Hierarchical Analysis of Short Defects between Metal Lines in CMOS IC

Witold A. Pleskacz[1], Maksim Jenihhin[2], Jaan Raik[2], Michal Rakowski[1], Raimund Ubar[2], Wieslaw Kuzmicz[1]

[1]Warsaw University of Technology, Poland
{W.Pleskacz, W.Kuzmicz}@imio.pw.edu.pl, M.Rakowski@elka.pw.edu.pl
[2]Tallinn University of Technology, Estonia
{maksim,jaan,raiub}@pld.ttu.ee

Abstract

Current paper proposes a new hierarchical approach to defect-oriented testing of CMOS circuits. The method is based on critical area extraction for identifying the possible shorted pairs of nets on the basis of the chip layout information, combined with logic-level test pattern generation. The novel contributions of the paper are a new bridging fault simulator and a test pattern generator, which are able to handle defects creating feedbacks into the circuit. As a preprocessing step, a combined stuck-at test set from two different test pattern generators implementing alternative strategies (pseudorandom and deterministic) were created. Nevertheless, many short defects were not covered by this extended stuck-at approach. Analyses carried out in this paper show that the stuck-at tests are not covering up to 4 % of the shorts (both testable and untestable). The test coverage (fault efficiency) can be increased by the new generator by up to 0.4 % in comparison to full stuck-at test. Layout analysis for a set of benchmarks has been performed. The experiments indicate how the number of bridging faults of non-zero probability is dependent on the circuit size.

1. Introduction

Transition towards nanometer technologies results in an increasing complexity of digital circuits and in a growth of circuit parameter variations. There are two, somewhat contradicting ways to cope with this challenge: applying hierarchical models to handle the test complexity, and relying on defect-oriented approaches to model the new failure mechanisms more accurately. Solutions combining both of these two strategies would therefore be desirable.

Conventional fault modeling methods rely on gate-level representations of digital circuits and the stuck-at fault (SAF) model. However, the SAF model, which has been extremely popular in test quality assessment over the past decades, has not withstood the test of time. It has been shown that high SAF coverage does not guarantee sufficient quality of testing, at least in the case of CMOS integrated circuits [1-3]. Physical defects that may occur in real circuits do not manifest themselves as stuck-at faults.

Previous studies have shown that shorts are the most common type of defects, followed by opens [4, 5]. Logic-level fault models for shorts have been known for decades. However, the traditional bridging fault model does not scale due to the fact that the number of faults to be considered would grow as a square of the total number of nets in the circuit. Thus, an approach is needed that would provide for an easy identification of net pairs that are most likely to get shorted.

In [7] the authors propose a method for hierarchical analysis of short defects. However, the method does not support fault simulation for feedback defects. Furthermore, the logic-level analysis resorted to deterministic ATPG, thus resulting in serious performance problems. Current paper includes a set of benchmark circuits of different complexity, which allows making conclusions about the scaling of the method.

The main contribution of current paper is a fast hierarchical method for targeting the interconnect shorts. First, at the layout level, a method for detecting the possible shorted pairs of nets is implemented. The concept of critical area is used to compute the probabilities of possible shorts. Then, at the logic level, the shorts with non-zero probabilities are tested using the bridging fault model.

Critical area extraction tools have been presented also in previous works (e.g. [8, 17]). In this paper we use a more accurate approach, which is based on industry standards and commercial CAD software.

Nigh et al. relied on a hierarchical approach [9], where defect analysis was applied for logic-level IDDQ testing. Here, we are considering a different problem, where shorts in CMOS circuits are targeted. Therefore, a more complex problem is studied as IDDQ test does not require fault effect propagation.

The novel contributions of the paper are a new bridging fault simulator and a test pattern generator, which are able to handle defects creating feedbacks into the circuit. As a preprocessing step, a combined stuck-at test set from two different test pattern generators implementing alternative strategies (pseudorandom and deterministic) were created. Nevertheless, many short defects were not covered by this extended stuck-at approach. Analyses carried out in this paper show that the stuck-at tests are not covering up to 4 % of the shorts (both testable and untestable). The test coverage (fault efficiency) can be increased by the new generator by up to 0.4 % in comparison to full stuck-at test. The experiments indicate that the number of bridging faults of non-zero probability is increasing linearly with the circuit size.

