustness of entire modules and chip regions, which in turn is a variable parameter. Hence, observed and evaluated regions are getting larger and complexity of diagnosis increases further.

D) Rapid prototyping: Systems-on-a-chip are hard to validate just by software simulation. Emulation machines and hardware-accelerated simulation reduce validation time down to a fraction and are most effective if the device under test and the testbench are synthesized simultaneously on the machine.

The basic technology is either multi-processor or FPGA-based, in both cases special care has to be taken for the observability of internal signals. For emulation machines, special FPGA-structures are available with increased observability, but in general these features are not sufficient for complete fault location. Scan design, test point insertion and all the other means for enhancing diagnosability may be reused already for design validation [Chen99, Lude04].

In summary, the important role of diagnosis at all the different stages of the design and lifecycle of a circuit leads to increased efforts in developing efficient and effective diagnosis methodologies in both hardware and CAD software. In the next section we discuss the type of flaws which are subject of diagnosis. In section 3, the diagnostic resolution of test patterns generated by standard BIST and embedded test methods are discussed. Section 4 deals with methods to encounter the information loss of the output data compression.

2. Locating faults and defects

2.1. Definition and concepts

In this section, a few definitions are given for avoiding misunderstandings and misconceptions. After that, the main flaws to be diagnosed are introduced and appropriate test generation is discussed.

A *failure* happens, if a system or component does not perform the function specified and expected by the user. A failure may be seen at all levels of the design, a single transistor may fail as well as the entire processor.

An *error* is related to the information produced by a component. A wrong bit or word found on a bus or in a memory are errors, and an incorrect check-sum or signature indicate errors, too.

While a failure deals with a real component of the system and an error with the actual information, a *fault* is related to a model rather than directly to reality. A fault may be defined at all the abstraction levels of a design. A fault of a certain model is a hypothesis, the corresponding test is an experiment, and if the circuit gives the correct output, the hypothesis is falsified.

A *defect* is the physical cause of a fault in the chip material. It is mainly a location with either missing material, additional material or the wrong material. It should be pointed out that not all errors can be connected to defects, for instance single event upsets (SEU) are transient errors due to an external source.

Fault diagnosis is the process of both, detecting and locating the fault at the various levels down to the real defect. Usually, the logic behavior of the design has been validated during simulation, verification and rapid prototyping. But numerous parasitic and timing effects may show up in the first silicon, identifying them is part of *silicon debug*. Hence, diagnosis is more related to defects and debug is closer to design errors, i. e. errors of the designer. However, there is a large overlap in between dealing with yield ramping and design for manufacturability.

2.2. Fault models

For test and diagnosis, fault models play an important role as it is not feasible to generate tests for an arbitrary faulty behavior. Fault models at the various abstraction levels restrict the complexity of test generation, on the other hand, they also reduce resolution. Faults of a certain model and abstraction level may not have a counterpart at lower levels or in reality. For instance, there may be a gate level model of a design with a corresponding fault, but actually, the layout was generated by a single pass synthesis directly from register transfer level. On the other hand, some defects may not be modeled at all.

In nanometer technology, one must be aware that the parametric and functional properties of each single gate, transistor or line may vary within a large range of values [Bork03]. *Specification compliance* testing tries to select a module and to check whether its properties are still within the allowed range despite the variations of its devices. If such a module is too large, compliance testing will not be feasible, but if it is too small, the test outcome may be invalidated into a false accept or false reject by the variations of the environment.

Path delay fault testing may be considered as a special case of compliance testing, where all the gates of a certain path are varying within the allowed range, but the path is still slow [Majh03, Padm03]. Applying a non-robust test may reduce yield, and the restriction to a robust test will provide insufficient fault coverage. Hence, statistical methods for both, fault and circuit modeling are required to obtain diagnostic test patterns [Krst03a, Krst03b, Krst03c, Huan04, Hora02, Hora02b]. *Gate delay faults* assume too long rise and fall times of a gate output. In any case, diagnostic patterns for delay faults require pattern pairs, one for initializing the suspected gate output, and the second one for moving it to the opposite value and observing this change at an output.

A special type of delay faults is formed by so called *cross talk faults*. This fault model describes the capacitive coupling of two lines. If these lines are switched in parallel to opposite values, the aggressor line may finally slow down the signal change of the victim line. Here, a diagnostic pattern pair must first initialize two nodes and, second, observe the transition speed of the victim node. Most often these tests are required during chip characterization and are part of silicon debug, since these faults mainly exhibit a design flaw [Metr00]. But also after manufacturing variations may cause cross talk effects on some dies to be diagnosed.

