Diagnosing At-Speed Scan BIST Circuits Using a Low Speed and Low Memory Tester

Yoshiyuki Nakamura, *Member, IEEE*, Thomas Clouqueur, *Member, IEEE*, Kewal K. Saluja, *Fellow, IEEE*, and Hideo Fujiwara, *Fellow, IEEE*

***Abstract*—Numerous solutions have been proposed to reduce test data volume and test application time during manufacturing testing of digital devices. However, time to market challenge also requires a very efficient debug phase. Error identification in the test responses can become impractically slow in the debug phase due to large debug data, slow tester speed, and limited memory of the tester. In this paper, we investigate the problems and solutions related to using a relatively slow and limited memory tester to observe the at-speed behavior of fast circuits. Our method can identify all errors in at-speed scan BIST environment without any aliasing and using only little extra overhead by way of a multiplexer and masking circuit for diagnosis. Our solution takes into account the relatively slower speed of the tester and the reload time of the expected data to the tester memory due to limited tester memory while reducing the test/debug cost. Experimental results show that the test application time by our method can be reduced by a factor of 10 with very little hardware overhead to achieve such advantage.**

***Index Terms*—Error analysis, fault diagnosis, self-testing.**

I. Introduction

BUILT-IN self-test (BIST) has become one of the major test techniques for today's large scale and high speed designs. Pseudo-random BIST designs are the most widely used due to their relative simplicity and low cost [1]. Since BIST compacts test responses, BIST requires only small tester memory and it can perform at-speed test even if the tester frequency is substantially lower than the frequency of the circuit during test.

On the other hand, BIST causes problems in diagnosis due to its compacted responses. Indeed, pass/fail information obtained from a BIST response analyzer is insufficient for diagnosis. Two kinds of information are required to identify a fault in a circuit under test (CUT), namely the time information, and the space information. Using time information, fault diagnosis can be performed for a given fault model by methods using dictionary or fault simulation [2]. Using space information, diagnosis can be performed by cone of logic methods [3]. High resolution diagnostic for a given fault model can be achieved by diagnosis techniques combining space information with time information [4], [5]. A number of methods to identify space information have been proposed, especially for scan-based BIST architecture [6]–[11], however, only a few practical techniques have been developed to identify time information.

Some of the existing techniques are based on signature analysis using cycling register [11], [12] and error correcting codes [13]. These methods compact the complete test response into one signature and attempt to identify errors from the signature. Since they observe signature only once, they require only small tester memory and they are also usable even if the circuit frequency is much higher than the tester frequency. However, for large number of error bits, say r errors, they need as many as r-LFSRs or signature registers and may have over 40% diagnostic aliasing if the actual number of errors is higher than r [13]. Thus, they either have poor diagnostic resolution or require impractically high hardware overhead to achieve maximum diagnostic resolution. An alternative approach trades off overhead for time by repeating the test sequence and compacting it at each iteration into a different signature [14]. Thus, instead of using r-LFSRs, the test sequence is repeated r times using programmable LFSR to identify r errors. Since it is mathematically equivalent to [13], diagnostic aliasing is the same as using r-LFSRs. Thus, identifying all the errors requires repeating the test sequence an impractically large number of times.

Techniques that use two phases for diagnosis have also been proposed [15]–[17]. During the first phase, intermediate signature is checked a few times during test in order to narrow down the failing candidates within some windows of fixed or variable size. The failing patterns are then identified inside the windows by applying the corresponding patterns one at a time [15], [16]. These methods use small hardware overhead and/or reduce test application time but they assume the existence of a mechanism to observe the at-speed behavior inside of failing windows. Enhancement of these methods has also been studied using multiple signature analyzers [17], but they do not achieve maximum diagnostic resolution.

A commonly used diagnosis technique that requires and collects the failing space and time information, without compacting responses, during the diagnosis phase [18] suffers from the following two problems. 1) it requires the circuit to operate at the tester frequency during test; therefore, the faults that affect only at-speed operation may not be diagnosable. Note that in at-speed scan based BIST environment, the circuit needs to be operated in at-speed conditions only for the capture cycles, and it needs not operate in at-speed conditions in the shift cycles. However,

Manuscript received October 21, 2006; revised January 30, 2007. This work was supported in part by the Japan Society for the Promotion of Science (JSPS) under Grants-in-Aid for Scientific Research B 15300018, JSPS Invitation Fellowship for Research L04509, and by the 21st Century Center of Excellence (COE) Program (Ubiquitous Networked Media Computing).

Y. Nakamura is with Design Systems Division, NEC Electronics Corporation, Nakahara, Kawasaki 211-8668, Japan (e-mail: y.nak@necel.com).

T. Clouqueur was with Nara Institute of Science and Technology, Nara 630-0192, Japan. He is now with Advanced Micro Devices Inc., Boxborough, MA 01719 USA (e-mail: thomas.cloucueur@amd.com).

