Variation-Aware Fault Modeling

Hopsch, Fabian; Becker, Bernd; Hellebrand, Sybille; Polian, Ilia; Straube, Bernd; Vermeiren, Wolfgang; Wunderlich, Hans-Joachim

SCIENCE CHINA Information Sciences Vol. 54(9) September 2011

doi: http://dx.doi.org/10.1007/s11432-011-4367-8

Abstract: To achieve a high product quality for nano-scale systems, both realistic defect mechanisms and process variations must be taken into account. While existing approaches for variation-aware digital testing either restrict themselves to special classes of defects or assume given probability distributions to model variabilities, the proposed approach combines defect-oriented testing with statistical library characterization. It uses Monte Carlo simulations at electrical level to extract delay distributions of cells in the presence of defects and for the defect-free case. This allows distinguishing the effects of process variations on the cell delay from defectinduced cell delays under process variations. To provide a suitable interface for test algorithms at higher levels of abstraction, the distributions are represented as histograms and stored in a histogram data base (HDB). Thus, the computationally expensive defect analysis needs to be performed only once as a preprocessing step for library characterization, and statistical test algorithms do not require any low level information beyond the HDB. The generation of the HDB is demonstrated for primitive cells in 45 nm technology.

Preprint

General Copyright Notice

This article may be used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

This is the author's "personal copy" of the final, accepted version of the paper published by *Science China Press and Springer-Verlag Berlin Heidelberg.*

©2011 **Science China Press and Springer-Verlag Berlin Heidelberg**

SCIENCE CHINA
Information Sciences

• RESEARCH PAPERS •

September 2011 Vol. 54 No. 9: 1813–1826
doi: 10.1007/s11432-011-4367-8

Variation-aware fault modeling

HOPSCH Fabian[1], BECKER Bernd[2], HELLEBRAND Sybille[3], POLIAN Ilia[4*],
STRAUBE Bernd[1], VERMEIREN Wolfgang[1] & WUNDERLICH Hans-Joachim[5]

[1]*Fraunhofer IIS/EAS, Dresden* D-010169*, Germany*
[2]*University of Freiburg, Freiburg* D-79110*, Germany*
[3]*University of Paderborn, Paderborn* D-33098*, Germany*
[4]*University of Passau, Passau* D-94023*, Germany*
[5]*University of Stuttgart, Stuttgart* D-70569*, Germany*

Received March 21, 2011; accepted June 21, 2011

Abstract To achieve a high product quality for nano-scale systems, both realistic defect mechanisms and process variations must be taken into account. While existing approaches for variation-aware digital testing either restrict themselves to special classes of defects or assume given probability distributions to model variabilities, the proposed approach combines defect-oriented testing with statistical library characterization. It uses Monte Carlo simulations at electrical level to extract delay distributions of cells in the presence of defects and for the defect-free case. This allows distinguishing the effects of process variations on the cell delay from defect-induced cell delays under process variations. To provide a suitable interface for test algorithms at higher levels of abstraction, the distributions are represented as histograms and stored in a histogram data base (HDB). Thus, the computationally expensive defect analysis needs to be performed only once as a preprocessing step for library characterization, and statistical test algorithms do not require any low level information beyond the HDB. The generation of the HDB is demonstrated for primitive cells in 45 nm technology.

Keywords process variations, test methods, statistical test, histogram data base

Citation Hopsch F, Becker B, Hellebrand S, et al. Variation-aware fault modeling. Sci China Inf Sci, 2011, 54: 1813–1826, doi: 10.1007/s11432-011-4367-8

1 Introduction

Nano-scale integrated circuits suffer both from high defect densities and from increasing parameter variations [1]. On the one hand, defect-oriented testing tries to overcome the deficiencies of classical fault models by extracting the behavior of library cells in the presence of defects and using this input for automatic test pattern generation (ATPG) at the gate level [2]. For cell characterization well-known techniques such as inductive fault analysis [3, 4] or inductive contamination analysis [5] can be used.

On the other hand, process variations have led to a paradigm shift in design methods towards what is known as statistical design [6], as well as to the development of novel, variation-tolerant architectures [7]. However, variation-aware testing is particularly challenging. While classical test approaches rely on a clear distinction between the "fault free" and the "faulty" case, parameter variations can change the impact of a defect, and also the parameter variations themselves can lead to an unacceptable

*Corresponding author (email: ilia.polian@uni-passau.de)

© Science China Press and Springer-Verlag Berlin Heidelberg 2011

behavior [8]. Thus the term "fault coverage" is no longer meaningful. Instead a test must be able to screen out defects for a maximum number of parameter combinations, which is reflected by new test quality metrics as "test robustness" [9] or "process coverage" [10]. As a basis for this new approach to testing, both the impact of parameter variations and the impact of defects must be characterized by appropriate statistical models.

Extensive literature is available on modeling process variations [11–17]. In the context of testing, a number of publications deal with analog and mixed-signal circuits [18–25]. For digital circuits most approaches either restrict themselves to special classes of defects [2, 9] or assume given probability distributions for parameter variations and defect impacts [10, 26, 27]. The authors of [2] analyze layout based intra-cell faults for an industrial library of digital standard cells and designs to develop and apply a new ATPG. As they only target faults, which do not need any sequential test patterns, analog DC fault simulation is sufficient to generate a set of voltage oriented cell-aware fault detection matrices as input for the ATPG algorithm. Ingelsson et al. [9] focus on resistive bridging faults under process variations.

Liou et al. [26] incorporate statistical information into a static and dynamic timing analysis tool. Their approach calculates delay distributions in the circuit and can be used to select appropriate paths for delay testing. Yilmaz et al. [27] calculate the probabilities that signal transitions will fail to propagate through logic gates within a given time limit. These probabilities are efficiently determined for all the lines in the circuit. Xiong et al. [10] model the variations distributed over the paths in the circuit as random variables for path slacks.

To bridge the gap between low level defect information and the statistical analysis on higher levels of abstraction, the work presented in this paper combines the concepts of defect-oriented testing with statistical library characterization. We present a systematic approach[1)] for variation-aware fault modeling for a primitive cell library. For every library cell possible physical defects are represented at the electrical level. To analyze the impact of a defect under process variations, electrical fault simulations with randomly changing circuit parameters are performed. A defect can affect the delay of a library cell or lead to a static fault, which can be viewed as an infinite delay. As a result of this Monte Carlo process for each cell a delay distribution is obtained for each defect and also for the defect-free case. This allows distinguishing between the effects of process variations on cell delays, defect-induced cell delays, and the combinations of both effects.

To provide a suitable interface for fault simulation and test generation tools at higher levels, the distributions are represented by histograms, which are stored in a histogram data base (HDB). The HDB generation step is computationally expensive, but it has to be performed only once as a preprocessing step for library characterization. Statistical test algorithms do not require any lower-level data beyond the histograms in the HDB. This separation is similar to mixed-level fault simulation approaches from the past [28] (these approaches did not incorporate process-variation data). It is also useful for handling the intellectual-property issues in a distributed design, manufacturing and test flow, as the test pattern generation can be done using the HDB only and no sensitive technology data must be given to the entity in charge of preparing the test sets.