A more direct coupling of two lines is formed by *bridging faults*. Additional material or missing insulators connect two lines a and b, and a standard fault model maps this fault to wired-AND or wired-OR structures [Ches95, Venk97].

To implement a more realistic model, resistive bridges are considered, too [RHB95]. Figure 1 shows the resistance R_{sh} introduced by some defect which will change the voltage on both nodes a and b.

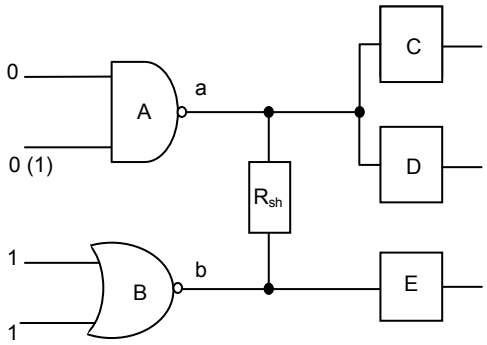

Figure 1: Resistive bridge

Depending on the transistor network of the succeeding gates C, D, E, the voltage at both nodes a, b varies between logic 0 and logic 1 around the threshold value (Figure 2). As a consequence, even byzantine faults may occur affecting both nodes, a and b, in a different way.

With the resistive bridge fault model, four test patterns are not any more sufficient. The resistance R_{sh} is an analog parameter, and for each value there may be a different pattern exhibiting the fault. An exact and complete fault coverage may not be obtainable any more, and a probabilistic analysis must be sufficient [EPRB03]. Diagnostic patterns must have

both, a high resolution and a high coverage of the intervals for R_{sh}.

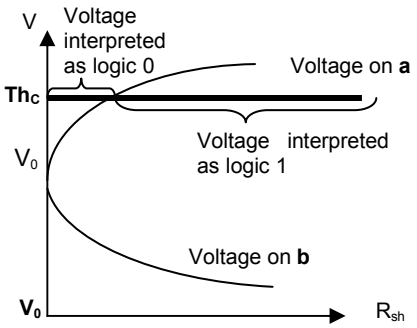

Figure 2: Analog detection interval

Open faults are concerned with interconnects [Venk00], and *stuck-open faults* model a non-conducting transistor.

Finally, the traditional stuck-at fault model is still useful. A large number of defects can also be detected by stuck-at faults, and diagnostic stuck-at patterns will help identifying the region where a fault is suspected.

2.3. Diagnostic pattern generation

The traditional approach for diagnosis is based on fault dictionaries for a test set T [PoRe97, Bopp94, Ches99]. It is mainly an efficiently organized spare matrix telling us the set of faults F_t for each faulty test outcome (a pattern and a response), and the set of outcomes T_f caused by a single fault f. For a fault f, the intersection

$$F_f := \bigcap_{t \in T_f} F_t$$

contains all the faults which cannot be distinguished from f by using T. In the best case, F_f will only include faults equivalent with f. The *resolution* of T with respect to f describes the size of F_f.

Working with fault dictionaries has its limits for large circuits. First, the fault dictionary size may explode. All fault models lead to a fault list size which is at least linear with the circuit size. The other dimension of the matrix is the test set which also increases with the size of the CUT. Second, in addition to the memory problem, constructing a fault dictionary requires long computing time as fault dropping is not any more allowed. All patterns have to be simulated for each fault at least as long as required for obtaining the specified resolution.

To avoid this high complexity more sophisticated dictionaries and algorithms [Liu04,Venk00a] are applied. Moreover, adaptive methods are used during characterization and statistical methods during manufacturing. Adaptive methods do not construct the fault dictionaries beforehand, but use simulation or test responses for identifying suspects [Gong95, Ghos99]. If the resolution of the test set is too low, ATPG and test application are intertwined. Statistical methods during manufacturing test rely on collecting as much test data as possible.

3. On-chip data generation for diagnosis

In this section, the diagnostic capabilities of design-for-test and BIST structures of digital cores and user defined logic are discussed. Memory analysis, diagnosis and repair are an already widely investigated and mature area, and analog, RF or mixed-signal diagnosis follow a different paradigm beyond the scope of this survey. Strategies for on-chip pattern generation reach from complete, autonomous built-in self-test to integrating additional hardware for decompressing, decoding or distributing external test-data.