K. K. Saluja is with Department of Electrical Engineering, University of Wisconsin-Madison, WI 53706-1691 USA (e-mail: saluja@ece.wisc.edu).

H. Fujiwara is with the Graduate School of Information Science, Nara Institute of Science and Technology, Kansai Science City, Nara 630-0192, Japan (e-mail: fujiwara@is.naist.jp).

Digital Object Identifier 10.1109/TVLSI.2007.899235

to contain the testing time, it is important that we operate shift cycles as fast as possible unless such operation can cause critical damage for layout or power reasons. We must also point out that in a BIST environment the frequency of shift cycles is not limited by the tester frequency because scan outs are not observed directly by the tester in product testing phase. Therefore, frequency of shift operation tends to exceed the tester frequency. In addition, the ratio of a frequency of capture clock cycle and frequency of shift cycle is usually fixed at BIST design phase, and it cannot be changed during testing. Therefore, slow down of frequency of shift cycles to meet the tester limit frequency, will also slow down the capture cycle frequency, thus making at-speed diagnosis impossible. 2) this method requires all expected responses of scan cells to be loaded into the tester. Clearly one of the reasons to move to BIST environments is to contain the cost of testing by employing low cost testers. However, the low cost testers are unlikely to have sufficient memory to store all expected scan cell data of all BIST sequences [20]. Therefore, expected data in such a scheme must be reloaded many times to the tester memory. Clearly, this can be very time consuming. The method proposed in [19] addresses the first problem, but the second problem remains.

In this paper, we investigate methods to identify every error occurrence in at-speed scan-based BIST environment. Every error can be identified even if the circuit test frequency is higher than the tester frequency. In addition, unlike the conventional approaches, we also consider tester loading time in optimizing the overall diagnosis effort. A preliminary version of this paper was presented at the 19th Asian Test Symposium [21]. This paper is organized as follows. In Section II, for the sake of completeness of this paper, we briefly introduce a procedure to identify every error without any aliasing in at-speed scan BIST environment proposed in [19]. In Section III, we formulate the problem of identifying every error occurrence in minimum test time, including tester loading time, and propose a method to reduce test application time by pattern grouping and using signature analyzers. In Section IV, we analyze the test time of the method proposed in Section III to find the optimal group size. In Section V, we show the effectiveness of our method through experimental results and discuss the relationship between the error probability and optimal group size. Section VI summarizes the conclusions of our analysis.

II. Observing Responses by a Slow Speed Tester

Before providing a formulation of the problem, we describe the method given in [19] that can identify all failing responses by observing scan outputs even if the CUT test frequency is faster than the maximum tester frequency (in sequel, we will call the maximum test frequency *tester frequency limitation*). Figs. 1–3 demonstrate this through an example. When the CUT clock period is 2 ns and the tester observing period is 6 ns, a tester can observe only 1/3 of the responses. Thus, it will not be able to identify all possible failing responses in one BIST sequence. Nevertheless, there is a way to observe all responses without adding any extra hardware. We assume that the pattern generator (PG) is reset to the initial state after generating the last pattern. If the BIST sequence is 17 cycles long, as shown in

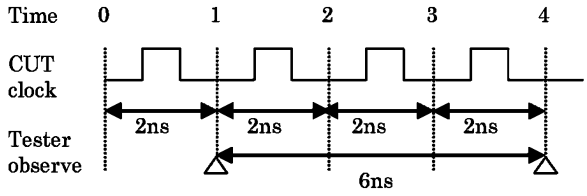

Fig. 1. CUT and observe intervals.

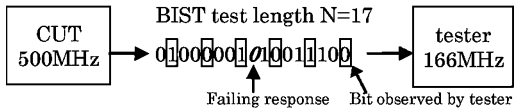

Fig. 2. Scan out and observed results (error unobservable).

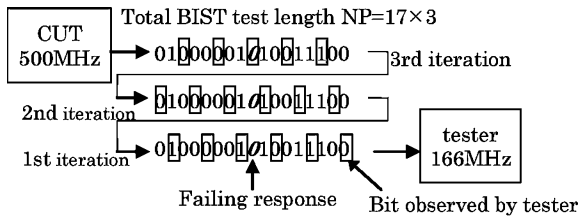

Fig. 3. Multiple iteration observation (successful).