The core advantages of the method presented in this paper are the following. It allows to minimize the number of considered shorted pairs and to assign a probabilistic measure to each of them. In addition, the research provides for an insight to defect distribution in real CMOS layouts. The methods proposed in this paper are supported by a design flow implementing commercial [10] and academic [11] CAD software.

The paper is organized as follows. In Section 2 we explain the probabilistic fault modeling and the concept of critical area. In Section 3 we describe the logic-level analysis of short defects. Section 4 presents experimental results on real layout data, and finally, conclusions are drawn.

2. Probabilistic defect modeling

A short is a piece of extra conducting material that connects a pair of separate conducting regions in the integrated circuit. This affects the connectivity of the circuit: two separate electrical nets become connected. It is intuitively obvious that probabilities of shorts depend on the layout of the circuit. Conducting regions that are adjacent to one another are more susceptible to shorts than regions that are separated by a large distance. We assume that every defect that results in a short can be approximated by a circle. In order to estimate the probabilities of shorts between pairs of nodes we use the concept of critical area for shorts [12]. The critical region for shorts is such a region in the circuit that, if the center of a defect of a given radius *R* is located anywhere inside the critical area, a short between two adjacent conducting regions occurs (see Fig. 1).

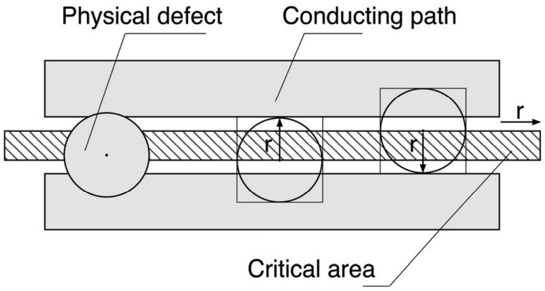

Fig. 1. The concept of critical area

The size of the critical area depends on the shapes and locations of the conducting regions that will be shorted and is a function of the defect radius *R*. The radii of defects are randomly distributed and can be characterized by a probability density function $P_{df}(R)$. This function was modeled in the following way [12]:

$$P_{df} = D_{oi} \cdot \begin{cases} \dfrac{2(p_i-1)\cdot R}{(p_i+1)X_{oi}^2} \forall 0 < R < X_{oi} \\ \dfrac{2(p_i-1)X_{oi}^{p_i-1}}{(p_i+1)\cdot R^{p_i}} \forall X_{oi} < R \end{cases}$$

Where *i* is the type of conducting layer, D_{oi} is the density of physical defects and p_i and X_{oi} are modeling parameters. p_i is set to 3, X_{oi} to 20% of the minimal distance between shapes of a given conducting layer and D_{oi} to 10 defects/cm^2.

We assume that the probability of a short between a pair of conducting regions that correspond to a pair of electrical nodes is proportional to the critical area for these two regions. The critical area for shorts is a function $P_S(R)$ of the defect radius *R*. Since the defect radii exhibit a random distribution, the product $P_{df}(R) \cdot P_S(R)$ integrated over the range of *R* where $P_S(R) > 0$ can be taken as the measure of the total probability of shorts between a given pair of nodes. Applying a Poisson based yield model the probability of a short is given by the following formula:

$$P_t = 1 - \prod_{i=1}^{N} \exp\left[-\int_0^{\infty} \mathrm{P_s(R)} \cdot \mathrm{P_{df}}(R)dR \right]$$

The first step in identification of logic faults and their probabilities is to calculate P_t for all pairs of conducting regions representing electrical nodes. If for a given pair P_t is zero, this pair of nodes is not taken into account. For the pairs that can be shorted tests are generated at the logic-level as explained in the Section 3.

Inductive fault analysis [3, 18] allows to predict faults that may occur in CMOS IC. It can be used to reduce the fault set size, but defect occurrence probabilities are not available at the time of test patterns generation. In [19] Monte Carlo-based inductive contamination analysis was demonstrated for complete fault characterization of standard cell libraries. Such a characterization can be applied for very accurate assessment of defect coverage and test generation. Nevertheless this method is too complex and time consuming to be used for test generation purposes. In [7] we described in detail the first step of the hierarchical test generation approach for CMOS shorts – the Critical Area Extraction (CAE). Compared to previously published CAE tools (e.g. FedEx [8]), the solution implemented in this paper has a number of benefits. Different from [8], our approach allows extracting full critical areas, to use complex defect size distribution (actual industry standard) and to assign probabilistic metrics to defects. Additionally, FedEx supports only flattened layouts, while with current tool also hierarchical layouts can be analyzed. The tool presented in this paper has been implemented in the form of approx. 4500 lines of Cadence SKILL scripts in the Warsaw University of Technology.