Most often the so-called STUMPS architecture is used for implementing a pseudo-random self-test (Figure 3). This scheme is reusable for diagnosis of gate level faults which do not require pattern pairs [BaOr02, Wu99, RaTy97].

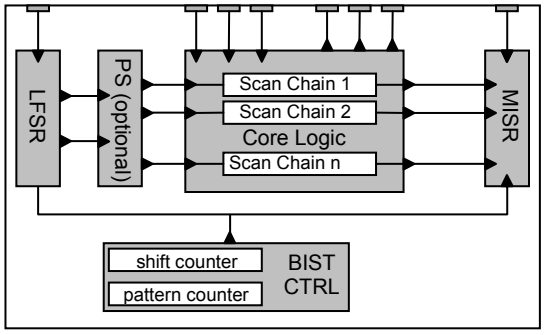

Figure 3: STUMPS [MWLE 83]

An LFSR generates pseudo-random vectors which fill multiple scan chains either directly or spread by a phase shifter. Even by applying rather a large vector number, the fault coverage obtained this way may not be sufficient, if the random-pattern testability of the CUT is too low. As a consequence, diagnoseability is low, too. The standard way to improve fault coverage is inserting test points into the CUT [VSW04, TaRa96, ChLi95]. These test points increase controllability, observability and hence diagnosability, but may affect both timing and area of the CUT.

Deterministic logic BIST schemes have the advantage not to touch the CUT, since they generate precomputed deterministic patterns. “Store-and-generate” schemes store the patterns on chip mostly in an encoded form, and offer some flexibility as patterns can be added or omitted. Reseeding of multi-polynomial LFSRs is a well known example of a “store-and-generate” scheme (Figure 4) [Hell95].

The seeds and encoded feedback values are stored in a memory, its width must be a few bits more than the number of specified bits in the encoded test vector and its length corresponds to the number of test vectors. Test sets with a high resolution are larger than standard test sets, and during characterization it may be necessary to reload the seed memory. There are also cases, where the number of specified bits of a test vector will increase in order to distinguish between two faults. As a result, the vector cannot be encoded any more by a seed and must be applied from external sources.

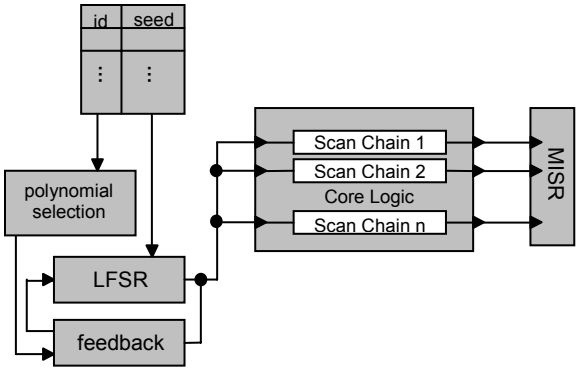

Figure 4: LFSR-reseeding [Hell95]

"Test-set-embedding" deterministic BIST schemes are based on a pseudo-random test pattern generator plus some additional circuitry that modifies the pseudo-random sequence in order to embed a set of deterministic patterns. Examples of such techniques are bit-fixing [ToMc96] and bit-flipping [WuKi96, Gher04] DLBIST schemes. The synthesis of these schemes requires first the mapping of the set of deterministic patterns to the initial pseudo-random sequence and the synthesis of the circuitry used to embed the target patterns. The basic scheme is shown in Figure 5.

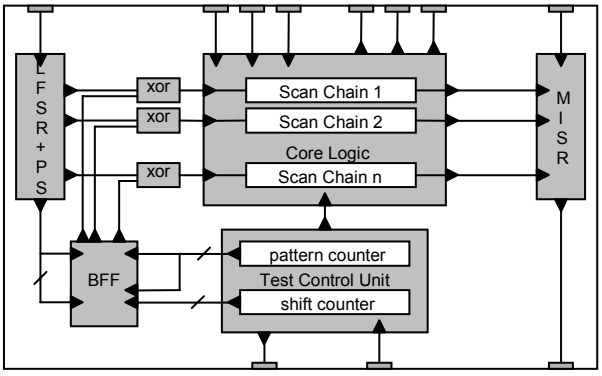

Figure 5: Bit-flipping DLBIST [WuKi96, Gher04]

The advantage of this scheme lies in the reduced hardware overhead compared with store-and generate schemes, whereas the drawback is its low flexibility. The generated deterministic test set is hardwired, but usually the high resolution test set for diagnosis and the compact test set after manufacturing differ and a reuse of the bit-flipping-function is not foreseen.