The remainder of the paper is organized as follows. Although the focus of this paper is on the generation of the HDB, statistical test algorithms are briefly sketched in the next section to clarify the intended usage of the data in the HDB. Section 3 outlines the procedures used to generate the HDB and demonstrates the application of the concepts in case of a NAND2 gate. Section 4 discusses the obtained results, and finally section 5 concludes the paper.

2 Statistical test algorithms

The proposed approach for variation-aware fault modeling has been developed within the framework of the project RealTest, which addresses the test of nano-scale systems and aims at integrating statistical modeling into test algorithms, whereby a special focus is put on the test of variation-tolerant architectures. This section briefly summarizes the global view and shows the interfaces between the HDB and the test

1) A preliminary version has been presented at the Asian Test Symposium 2010.

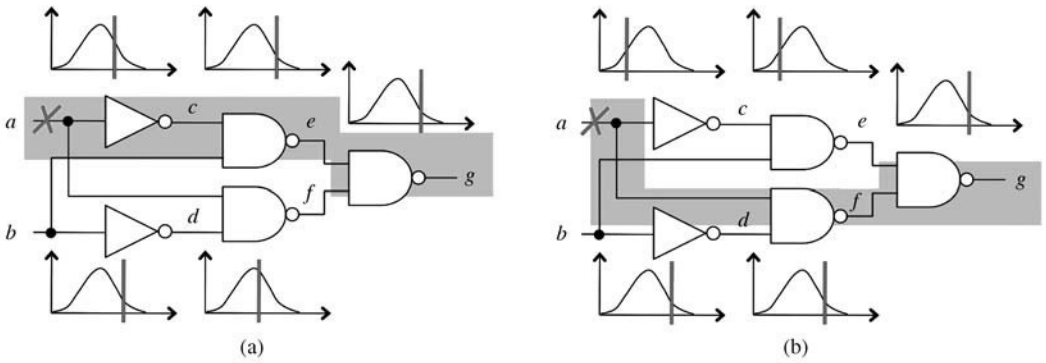

Figure 1 Fault detection under parameter variations.

procedures at higher levels.

As pointed out in the introduction, parameter variations change the behavior of defect-free cells as well as the behavior in the presence of defects, and a clear distinction between fault free and faulty circuits is no longer possible. To reflect the impact of parameter variations, a circuit is called robust for a given range of parameters $P = P_1 \times P_2 \times \cdots \times P_N$, if its functional and delay specifications are fulfilled for all parameter values from that range. For statistical test algorithms the interaction of process variations, defects and delays is of special interest. A defect in a primitive cell or an interconnect may lead to an increased delay or to a static fault (e.g., a stuck-at fault), where the latter can be considered as an instance of an infinite defect-induced delay. The interaction of the defective cell with its surrounding cells and interconnects, which are affected by process variations, may not allow unambiguous decisions whether a given defect is "critical" and should be targeted during test generation. To obtain an appropriate testability assessment on higher levels of abstraction, it is necessary to use probability density functions to describe the behavior of the affected cells.

While classical algorithms are based on the notion of fault coverage as defined in (1), variation-aware testing must be based on new coverage metrics measuring the number of parameter combinations for which the test is effective.

$$FC = \#\text{detected faults } /\# \text{ modeled faults.} \tag{1}$$

If delay faults with continuous sizes D are considered as in [29] and f_{ds} denotes the density function for defect sizes, eq. (1) becomes

$$FC = \int FC(D) f_{ds}(D) dD. \tag{2}$$

Under process variations, the fault detection depends on the parameter configuration

$$p = (p_1, p_2, \ldots, p_N) \in P,$$

and the fault coverage is determined by

$$FC(D) = \int_{p \in P} FC_p(D) f_{pc}(p) dp, \tag{3}$$

where $f_{pc}(p)$ is the probability that the parameter configuration p actually appears in a manufactured circuit instance. In contrast to the conventional fault coverage, i.e., the percentage of faults detected by a test set in a representative circuit with fixed parameter values, eq. (3) describes the percentage of the manufactured instances of the circuit in which the test set detects a given fault. Statistical ATPG must try to maximize this number and generate (compact) test sets identifying the fault in as many valid circuits as possible. This problem is illustrated with the help of Figure 1.

The circuit in Figure 1 implements an EXOR function using NAND2 gates. To detect a delay fault on input line a, conventional delay test generation would try to propagate a transition along the longest path (a, c, e, g) in the circuit using the pattern sequence $01 \rightarrow 11$, i.e., input a switches from 0 to 1 and input b remains at the non-controlling value 1. For variation-aware testing the delay distributions of

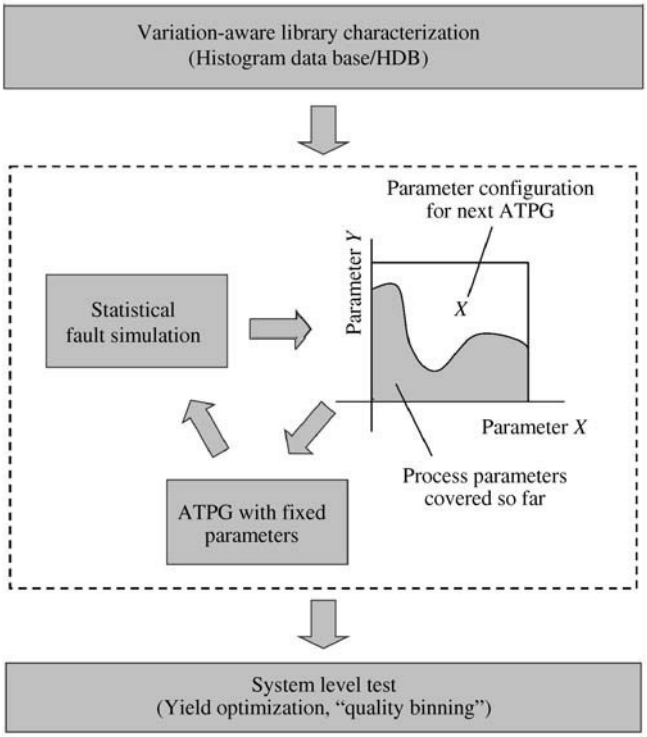

Figure 2 Overview of the statistical test flow.

the cells have to be taken into account. For a circuit instance with delays as shown in Figure 1(a) the path (a, c, e, g) is actually the longest path, and the test is a valid test for the delay fault on input line a. However, if the actual delays in a circuit instance assume the values as shown in Figure 1(b), then the longest path is (a, f, g) and the test is no longer valid. Instead, the test sequence $00 \rightarrow 10$ will detect the fault. To maximize the coverage as defined in eq. (3), a test set for this circuit must include both patterns, i.e., $T = \{01 \rightarrow 11, 00 \rightarrow 10\}$. This example also clearly shows that the concept of robust delay tests, which detect a delay fault independent of other circuit delays, is of limited use under parameter variations [30].