3. Logic-level analysis of short defects

In this Section, we will briefly explain the bridging fault model and put a stress on the different types of tests that might be required in order to cover such faults. It is important to note that the hierarchical method presented in this paper is not limited to the bridging fault model. Other models for short defects, including the dominating line fault model, can be easily obtained by using the test results of the proposed test pattern generator.

3.1 The fault model

A *bridging fault* (BF) is a fault modeled as a short between two or more usually unconnected circuit lines. In order to avoid a combinational explosion of the fault list, only faults between two lines will be considered. The model assumes also that only one pair of lines is shorted at a time. In cases when the number of shorted lines is larger than 2 the shorts can be examined as separate pairs. In practice a short can exist only between closely located lines and thus its probability is highly dependent on the particular circuit layout. In current work, actual CMOS layouts for six of ISCAS'85 logic benchmark circuits were created. The list of shorts investigated for the circuit under test is based on a layout analysis explained in Section 2. A detailed description of the defect probability calculation approach can be found in [7].

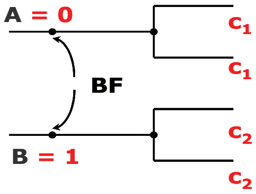

Fig. 3. A bridging fault between lines A and B

In order to observe the behavior caused by a bridging fault (Fig. 3) the lines *A* and *B* should be driven by opposite logic values. The resulting logic values on the shorted lines c_1 and c_2 become equal to some determined value *c*: the one that is 'stronger' and overrides the other in the current technology. In case if *c* holds a logic zero the fault behaves like if an AND-gate was inserted between the lines. Such a fault is known as *wired-AND bridging fault* (Fig. 4).

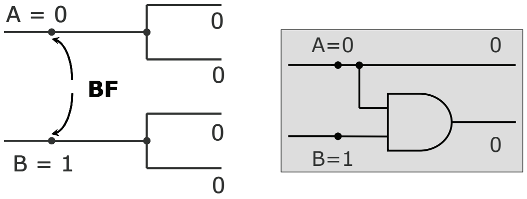

Figure 4. Wired-AND bridging fault

Correspondingly, in the case where *c* is *logic one* the fault behaves like if an OR-gate was inserted between the lines. Such a fault is known as *wired-OR bridging fault* (Fig. 5).

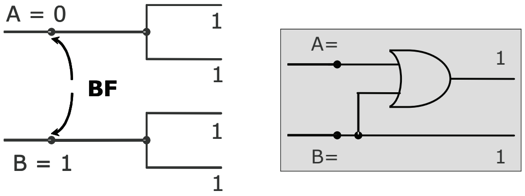

Fig. 5. Wired-OR bridging fault

If one of the shorted nodes depends on the other then a feedback loop occurs. *Feedback bridging fault* (FBF) converts a combinational circuit into sequential one. As we will see below, depending on the logic values in the test pattern, this fault can behave in different ways. While for combinational tests only one test vector is needed, for sequential tests the constraints which have to be met by the ATPG for activating a defect might be spread over different clock cycles and represent a sequence of patterns. A long path of the dependency between the shorted nodes and its complexity can lead even to difficult to model *oscillation* effects.

In many cases a FBF can be detected by a combinational test in only one clock cycle. This is the case, when the faulty voltage signal is fed forward through the short-path, i.e. the node which is assigned '1' is closer to a primary output than the shorted node being assigned to '0'. We will refer to these nodes as "front-node" and "back-node". When detecting a defect, preference should be given to such a test.

3.2 The defect-oriented test generator

In this paper we implemented a new bridging fault simulator and an automated test pattern generator (ATPG) that support single-pattern feedback tests. The ATPG tool is based on the PODEM algorithm [16] and utilizes circuit modifications for representing faulty circuits as shown below.