All of these BIST schemes also qualify for two pattern test for delay faults. Mainly three different techniques are known for this: *Enhanced scan* stores the required two patterns at the same time in the scan chain [Dasg81, Chen91]. It is most flexible and provides highest fault coverage. *Shifted scan* or *skewed load* test generates the activation pattern by shifting the initialization pattern a single bit [Savi92]. It needs less hardware overhead as the scan chains are not changed. But obviously not all possible pattern pairs can be generated this way, and the clocking scheme gets rather complex.

Finally, *functional justification* or *broadside test* uses the CUT response to the initialization pattern as the activation pattern [Savi94]. Here, clocking scheme and BIST control are simpler than in the shifted scan approach, but still not all possible inputs can be generated and there is a loss in fault coverage, too. The reduced fault coverage turns immediately into reduced diagnostic capabilities.

If the test control for the deterministic BIST hardware is not generated on chip but provided by a low cost tester, we would have more diagnostic options. Commercial schemes like EDT or OPMISR implement this type of test resource partitioning [Rajs02, Barn02]. Figure 6 illustrates this principle on the basis of EDT.

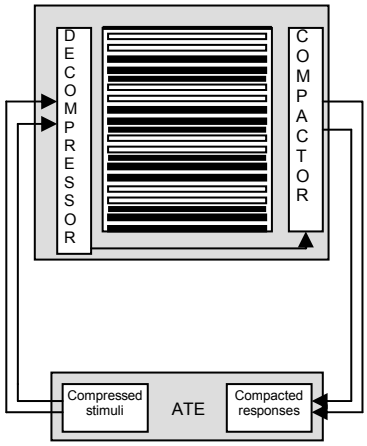

Figure 6: Principle of data compression

In recent years, a plethora of compression techniques has been published. They work well for diagnosis as long as the on-chip decompressor/decoder does not depend on the test set. Otherwise, decompression will loose its efficiency for diagnostic patterns [Arsl03].

4. On-chip response evaluation

A widely used technique for output data compression during BIST is signature analysis. Unfortunately, if an unknown value is shifted into the signature, after a few clock cycles the entire signature is corrupted. Hence, additional logic is required for masking unknowns at the CUT outputs [Tang04, Naru 03].

The X-masking logic may become rather large, if exactly the X's are masked, and is reduced, if some other outputs are used as don't cares. But the more outputs we mask, the lower both defect coverage and resolution will be [Tang04].

If compaction schemes are used for test resource partitioning, we have more choices. Space compaction maps the large output vector of multiple scan chains to smaller words and reduces bandwidth requirements Mitra proposed a compaction scheme based on XOR-trees which tolerate a certain number of X's without loss of fault coverage [MiKi04].

Convolutional compactors combine space and time compaction. Reponses are continuously collected and compressed, but after a certain while the compactor will not depend on previous states any more and will also forget the previous unknowns [Rajs03, Wang03]. During BIST, the signature register cannot be observed in general, and it takes

more effort to determine all the patterns the circuit fails with. The classical approach is based on bisection, and a large variation has been published [Liu03, Pate02, Wohl02, Clou01].

The small procedure below describes how to find all failing patterns within [1,N], if after N patterns the signature is corrupted:

Perform BIST for all patterns within $\left[1, \frac{N}{2}\right]$

If the signature after $\frac{N}{2}$ patterns is correct:

Find all the failing patterns within $\left[\frac{N}{2}+1, N\right]$.

Else

Find all the failing patterns within $\left[1, \frac{N}{2}\right]$.

Load the correct seeds for pattern $\frac{N}{2}+1$ into the random pattern generator and the MISR.

Find all the failing patterns within $\left[\frac{N}{2}+1, N\right]$.

This process can speed up, if also results of fault simulation are taken into account. They allow a weighted process where the interval is not divided at $\frac{N}{2}$ but at a number M where half of the faults are detected for the first time by patterns in [1, M] and the other half in [M+1, N]. Such an approach requires a fault dictionary which has to be constructed for diagnosis anyway.