Figure 2 shows a possible iterative procedure to solve the problem of test generation in this case. Statistical fault simulation determines the parameter range covered by the test patterns generated so far, and a variation-unaware ATPG is invoked with fixed parameter values to cover a further parameter set in the range. This is iterated until an acceptable coverage of the complete range is achieved and can be followed by the compaction of the obtained test set. The data from the histogram data base (HDB) depicted at the top of Figure 2 play a crucial role in these analysis steps.

Based on the knowledge of process-induced variations in the individual circuit components and using high-quality variation-aware test patterns, it is possible to separate the different manufactured instances of a circuit into classes or "bins" according to the frequency or voltage they can handle ("frequency binning" [31] and "voltage binning" [32]), thus maximizing yield. The emerging concept of "quality binning" takes into account the circuit's robustness, i.e., its expected ability to tolerate the effects of aging or to recover from transient faults. Using the HDB data, it is possible to judge whether the circuit is sufficiently robust, such that its deteriorations will not manifest themselves as observable defects. These system-level approaches based on the HDB data and the outcome of the variation-aware test algorithms are shown at the bottom of Figure 2.

3 Histogram data base generation

To describe the generation of the histogram data base, this section first briefly outlines the overall characterization flow and then describes the steps in detail for a NAND2 gate.

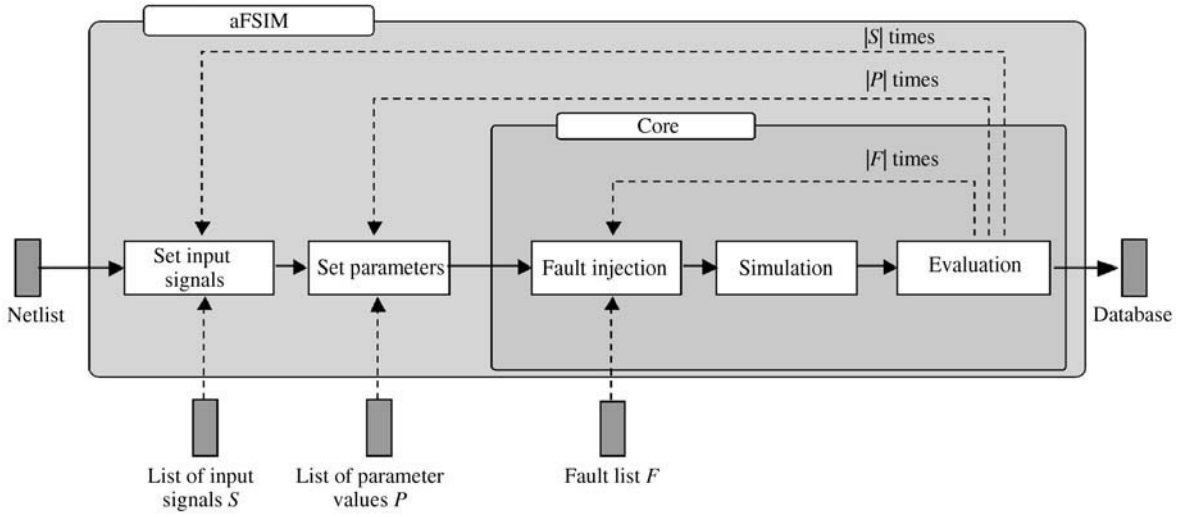

Figure 3 Diagram of the fault simulator aFSIM. Simulations are automatically distributed.

3.1 Characterization flow overview

The primitive-cell characterization is done by means of Monte-Carlo simulations at the electrical level. For this purpose the analogue fault simulator aFSIM [33] has been extended to allow simulations for varying configurations of process parameters. The simulator takes the following information as inputs:

- a transistor-level netlist,
- a fault list F modeling the effects of realistic physical defects at the electrical level,
- a list of N-tuples $(p_1, p_2, \ldots, p_N) \in P$ representing the parameter configurations to be considered,
- a list of input signals S to be used as test stimuli, and
- an evaluation criterion describing the properties of the circuit to be observed.

The simulator aFSIM is able to model various types of faults including modification, replacement or removal of any circuit element as well as addition of new elements. For the list of input signals, all kinds of analogue and digital sources are supported by aFSIM. The simulator can employ different criteria to classify circuits as defective or defect-free, but in this work we use only the cell's delay for classification.

The simulator aFSIM automatically computes the fault effects excited by a test stimuli for a given manufacturing process parameter configuration. For this purpose, it injects a fault $f \in F$ into the netlist simultaneously with a parameter configuration $(p_1, p_2, \ldots, p_N) \in P$ and hands the resulting circuit to an electrical-level simulator such as SPICE, Spectre, Eldo or TITAN. A Monte-Carlo fault simulation across the parameter space P consists of $|P|$ repetitions of the same simulation for different parameter configurations. Overall, $|S| \cdot |P| \cdot |F|$ modified netlists are created and simulated to generate the complete HDB. To deal with the high computational complexity of this procedure, the simulations are automatically distributed on a high-performance-computing (HPC) cluster. The overall architecture of the simulation tool is shown in Figure 3. The subsequent sections provide details on the fault model used, the modeling of manufacturing process parameter variations, and the input sequences considered.

In order to obtain accurate responses of a cell, it must be driven by signals similar to the ones it will see in an actual circuit. Therefore, we embed each cell, as shown in Figure 4 for a NAND2 cell, which consists of four transistors. Both inputs of the NAND2 cell are driven by a pair of inverters connected in series. For example, consider input transition $01 \rightarrow 11$, i.e., input IN1 sees a rising transition while input IN2 remains at the non-controlling logic value. The inverters ensure that the transition at input IN1 has a realistic time constant. A capacitive load C_L is attached to the cell's output to represent subsequent logic. We used the Nangate 45 nm Open Cell Library (OCL) [34] for all experiments reported in this paper and set C_L=0.4 fF, a value from OCL statistics [34]. Next, we will describe fault injection in detail, followed by a discussion of how statistical process variations are modeled and which input transitions are included in the HDB.

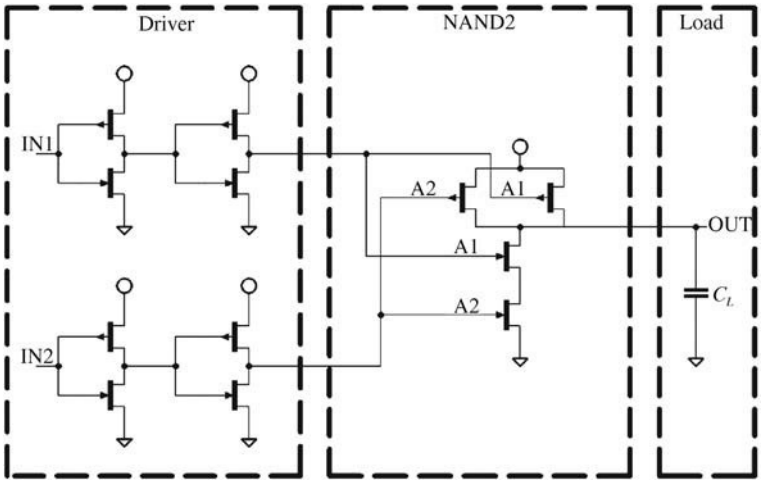

Figure 4 NAND2 gate under characterization with drivers and load.