Let us consider the case, where our goal is to test the wired-AND short where line B drives line A to the value "0". In that case we introduce circuit modification shown in Figure 6. We insert an OR-gate connecting lines A and B, and generate stuck-at-0 test for the line A. This forces the test generator to assign value "1" to the faulty line A in order to sensitize the fault effect. To allow propagation through the gate, the SAF ATPG will have to set line B to "0". This modification is exactly equivalent to wired-AND fault model.

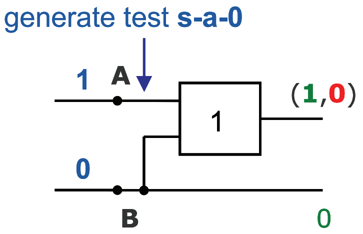

Fig. 6. Circuit modification for testing wired-AND

However, single-pattern testable shorts can not be tested by similar modifications. Therefore, in this paper we have introduced a combination of a simple modification and an improvement to a combinational ATPG. Consider the circuit shown in black color in Fig. 7. Let us assume that there is an AND-short between A and B, where front-node B forces back-node A to value "0". Similar to Fig. 6, an OR-gate is included and s-a-0 fault at its input is targeted. However, in single-pattern feedback shorts we must guarantee that fault effect will not propagate through B, thus creating an oscillation. For that purpose an extra fanout output is added and checked by the ATPG to make sure that the faulty value of B remains "0".

In the proposed hierarchical analysis method for CMOS defects we used the following strategy:

1. Generate 100 % test coverage stuck-at tests by two different tools (pseudorandom, deterministic).
2. Combine the two test sets obtained.
3. Fault simulate them using the new fault simulator for short defects. Discard the detected shorts.
4. Run the new defect-oriented ATPG for the remaining faults. If a fault may act as a feedback then first try to generate a combinational test for it.

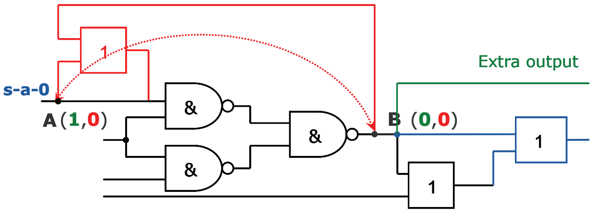

Fig. 7. Modification for testing feedback shorts

4. Experimental results

Experimental results were carried out on six ISCAS'85 circuits. In order to run the layout analysis the Verilog versions of the circuits had to be resynthesized by performing mapping into standard cell library HRDLIB of AMS CMOS 0.8 µm CYE technology. Synopsys Design Compiler was used in the synthesis process. The set of circuits was limited to the six examples shown in Table 1 because of the fact that for the other benchmarks the Verilog versions were unavailable. The layout synthesis was carried out with a commercial tool Cadence Silicon Ensemble (version 5.4).

Table 1. Characteristics of the stuck-at tests

		Benchmarks					
		c432	c880	c1355	c1908	c3540	c5315
SAF faults		678	1512	2014	1472	3284	6792
Deterministic SAF	Tested	678	1512	2014	1472	3165	6777
	Untestable	0	0	0	0	119	15
Pseudorand. SAF	Covered	678	1512	2014	1472	3165	6777

As it can be seen from Table 1 deterministic and pseudorandom generators from the Turbo Tester suite [11] generated 100 % fault efficiency stuck-at fault (SAF) tests. Note, that the number of SAFs in the resynthesized circuits does not match the original circuits. Also, the number of untestable faults after synthesis is lower than initially in ISCAS'85 designs. Nevertheless, the logic functionality of the benchmarks is not altered. Apart from practical considerations, such as relying on commercial CAD utilities, the resynthesis offers some benefits also to the experimental analysis. First, circuits obtained by logic synthesis represent a more realistic case of combinational logic designs. Second, by minimizing the number of untestable SAF faults the set of combinationally redundant short defects is also kept small. Still a large amount of untestable shorts is reported as a result of the analysis (see Tables 2 and 3).