An alternative to the iterative methods is analyzing the signatures for finding the failing flip-flops [Karp94, Pate02a, RaTy99, BaOr01,02]. This is not an option for the complete BIST run as we will face multiple faults or burst faults which destroy the information. The resolution of signature analysis can be improved if multiple runs with different feedback polynomial are applied and various scan chains are masked out during these runs [Lein04, Goes04].

5. Summary

Debug and diagnosis are receiving increasing attention and are of growing economic relevance for nanometer scale technologies. Techniques for BIST and embedded test support diagnosis significantly, and most benefits are provided if the embedded test solution is flexible and programmable. Changing and increasing test sets has to be possible without restrictions.

On the output side, embedded test and BIST solutions have to provide the option of complete observability, and they have to support collecting data for diagnosis already during manufacturing test.

6. References

[Arsl03] B. Arslan, A. Orailoglu, "Perfect Conservation of Diagnostic Information in Aggressive Reduction of SOC Test Bandwidth and Use," *IEEE European Test Workshop (ETW)*, p. 13, May 2003

[Bopp94] V. Boppana, W. K. Fuchs, "Fault Dictionary Compaction by the Elimination of Output Sequences," *SIGDA Super Compendium ICCAD,* November 1994

[BaOr01] I. Bayraktaroglu and A. Orailoglu, "Diagnosis for Scan-Based BIST: Reaching Deep into the Signatures," *Proc. Design, Automation and Test in Europe (DATE)*, pp. 102 - 109, March 2001

[BaOr02] I. Bayraktaroglu and A. Orailoglu, "Gate Level Fault Diagnosis in Scan-Based BIST," *Proc. Design Automation and Test in Europe (DATE),* pp. 376-381, March 2002

[BaOr02a] I. Bayraktaroglu and A. Orailoglu, "Cost-effective deterministic partitioning for rapid diagnosis in scan-based BIST," *IEEE Design & Test*, vol. 19, no. 1, pp. 42-53, January 2002

[Barn02] C Barnhart et al., "Extending OPMISR beyond 10x scan test efficiency," *IEEE Design and Test*, vol. 19, No. 5, pp. 65-73, 2002

[Bork03] S. Borkar et al., "Parameter Variations and Impact on Circuits and Microrarchitecture, *Design Automation Conference (DAC)*, pp. 338-342, June 2003

[Chen91] K.-T. Cheng et al., "A partial enhanced scan approach to robust delay-fault test generation for sequential circuits", *Proceedings of the IEEE International Test Conference (ITC)*, pp. 403-410, Oct. 1991

[Chen99] K.T. Cheng et al., "Fault Emulation: A New Methodology for Fault Grading," *Trans. On Computer-Aided Design of Integrated Circuits and Systems*, vol.18, no. 10, October 1999

[Ches95] B. Chess et al., "Diagnosis of Realistic Bridging Faults with Single Stuck-at Information," *Proc. Int'l Conf. Computer-Aided Design*, pp. 185-192, November 1995

[Ches99]B. Chess, T. Larrabee, "Creating small fault dictionaries," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 3, pp. 346 – 356, March 1999

[ChLi95] K.-T. Chen, C.-J. Lin, "Timing-Driven Test Point Insertion for Full-Scan and Partial-Scan BIST," *Proceedings of the IEEE International Test Conference (ITC)*, pp. 506-514, Oct. 1995

[Clou01] T. Clouqueur et al., "Efficient Signature-Based Fault Diagnosis UsingVariable Size Windows," Proc. VLSI Design Conference, pp. 387-392, January 2001

[Dasg81] S. Dasgupta et al., "An enhancement to LSSD and some applications of LSSD in reliability, availability, and serviceability," *Fault-Tolerant Computing Symposium (FTCS)*, pp. 32-34, June 1981

[EPRB03] P. Engelke et al, "Simulating resistive bridging and stuck-at faults," *Proceedings of the IEEE International Test Conference (ITC)*, pp. 1051-1059, Sept. 30-Oct. 2, 2003

[Gher04] V. Gherman et al., "Efficient Pattern Mapping For Deterministic Logic BIST," IEEE International Test Conference (ITC 2004), Charlotte, NC, USA, pp. 48 - 56, October 25-28, 2004