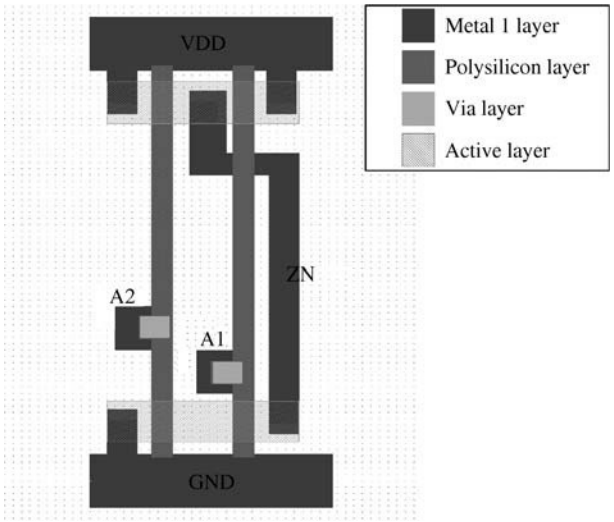

Figure 5 Layout of the NAND2 gate under characterization.

3.2 Fault injection

We consider a fault list which contains intra-cell shorts (i.e., shorts between wires within the cell) as well as opens of single wires. Note that no faults are injected in the surrounding logic (the inverter pairs and the capacitive load), which is part of the simulation setup and is not manufactured. The realistic fault list for the NAND2 gate is generated based on the parasitic resistors and capacitances in the transistor-level netlist extracted from the cell's layout. The layout of the NAND2 cell is shown in Figure 5.

The figure shows the metal1 layer, the polysilicon layer, the via layer, and the active region, which forms two pMOSFETs (in the upper part of the figure) and two nMOSFETs (in its lower part). In order to keep the figure readable, *n*-well, *p*-well, *n*-implant and *p*-implant are not depicted. The signal lines (the inputs A1 and A2 and the output ZN) and the power-supply lines VDD and GND are marked in the layout. All outside connections of the cell are on the metal1 layer. The input signals A1 and A2 are passed, through the vias, from the metal1 layer to the polysilicon layer, which leads to the gate-terminals of the four transistors. The output signal ZN is transferred from the transistors to the metal1 layer through two vias.

The transistor-level netlist in Figure 6 shows the four MOSFETs M_M0, M_M1, M_M2 and M_M3 along with all the parasitic resistances and capacitances. There is a total of 27 resistances and 19 capacitances,

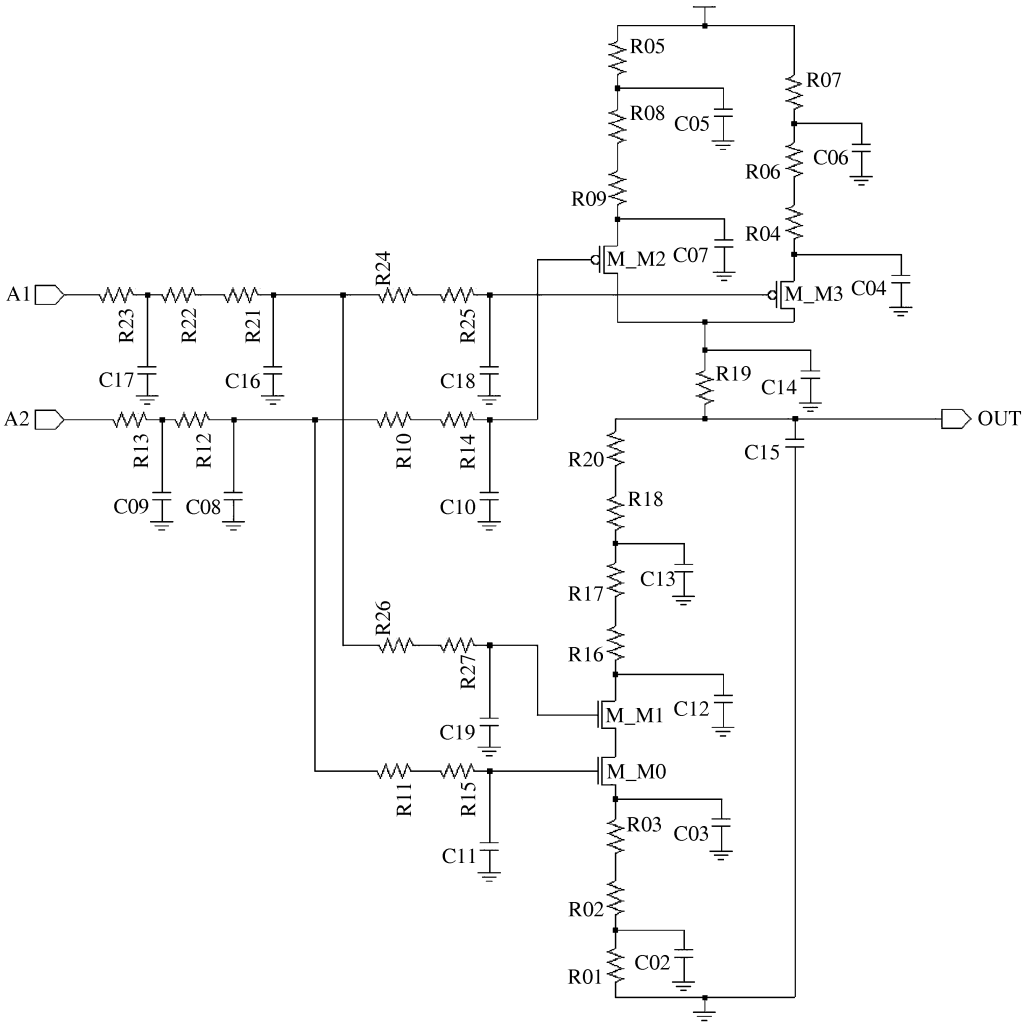

Figure 6 Detailed extracted transistor-level netlist of the NAND2 gate under characterization.

all to ground. For example, consider the line driven by input A1. This line goes through three resistors before it fans out. Each of the resistors corresponds to a different physical object. R23 represents the part of A1 located on the metal1 layer. Its defect-free resistance is approximately 0.1 Ω. R22 is the via between metal1 and polysilicon ($R \approx 25$ Ω), and R21 is the shared part (fanout stem) of the line in polysilicon ($R \approx 3.9$ Ω). The resistors R24 and R25 refer to the fanout branch of line A1 on the polysilicon layer, running to the *p*-channel transistors. The sum of their resistances is about 130 Ω. The sum of resistances of R26 and R27, which represent the fanout branch to the nMOSFETS, is about 20 Ω. Note that this large difference is due to the respective fanout branch being considerably shorter.