Tables 2 and 3 present experimental results for wired-AND and wired-OR models, respectively. The first row in the tables shows the number of short defects with non-zero probability. As it was mentioned above, in practice a short can exist only between closely located lines and thus its probability is highly dependent on the particular circuit layout. In the theoretical worst case the number of short defects to be analyzed is equal to $n(n-1)/2$, where n is the number of conducting lines in the circuit under test. However, our analysis shows that in a real layout the number of shorts of non-zero probability is much lower and grows linearly with respect to the circuit size. Figure 8 presents the dependency of the number of such shorts to the number of SAF faults.

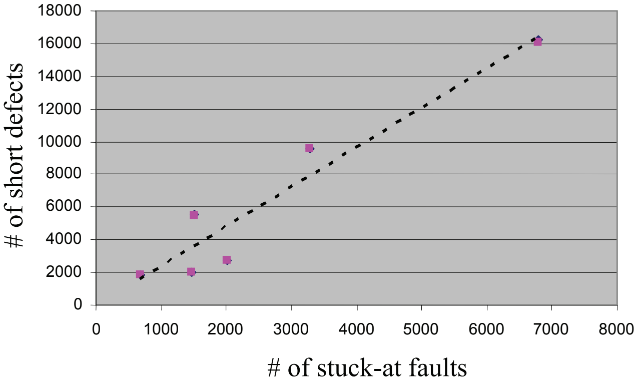

Fig. 8. Number of considered shorts vs. SAFs

The 2nd and 3rd rows show the number of short defects not covered by the full SAF tests of the deterministic and the pseudorandom test generators. The 4th row shows the same number for the combined tests of the two generators. As it can be seen, the SAF tests are almost never able to fully cover the short defects (except for c1355). In the case of c432, almost 4 % (!) of the shorts are not covered by SAF.

Rows 5 and 6 present the number of short defects detected by combinational and by feedback tests, respectively. Row 7 shows the total number of defects not covered by any SAF test but tested by the defect-oriented test pattern generator proposed in this paper (denoted with bold font). Finally rows 8, 9 and 10 show the number of short defects proven untestable by the new test generation tool (also printed in bold). Row 11 presents the defects that were aborted.

The most unexpected result of the analysis carried out in this paper is the fact that a considerable amount of short defects were untestable by single-pattern tests. (See row 9 marked with grey background). All the previous works have suggested that the number of bridging faults requiring oscillation tests should be negligible (e.g. [6, 7]). However, experimental proof has been missing until now. Current paper proves that this assumption is not true. We show that single-pattern-untestable defects make in average 0.45 % per cent of the total shorts and in the case of c432 they can reach as high as 2 per cent.

6. Conclusions

Current paper proposes a new hierarchical approach to defect-oriented testing of CMOS circuits. The method is based on critical area extraction for identifying the possible shorted pairs of nets on the basis of the chip layout information, combined with logic-level test pattern generation.

The novel contributions of the paper are a new bridging fault simulator and a test pattern generator, which are able to handle defects creating feedbacks into the circuit. As a preprocessing step, a combined stuck-at test set from two different test pattern generators implementing alternative strategies (pseudorandom and deterministic) were created. Nevertheless, many short defects were not covered by this extended stuck-at approach. Analyses carried out in this paper show that the stuck-at tests are not covering up to 4 % of the shorts (both testable and untestable). The test coverage (fault efficiency) can be increased by the new generator by up to 0.4 % in comparison to full stuck-at test. Layout analysis for a set of six benchmarks has been performed. The experiments indicate that the number of bridging faults of non-zero probability is increasing linearly with the circuit size.

The core advantages of the method presented in this paper are the following. It allows to minimize the number of considered shorted pairs and to assign a probabilistic measure to each of them. In addition, the research provides for an insight to defect distribution in real CMOS layouts. The methods proposed in this paper are supported by a design flow implementing commercial and academic CAD software.

References

[1] J.M. Soden, C.F. Hawkins. Quality Testing Requires Quality Thinking. *Int. Test Conference*, 1993,pp. 596.

[2] W.Maly, et al. System Characterization of Physical Defects for Fault Analysis of MOS IC Cells. *Proc. 1984 ITC*, pp. 390-399, Philadelphia, 1984.

[3] J.P.Shen, W.Maly, J.Ferguson. Inductive Fault Analysis of MOS ICs. *IEEE Design&Test*, pp.13-26, 1985.