[Ghos99] J. Ghosh-Dastidar, N. A. Touba, "Adaptive Techniques for Improving Delay Fault Diagnosis,"*Proc. VLSI Test Symposium*, pp. 168 – 172, 25-29 April 1999

[Goes04] M. Goessel et al., "A Signature Analysis Technique for the Identification of Failing Vectors with Application to Scan-BIST," *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 20, pp. 611-622, Dec. 2004

[Gong95] Y. Gong, S. Chakravarty, "On Adaptive Diagnostic Test Generation," *Conference on Computer-Aided Design (ICCAD),* p. 0181, November 1995

[Hell95] S. Hellebrand et al., "Built-In Test for Circuits with Scan Based on Reseeding of Multiple-Polynomial Linear Feedback Shift Registers," *IEEE Transactions on Computers*, vol. 44, no. 2, pp. 223-233, February 1995

[Hora02a] C. Hora et al. "An Effective Diagnosis Method to Support Yield Improvement," *Proc. VLSI Test Symp*., pp. 260 – 269, 7-10 Oct. 2002

[Hora02b] C. Hora et al. "On a statistical fault diagnosis approach enabling fast yield ramp-up," *IEEE European Test Workshop (ETW*), pp. 193, May 2002

[Huan04] Y. Huang et al., "Intermittent Scan Chain Fault Diagnosis Based on Signal Probability Analysis," *Proc. Design, Automation and Test in Europe (DATE)*, p. 21072, February 2004

[Karp94] M. G. Karpovsky et al., "Diagnosis by signature analysis of test responses," *IEEE Trans. on Computers*, vol. 43., no. 2. pp. 141 – 152, February 1994

[Krst03a] A. Krstic et al., "Delay Defect Diagnosis Based Upon Statistical Timing Models - The First Step," *Proc. Design, Automation and Test in Europe (DATE)*, p. 10328, March 2003

[Krst03b] A. Krstic et al., "Diagnosis-Based Post-Silicon Timing Validation Using Statistical Tools and Methodologies," *Proc. IEEE International Test Conference,* pp. 339-348, Sept. 30-Oct. 2, 2003

[Krst03c] A. Krstic et al., "Enhancing Diagnosis Resolution For Delay Defects Based Upon Statistical Timing and Statistical Fault Models," *Proc. IEEE/ACM Design Automation Conference,* pp. 668-673, June 2003

[Lein04] A. Leininger et al, "Diagnosis of Scan-Chains by Use of a Configurable Signature Register and Error-Correcting Codes," *Proc. Design, Automation and Test in Europe (DATE)*, p. 21302, February 2004

[Liu03] C. Liu, K. Chakrabarty, "Failing Vector Identification Based on Overlapping Intervals of Test Vectors in a Scan-BIST Environment," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 4, pp. 593-604, May 2003

[Liu04] C. Liu, K. Chakrabarty, "Compact dictionaries for fault diagnosis in scan-BIST," *IEEE Trans. on Computers*, vol. 53, no. 6; pp. 775-780, June 2004

[Lude04] R. Ludewig et al., "Architecture for Testing and Debugging of System-On-Chip Components," *IEEE Workshop on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, pp. 91-98, April 2004

[Majh03] A. K. Majhi et al., "Improving Diagnostic Resolution of Delay Faults using Path Delay Fault Model," *Proc. VLSI Test Symp*., pp. 345, Apr. 27-May 1, 2003

[Metr00] C. Metra et al, "Self-checking detection and diagnosis of transient, delay, and crosstalk faults affecting bus lines," *IEEE Trans. on Computers*, vol. 49, no. 6; pp. 560-574, June 2000

[MiKi04] S. Mitra and K. S. Kim, "X-Compact: an efficient response compaction technique," *IEEE Transactions on Circuit-Aided Design (CAD)*, vol. 6, pp. 290-294, March 2004.