To model an open defect, one of the parasitic resistors is replaced by one with a considerably higher resistance. For instance, a defect due to missing conducting material in the via is represented by the resistance of R22 set to, say 1 MΩ. We employ the following reduction technique [35]: If several resistors are connected in series without any fanout in between, their electrical behavior is equivalent and only one of them is considered as a potential open-defect location. For example, even though actual defect mechanisms leading to missing material in the metal1 portion of A1, in the via, and in its polysilicon portion may be totally different, the electrical behavior of a circuit with R23, R22 or R21 set to a large value will be indistinguishable. The NAND2 cell's netlist has 27 parasitic resistors before the above-mentioned reduction; this number becomes 11 after the reduction. We use 10 different defect resistances between 100 kΩ to 100 MΩ, because this range covers both weak and strong open defects [36]. This yields a total of 110 open defects.

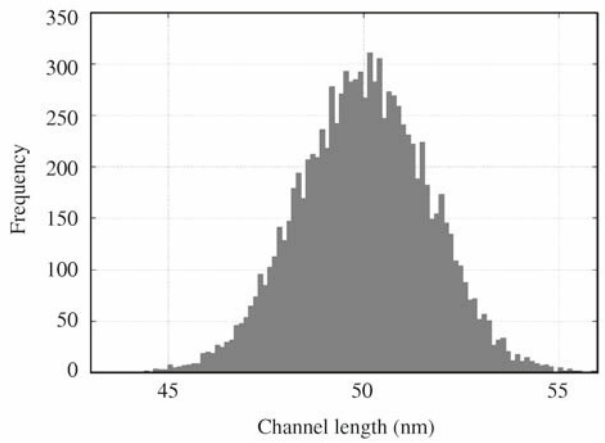

Figure 7 Histogram of the channel length L for μ= 50 nm, 3σ= 5 nm, and 10000 samples.

Shorts are considered between all terminals of the same transistor and between the gate's inputs, which results in 13 possible short locations. To model a short between two wires, a new resistor connected to both of them is inserted. We consider 10 different short circuit resistances between 10 to 15000 Ω, which covers the range of hard and resistive shorts [37]. Altogether, there are 13 short locations and 10 different short circuit resistances between 10 to 15000 Ω. In total, the fault list consists of 130 shorts and 110 opens, i.e., 240 faults.

3.3 Modeling of statistical manufacturing process parameters

The parameter variation model incorporates the following parameters for p- and n-channel transistors, respectively:

- channel length L,
- length reduction parameter L_{INT},
- threshold voltage V_{TH0},
- bulk effect coefficient K_1,
- low-field mobility μ_0,
- junction depth X_{J}, and
- oxide thickness TOX.

Hence, an instance of the cell under characterization is represented by a 14-tuple specifying the parameters. The variations of the channel length L are modeled based on the data provided by an industrial partner. The parameters are assumed to be uncorrelated as no specific information about correlations was available. All considered parameters are assumed to be described by a normal distribution with mean μ and variance σ derived from the OCL data. OCL provides all the parameters for slow, typical, and fast process corner. The mean μ of a parameter is set to this parameter's value in the typical corner. The variance σ of a parameter is set such that this parameter's values in the slow and fast corners are assumed to equal μ–3σ and μ+3σ, respectively. Figure 7 shows a possible histogram for one parameter and 10000 samples.

3.4 Stimuli generation

A combinational cell with n inputs has $2^n \cdot 2^n$ different test sequences of length 2 (test pairs). We consider all test pairs causing a transition on the output of the defect-free circuit. For each such test pair, we record the timing behavior of the defect-free circuit and the defective circuits under all the considered process parameter tuples (configurations) into the histogram database. For example, the NAND2 gate has two inputs. Out of $2^2 \cdot 2^2 = 16$ different test pairs that can be applied to the gate, six resulting in an output transition are considered for fault analysis. Table 1 shows the respective test sequences together with the expected digital responses of the defect-free cell.

Table 1 Simulated input sequences and corresponding output values for the NAND2 gate

Test sequence				Output	
t_n		t_{n+1}		t_n	t_{n+1}
0	0	1	1	1	0
0	1	1	1	1	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	1

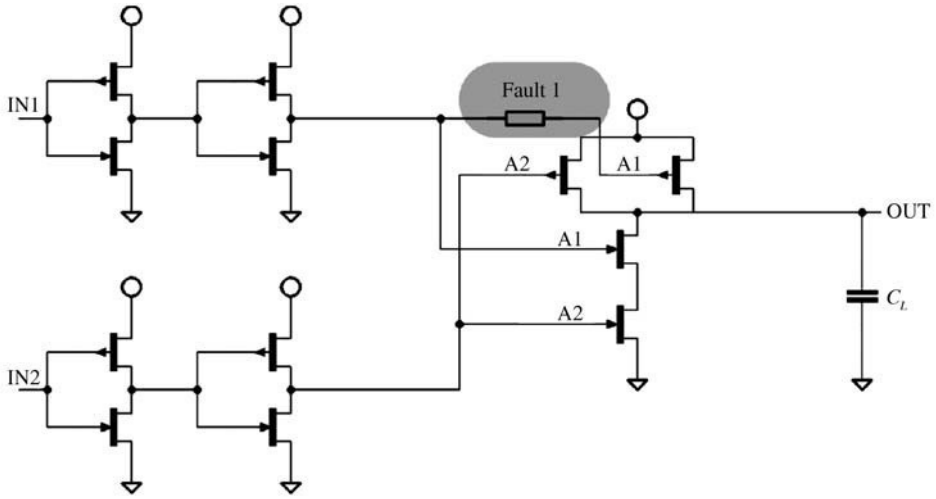

Figure 8 Schematic of the embedded NAND2 gate with injected fault 1.

These six test sequences, modeled as piecewise linear sources with a slew rate of 7.5 ps (a value from the OCL library), are used as input signals for electrical simulations. Recall that the cell under analysis is embedded and the stimuli are applied to the inverters (in the "Drivers" part of Figure 4, which, in turn, generate a realistic waveform at the inputs of the cell under characterization.

For each considered test pair, $|P| = 10000$ process parameter configurations are generated using the model from the previous section. The delay of the defect-free cell under characterization is determined by electrical simulation and the results are aggregated in histogram form and stored in the HDB. The same experiment is repeated for each considered defect. For the NAND2 cell with six considered input sequences ($|S| = 6$), the defect-free cell characterization requires a total of $6 \times 10000 = 60000$ simulation runs. The defective-cell characterization yields $6 \times 240 \times 10000 = 14400000$ simulation runs, because there are $|F| = 240$ modeled defects (11 open locations and 13 short locations, each for 10 different resistance values).

4 Results

The characterization procedure described in section 3 has been applied to a library containing NAND2, NOR2, and inverter cells. For simulation, time periods of 20 ns have been considered, where the transition of the input signals begins at 10 ns. The delay has been measured as the time the output waveform needs to reach half of the supply voltage in a rising or falling transition [38]. The results are aggregated to histograms and stored in the HDB. To access a histogram, the user must specify a fault and an input sequence; the information which parameter tuples led to which delays are not stored.