[4] M. Sachdev "Defect Oriented Testing for CMOS Analog and Digital Circuits" Kluwer Academic Publisher,1998

[5] Prediction Software Ltd., March 2005, URL: http://www.icyield.com/yield_enhan_proc.html

[6] Abramovici, Breuer, Friedmann "Digital Systems Testing and Testable Design" IEEE press 1990

[7] M. Jenihhin, J. Raik, R. Ubar, W. Pleskacz, M. Rakowski. Layout to Logic Defect Analysis for Hierarchical Test Generation. *DDECS'07*, pp. 35-40, 2007.

[8] Z. Stanojevic, D. Walker. FedEx – A Fast Bridging Fault Extractor. *ITC'01*. pp. 696-703, 2001.

[9] P. Nigh, D. Forlenza, F. Motika. Application and Analysis of IDDQ Diagnostic Software, *ITC'97*, pp.319-327, 1997.

[10] Cadence tools, URL:http://www.cadence.com

[11] Turbo Tester tools, URL:http://www.pld.ttu.ee/tt

[12] P.Nigh and W.Maly. Layout - Driven Test Generation. *Proc.1989 ICCAD*, pp. 154-157, 1989.

[13] S. Xu, S. Su "Detecting I/O and internal feedback faults", *IEEE trans. on computers*, vol. c-34, no. 6, June 1985

[14] W. A. Pleskacz, D. Kasprowicz, T. Oleszczak, W.Kuzmicz. CMOS Standard Cells Characterization for Defect Based Testing. *Proc. of IEEE Intl. Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 384-392, San Francisco, USA, October 2001.

[15] D. Kasprowicz, W. A. Pleskacz: "Improvement of Integrated Circuit Testing Reliability by Using the Defect Based Approach, *Microelectronics Reliability*, Elsevier Science, vol. 43/6, pp. 945-953, June 2003.

[16] P.Goel. An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits. *IEEE Trans. on Computers*, vol. C-30, no.3 pp. 215-222, March 1981.

[17] G.A. Allan. Targeted layout modifications for semiconductor yield/reliability enhancement. *IEEE Trans. on Semiconductor Manufacturing,* Volume 17, Issue 4, Nov. 2004 Page(s):573 – 581.

[18] F.J. Ferguson, J.P. Shen, "Extraction and simulation of realistic CMOS faults using inductive fault analysis," *Proc. Int. Test Conf.*, pp. 475-484, USA, 1988.

[19] J. Khare, W. Maly, N. Tiday, "Fault characterization of standard cell libraries using inductive contamination analysis (ICA)," *Proc. VLSI Test Symp.*, pp. 405-413, USA, April 1996.

Table 2. Experimental analysis of wired-AND shorts

		Benchmarks					
		c432	c880	c1355	c1908	c3540	c5315
Pairs of nets with a short probability		1807	5476	2757	1988	9446	16092
Not covered by simulation of SAF sequences	Deterministic	70	64	0	6	167	116
	Pseudorandom	61	63	0	9	162	111
	in Total	57	61	0	5	154	111
Tested by Defect TPG	Non-feedback test	1	0	0	0	3	1
	Feedback test	0	0	0	0	0	1
	in Total	**1**	**0**	**0**	**0**	**3**	**2**
Proven Untestable by Defect TPG	Non-feedback test	20	54	0	4	106	96
	Feedback test	36	7	0	1	45	11
	in Total	**56**	**61**	**0**	**5**	**151**	**107**
Aborted by Defect TPG		0	0	0	0	0	2

Table 3. Experimental analysis of wired-OR shorts

		Benchmarks					
		c432	c880	c1355	c1908	c3540	c5315
Pairs of nets with a short probability		1807	5469	2759	2000	9440	15989
Not covered by simulation of SAF sequences	Deterministic	94	25	0	7	153	92
	Pseudorandom	78	27	0	5	154	82
	in Total	70	22	0	5	153	79
Tested by Defect TPG	Non-feedback test	7	0	0	0	4	4
	Feedback test	1	0	0	0	0	0
	in Total	**8**	**0**	**0**	**0**	**4**	**4**
Proven Untestable by Defect TPG	Non-feedback test	29	20	0	4	86	58
	Feedback test	33	2	0	1	63	14
	in Total	**62**	**22**	**0**	**5**	**149**	**72**
Aborted by Defect TPG		0	0	0	0	0	3