[MWLE83] F. Motika et al., „An LSSD Pseudo-Random Pattern Test System," *Proceedings of the IEEE International Test Conference (ITC)*, pp. 283-288, November 1983

[Naru03] M. Naruse et al., "On-chip compression of output responses with unknown values using LFSR reseeding," *Proceedings of the IEEE International Test Conference (ITC)*, pp.1060-1068, Sept. 30-Oct. 2, 2003

[Padm03] S. Padmanaban and S. Tragoudas, "Non-Enumerative Path Delay Fault Diagnosis," *Proc. Design, Automation and Test in Europe (DATE)*, p. 10322, March 2003

[Pate02] S. Pateras, "Embedded Diagnosis IP," *Proc. Design, Automation and Test in Europe (DATE)*, pp. 242-244, March 2002

[Pate02a] J. H. Patel et al., "Application of Saluja-Karpovsky compactors to test responses with many unknowns," *Proc. VLSI Test Symposium*, pp. 107- 112, 27 April-1 May 2003

[PoRe97] I. Pomeranz, S. M. Reddy, "On dictionary-based fault location in digital logic circuits," *IEEE Trans. on Computers*, vol. 46, no. 1; pp. 48-59, January 1997

[Rajs02] J. Rajski et al., "Embedded deterministic test for low cost manufacturing test," *Proceedings of the IEEE International Test Conference (ITC)*, pp. 301-310, 2002

[Rajs03] J. Rajski et al., "Convolutional compaction of test responses, " *Proceedings of the IEEE International Test Conference (ITC)*, pp. 745-754, Sept. 30-Oct. 2, 2003.

[RaTy97] J. Rajski and J. Tyszer, "Fault Diagnosis In- Scan-Based BIST," *Proceedings IEEE International Test Conference*, pp. 894-902, 1997

[RaTy99] J. Rajski, J. Tyszer, "Diagnosis of scan cells in BIST environment," *IEEE Trans. on Computers*, vol 48, no. 7; pp. 724-731, July 1999

[RHB95] M. Renovell et al., "The concept of resistance interval: A new parametric model for resistive bridging fault," *Proceedings of the IEEE VLSI Test Symposium (VTS)*, pp. 184-189, April 30 - May 3, 1995

[Savi92] J. Savir, "Skewed-Load Transition Test: Part I, Calculus," *Proceedings of the IEEE International Test Conference (ITC)*, pp. 705-713, Sep.1992

[Savi94] J. Savir, S. Patir, "Broad-Side Delay Test," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 13, no. 8, pp. 1057-1064, 1994

[Tang04] Y. Tang et al., "X-Masking During Logic BIST and Its Impact on Defect Coverage," *IEEE International Test Conference (ITC)*, pp. 442 - 451, October 25-28, 2004

[TaRa96] N. Tamarapalli, J. Rajski, "Constructive Multi-Phase Test Point Insertion for Scan-Based BIST," *Proceedings of the IEEE International Test Conference (ITC)*, pp. 649-658, Oct. 1996

[ToMc96] N. A. Touba, E.J. McCluskey, "Altering a pseudo-random bit sequence for scan-based BIST," *Proceedings IEEE International Test Conference*, pp.167-175, October 1996

[Venk97] S. Venkataraman, W.K. Fuchs, "A Deductive Technique for Diagnosis of Bridging Faults," *Conference on Computer-Aided Design (ICCAD),* pp. 562-567, November 1997

[Venk00] S. Venkataraman, S. B. Drummonds, "A Technique for Logic Fault Diagnosis of Interconnect Open Defects," *Proc. IEEE VLSI Test Symposium (VTS),* p. 313, April 30 - May 4, 2000

[Venk00a] S. Venkataraman, S. B. Drummonds, "POIROT 1: A Logic Fault Diagnosis Tool and Its Applications," *Proc. International Test Conference (ITC)*, pp. 253 – 262, 3-5 Oct. 2000

[VSW04] H.Vranken et al., "Impact of Test Point Insertion on Silicon Area and Timing during Layout," *Proc. Design, Automation and Test in Europe (DATE)*, pp.810 - 815, February 16-20

[Wang03] C. Wang et al., "On compacting test responses data containing unknown values," *Proc. International Conference on Computer-Aided Design*, pp. 855-862, November 2003

[Wohl02] P. Wohl et al., "Effective diagnostics through interval unloads in a BIST environment," *Proc. IEEE/ACM Design Automation Conference*, pp. 249-254, June 2002

[Wu99] Y. Wu and S. M. I. Adham, "Scan-Based BIST Fault Diagnosis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 2, , pp. 203-211, Feb. 1999

[WuKi96] H.-J. Wunderlich, G. Kiefer, "Bit-Flipping BIST," *Proceedings ACM/IEEE International Conference on CAD (ICCAD)*, pp. 337-343, November 1996

Acknowledgement: Part of the work has been funded by the DFG under DFG FOR 460/1-1.6.