In the following some representative examples are discussed in more detail. Figure 8 shows an open fault in the NAND2 cell under characterization (fault 1), modeled by replacing the parasitic resistor ($R \approx 50\ \Omega$) representing the corresponding wire by a resistor with $R = 500\ \text{k}\Omega$.

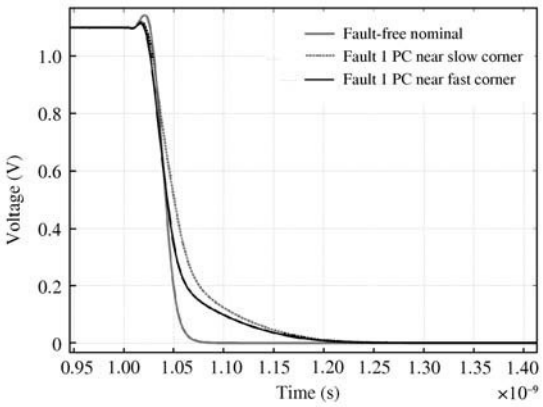

Figure 9 Output waveforms for the NAND2 cell for fault 1 and for input sequence 00 → 11.

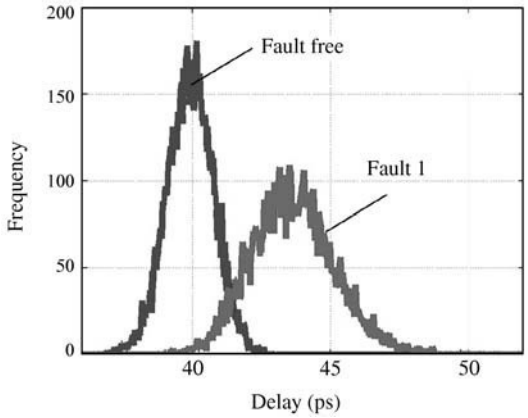

Figure 10 Histograms for the fault free case and for fault 1 obtained for the transition 00 → 11.

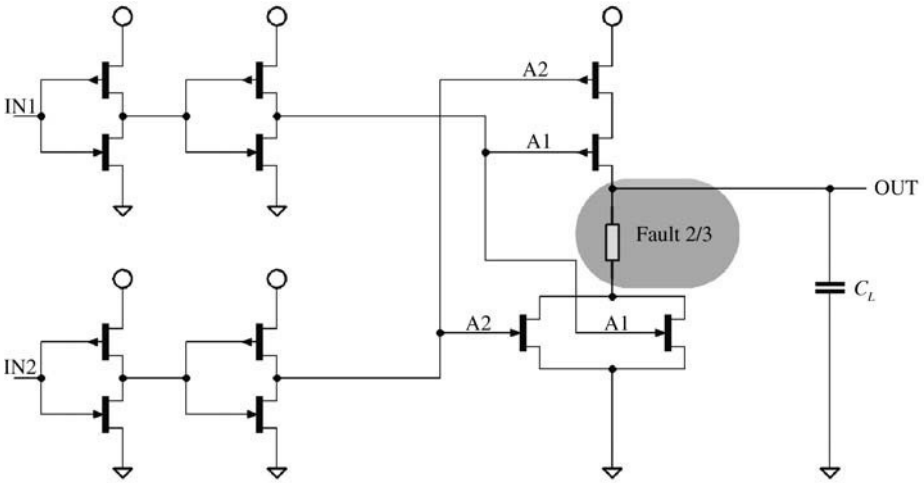

Figure 11 Schematic of the embedded NOR2 gate with injected open (fault 2 and fault 3).

In Figure 9 three different output waveforms for the input sequence 00 → 11 (both inputs switch from 0 to 1) are shown. The fault-free circuit with the nominal parameter configuration is compared to the faulty circuit with parameter configurations near the fast and near the slow corner. It can be observed that all three waveforms represent a falling transition from 1 to 0, but the delays for the faulty circuit are a little bit higher than the delay for the fault-free case. Furthermore, the two faulty cases are characterized by slightly different waveforms.

Overall, the simulations for this fault and this input sequence have shown falling transitions for all parameter configurations. Figure 10 presents the resulting histogram stored in the HBD. An entry in the histogram shows for how many parameter configurations a certain delay occurs. Here both histograms overlap, which means that some manufactured instances of a circuit with fault 1 in a NAND2 cell may fail the test and others may pass. Furthermore it can be seen that the variation of the delay for fault 1 is greater than the delay variation of the fault free circuit.

To further illustrate the sensitivity to input sequences Figure 11 shows an open fault in a NOR2 cell. The fault is analyzed for two different resistances: 250 kΩ (fault 2) and 500 kΩ (fault 3). Furthermore a delay specification of 70 ps maximum is assumed for the NOR2 cell.

Figures 12 and 13 show the histograms for the test sequences 00 → 11 and 10 → 11. For both input sequences fault 2 and fault 3 result in delayed transitions. However, for input sequence 00 → 11, the delay induced by fault 2 is still acceptable for part of the parameter configurations. Thus input sequence 00 → 11 cannot provide complete parameter coverage for fault 2, if the specification of 70 ps is used to

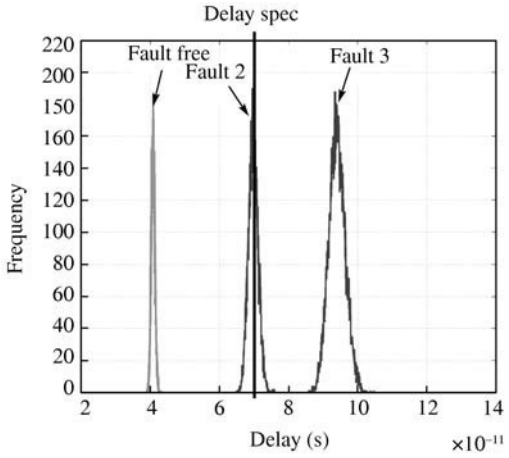

Figure 12 Histograms for faults 2 and 3 for input sequence 00→ 11.

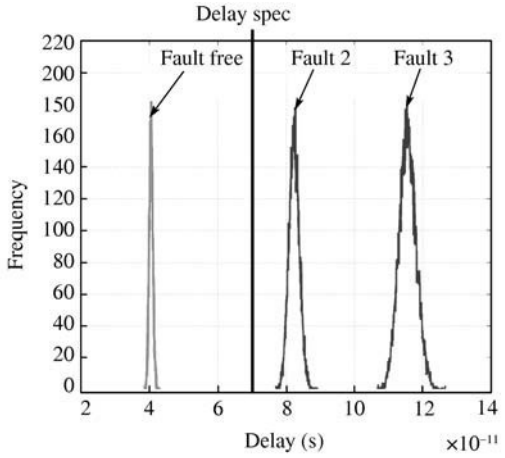

Figure 13 Histograms for faults 2 and 3 for input sequence 10→11.

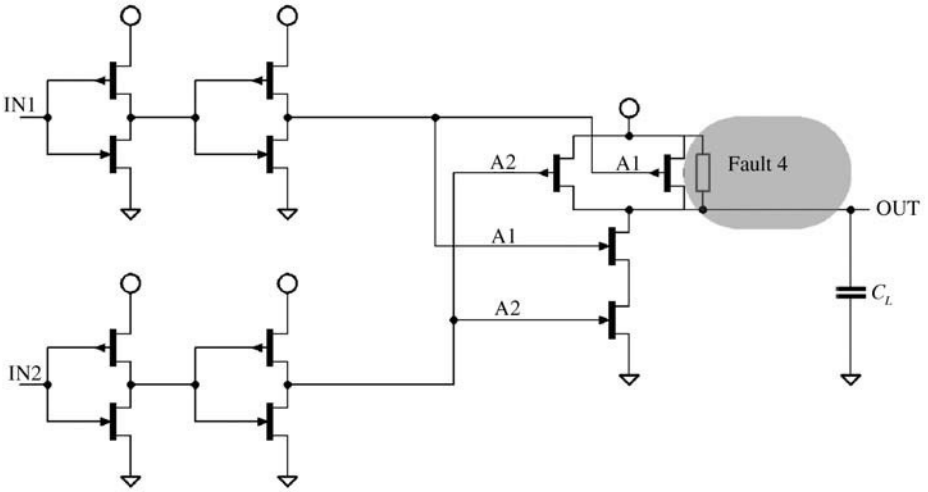

Figure 14 Schematic of the embedded NAND2 gate with injected fault 4.

distinguish between fault free and faulty devices. In contrast to that, input sequence 10 → 11 provides complete parameter coverage for both faults. In the general case, however, several test sequences may be necessary to cover the parameter space for one fault as shown in the introductory example of Figure 1.

The histogram for every fault and input sequence can be used as an approximation of the delay distribution across all parameter values. However, there are also faults for which no delay can be determined because no signal transition occurs at the outputs during the observation time. This indicates a static behavior similar to stuck-at faults, which can be interpreted as an infinite delay. For some faults, a finite delay is induced for a number of parameter configurations and an infinite delay for the remaining ones. This is illustrated by fault 4, a 7.5 kΩ short in a NAND2 cell shown in Figure 14.

Here the output waveforms produced by the input sequence 00 → 11 in the faulty case show a significant difference for parameter configurations (PC) near the slow and near the fast corner (Figure 15). While fault 4 leads to a delayed falling transition near the fast corner, it results in an infinite delay near the slow corner, because half of the supply voltage is not reached during the observation time.

To represent such a behavior in the histogram, a special class named ∞ is introduced. In Figure 16, this class represents all parameter tuples for which fault 4 results in static behavior.

Out of 14400000 simulations for the NAND2 gate a delay could be determined in 67.7% of the cases. For the remaining 32.3% no signal transition occurred during the observation time. The aggregation

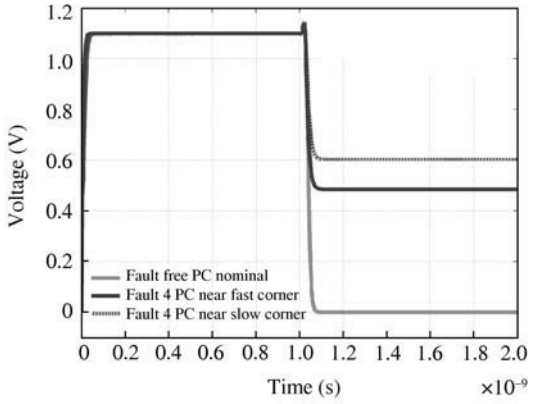

Figure 15 Output waveforms for the NAND2 cell for fault 2 and for input sequence 00 → 11.

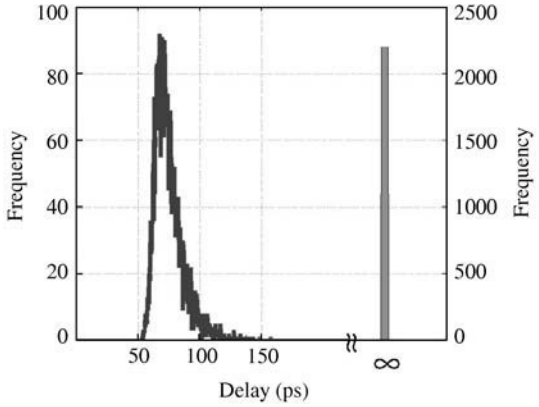

Figure 16 Histograms for fault 4 showing finite and infinite delays.

Table 2 Simulation effort for some primitive cells

Cell	Input sequences	Injected faults	Simulations
INV	2	150	3000000
NAND2	6	240	14400000
NOR2	6	240	14400000

Table 3 Static vs. dynamic effects in histograms

Cell	Effects observed		Classes present in histograms		
	Dynamic	Static	No ∞	Only ∞	Both
INV	55.2%	44.8%	51.0%	37.7%	11.3%
NAND2	67.7%	32.3%	64.0%	26.7%	9.3%
NOR2	62.2%	37.8%	58.8%	32.7%	8.5%

across the parameter tuples resulted in 1440 single histograms. 64% of the histograms include no element in class ∞, i.e., they describe only dynamic effects. For 26.7% of the histograms, all elements belong to class ∞; these faults have only static effects. The remaining 9.3% histograms have elements both in class ∞ and in other classes.

Finally, the simulation effort and the obtained results for the complete library are summarized in Tables 2 and 3. Table 2 lists the number of simulated input sequences, the number of injected faults and the number of simulations for the analyzed NAND2, NOR2 and inverter (INV) cells. The complete characterization takes about 10 days on an HPC-cluster with 32 nodes.

Table 3 classifies the fault effects into static and dynamic effects as explained above. It can be observed that in the majority of cases a fault only leads to delayed transitions ("no ∞"). In approximately one third of the cases, a fault only leads to a static behavior ("only ∞"). In the remaining cases, a fault can result in both a static and dynamic effect depending on the parameter configuration.

5 Conclusions

Defect-oriented testing for nano-scale systems must incorporate awareness to massive process variations. The proposed approach combines defect-based fault modeling with statistical library characterization. It is based on an enhanced version of the analogue fault simulator aFSIM, which incorporates transistor-level fault modeling and parameter variations. The obtained delay distributions are aggregated to histograms and stored in a histogram data base (HDB). The HDB can be flexibly accessed and efficiently used by

algorithms such as delay fault simulation or ATPG at logic level. This allows to clearly separate the low-level electrical information from test algorithms operating at the higher abstraction levels.

HDB generation has been demonstrated in detail for a NAND2 gate, and it has also been applied to an inverter cell and a NOR2 gate using the data for primitive cells in Nangate 45 nm OCL. The obtained information accurately distinguishes between effects of process induced parameter variations and the effects of manufacturing defects. Furthermore, it has been shown that the same manufacturing defect can lead to a delay fault or to static fault (infinite delay) depending on the parameter configuration.

Acknowledgements

This work was performed within the framework of the project RealTest supported by the German National Science Foundation (DFG).

References

1 ITRS. International Technology Roadmap for Semiconductors. 2009 ed. http://www.itrs.net/Links/2009ITRS/Home2009.htm
2 Hapke F, Krenz-Baath R, Glowatz A, et al. Defect-oriented cell-aware ATPG and fault simulation for industrial cell libraries and designs. In: Proceedings of IEEE International Test Conference, Austin, Texas, USA, 2009. Paper 1.2
3 Ferguson F J, Shen J. Extraction and simulation of realistic CMOS faults using inductive fault analysis. In: Proceedings of IEEE International Test Conference, Washington DC, USA, 1988. 475–484
4 Shen J P, Maly W, Ferguson F J. Inductive fault analysis of NMOS and CMOS circuits. IEEE Des Test, 1985, 2: 13–26
5 Khare J, Maly W. From Contamination to Defects, Faults and Yield Loss. Dordrecht: Kluwer Academic Publishers, 1996
6 Srivastava A, Sylvester D, Blaauw D. Statistical Analysis and Optimization for VLSI: Timing and Power. New York: Springer, 2005
7 Stefano B, Bertozzi D, Benini L, et al. Process variation tolerant pipeline design through a placement-aware multiple voltage island design style. In: Proceedings of Design, Automation and Test in Europe, Munich, Germany, 2008. 967–972
8 Aitken R C. Defect or variation? Characterizing standard cell behavior at 90 nm and below. IEEE Trans Semicond Manuf, 2008, 21: 46–54
9 Ingelsson U, Al-Hashimi B M, Khursheed S, et al. Process variation-aware test for resistive bridges. IEEE Trans CAD Integr Circ Syst, 2009, 28: 1269–1274
10 Xiong J, Shi Y, Zolotov V, et al. Statistical multilayer process space coverage for at-speed test. In: Proceedings of Design Automation Conference, San Francisco, CA, USA, 2009. 340–345
11 DATE Workshop on Process Variability. New techniques for the design and test of nanoscale electronics. In: Proceedings of Design, Automation and Test in Europe, Nice, France, 2009
12 Gizopoulos D, ed. Advances in Electronic Testing: Challenges and Methodologies (Frontiers in Electronic Testing). New York: Springer, 2006
13 Menezes N. The good, the bad, and the statistical. In: Proceedings of International Symposium on Physical Design, Austin, Texas, USA, 2007. 168
14 Sachdev M, Pineda de Gyvez J. Defect-Oriented Testing for Nano-Metric CMOS VLSI Circuits (Frontiers in Electronic Testing). 2nd ed. New York: Springer, 2006
15 Sylvester D, Agarwal K, Shaha S. Variability in nanometer CMOS: Impact, analysis, and minimization. Integrat VLSI J, 2008, 41: 319–339
16 Visweswariah C. Fear, uncertainty and statistics. In: Proceedings of International Symposium on Physical Design, Austin, Texas, USA, 2007. 169
17 Schlichtmann U, Schmidt M, Kinzelbach H, et al. Digital design at a crossroads-how to make statistical design industrially relevant. In: Proceedings of Design, Automation and Test in Europe, Nice, France, 2009. 1542–1547
18 Bounceur A, Mir S, Simeu E, et al. Estimation of Test Metrics for the Optimisation of Analogue Circuit Testing. J Electron Test Theory Appl, 2007, 23: 471–484
19 Devarayanadurg G, Goteti P, Soma M. Hierarchy based statistical fault simulation of mixed-signal ICs. In: Proceedings of IEEE International Test Conference, Washington DC, USA, 1996. 521–527
20 Gomes A V, Voorakaranam R, Chatterjee A. Modular fault simulation of mixed signal circuits with fault ranking by severity. In: Proceedings of International Symposium on Defect and Fault-Tolerance in VLSI Systems, Austin, TX, USA, 1998. 341-348
21 Khouas A, Derieux A. Fault simulation for analog circuits under parameter variations. J Electron Test Theory Appl,

2000, 16: 269–278
22 Liu F, Ozev S. Statistical test development for analog circuits under high process variations. IEEE Trans CAD Integr Circ Syst, 2007, 26: 1465–1477
23 Peralta J, Peretti G, Romero E, et al. A new performance characterization of transient analysis method. Int J Electron Commun Comput Engineer, 2009, 1: 12–19
24 Saab K, Ben-Hamida N, Kaminska B. Parametric fault simulation and test vector generation. In: Proceedings of Design, Automation and Test in Europe, Paris, France, 2000. 650–657
25 Spinks S J, Chalk C D, Bell I M, et al. Generation and verification of tests for analog circuits subject to process parameter deviations. J Electron Test Theory Appl, 2004, 20: 11–23
26 Liou J J, Krstic A, Jiang Y M, et al. Modeling, testing, and analysis for delay defects and noise effects in deep submicron devices. IEEE Trans CAD Integr Circ Syst, 2003, 22: 756–769
27 Yilmaz M, Chakrabarty K, Tehranipoor M. Interconnect-aware and layout-oriented test pattern selection for small-delay defects. In: Proceedings of IEEE International Test Conference, Santa Clara, CA, USA, 2008. Paper 28.3
28 Santos M B, Teixeira J P. Defect-oriented mixed-level fault simulation of digital systems-on-a-chip using HDL. In: Proceedings of Design, Automation and Test in Europe, Munich, Germany, 1999. 549–553
29 Sato Y, Hamada S, Maeda T, et al. Invisible delay quality-SDQM model lights up what could not be seen. In: Proceedings of IEEE International Test Conference, Austin, TX, USA, 2005. Paper 47.1
30 Pramanick A K, Reddy S M. On the detection of delay faults. In: Proceedings of IEEE International Test Conference, Washington DC, USA, 1988. 845–856
31 Zeng J, Abadir M S, Kolhatkar A, et al. On correlating structural tests with functional tests for speed binning of high performance design. In: Proceedings of IEEE International Test Conference, Charlotte, NC, USA, 2004. 31–37
32 Zolotov V, Visweswariah C, Xiong J. Voltage binning under process variation. In: Proceedings of International Conference on Computer-Aided Design, San Jose, CA, USA, 2009. 425–432
33 Straube B, Müller B, Vermeiren W, et al. Analogue fault simulation by aFSIM. In: Proceedings of Design, Automation and Test in Europe Conference and Exhibition-User Forum, Paris, France, 2000. 205–210
34 Nangate 45 nm Open Cell Library. http://www.nangate.com
35 Hillebrecht S, Polian I, Engelke P, et al. Extraction, simulation and test generation for interconnect open defects based on enhanced aggressor-victim model. In: Proceedings of IEEE International Test Conference, Santa Clara, CA, USA, 2008. Paper 33.3
36 Rodriguez-Montanes R, Pineda de Gyvez J, Volf P. Resistance characterization for weak open defects. IEEE Des Test Comput, 2002, 19: 18–26
37 Rodriguez-Montanes R, Bruls E M J G, Figueras J. Bridging defects resistance in the metal layer of a CMOS process. J Electron Test Theory Appl, 2006, 8: 35–46
38 Hirata A, Onodera H, Tamaru K. Estimation of propagation delay considering short-circuit current for static CMOS gates. IEEE Trans Circ Syst I Fund Theor Appl, 1998, 45: 1194-1198