Fig. 3, every cycle can be observed by repeating the BIST sequence three times (i.e., applying 51 clocks). During the first sequence, the tester observes response bits $0, 3, \ldots, 15$. Then bits $1, 4, \ldots, 16$ are observed during the second sequence and bits $2, 5, \ldots, 14$ during the third sequence. Note that such a method may not always allow observing all bits by simple repetition of the sequence. For example, if the length of the BIST sequence is 18, the tester cannot observe every response by simply repeating the sequence since only bits $0, 3, \ldots, 15$ are observed repeatedly. Let N be the length of the BIST sequence, f_c be the clock frequency of the CUT, and f_t be the tester frequency limitation, then the conditions to observe all responses is $\gcd(N, P) = 1$, where $P = f_c/f_t$. The tester can observe all responses by applying the BIST test sequence P times, while observing its response at every time period P. If N and P are not co-prime then N or P or both can be adjusted to make them co-prime. It is shown in [19] that increasing only the length of the BIST sequence N, by inserting no more than P additional dummy clock cycles, achieves minimum test application time in general. A formal description of this problem and its proof are given in Appendix II.

We can identify all erroneous responses with a slow speed tester using this method, albeit requiring all expected responses of scan cells to be loaded into the tester. However, often testers do not have enough memory to store all expected scan cell data of all BIST sequences. Therefore, we must reload the expected data many times to the tester, and this can be very time consuming.

III. Using Signature Analyzers

A. Problem Formulation

In this section, we formulate the problem of identifying failing responses in minimum test time. We first identify some

characteristics of the diagnosis and production testing processes.

Diagnosis can be performed for devices that did not pass the production test or devices that passed the production test and were found to be faulty in the field. In each case, testing during diagnosis should be performed at the speed that resulted in the failure of the device.

Another characteristic of the diagnosis process is that the quality of diagnosis is far more important than any other factor, including the test application time. Thus, the formulation we are interested in this paper is to optimize the test application time subject to meeting the given quality of diagnosis. However, the fault diagnosis must not be overloaded by the error information to achieve the required diagnostic resolution. It is imperative that the reported (required) number of errors be limited to achieve the targeted diagnostic resolution.

Our formulation concerns the error identification of a scan-based BIST circuit. We constrain BIST to operate at-speed during diagnosis. In the at-speed BIST environment, we assume that the CUT scan out operates at frequency f_c, whereas the tester has a frequency limitation and cannot operate at a frequency higher than f_t, such that $f_t < f_c$. Note that f_c can be slower than scan capture clock frequency which should be operated at-speed frequency. We also constrain the number of errors to be identified to E, and an error-free response must not be identified as erroneous. The objective of the problem is to minimize the test time while meeting the error identification requirement. Note that the tester loading time should be included in the test time as mentioned in Section II.

B. Signature Analyzers for Error Identification

The method introduced in Section II does not use signature analyzers. Therefore, it can achieve maximum diagnostic resolution without aliasing; however, the test time, including tester loading time significantly increases when the test sequence becomes very long.

On the other hand, error detection methods using signature analyzer(s) require less tester memory, i.e., fewer reloadings, therefore, test time will be much shorter than the method introduced in Section II. Note that the methods using signature analyzers cause diagnostic aliasing and it is undesirable when an erroneous response is miss-identified as error-free since it can lead to misdiagnosis. Also, as argued in the previous subsection, it is important that the fault diagnosis algorithm is not overwhelmed by excessive error information. Therefore; it will be acceptable to base a diagnostic decision when sufficiently many erroneous responses for diagnosis are identified.

In this paper, we propose an error identification procedure that uses signature analyzers in two phases. During the first phase, the intermediate signature is checked in order to narrow down the error candidates within some windows [15]–[17]. The failing responses are then identified inside the windows during the second phase using the method introduced in Section II for at-speed BIST environment. This study aims at minimizing the test time, in particular by determining the optimal window size for use in phase one.

1) Two-Phase Error Identification: To enable two-phase error identification, all the BIST sequences are divided into groups. Each pattern group includes the seed for the pattern generator that consists of the state of the pattern generator at the end of the previous group. Each pattern group also includes the expected signature for the group.

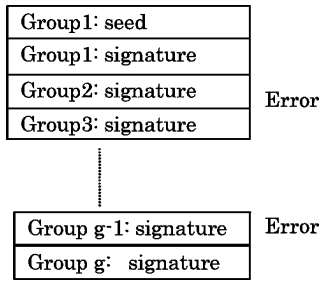

Fig. 4. Test data of first phase.

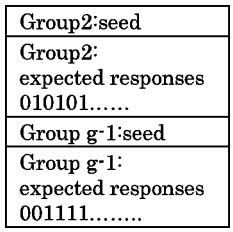

Fig. 5. Test data of second phase.

The first phase identifies erroneous groups. The seeds of the first group and all expected signatures are loaded onto the tester. The test pattern is then generated by the pattern generator and the scan out responses are compacted into signature analyzers. The erroneous pattern groups are identified by observing the results of the signature analyzers. The group size should be large enough to reduce the test data in this first phase.

Let us explain this through an example as shown in Fig. 4. The test data is divided into g groups. Notice that the seed of PG is needed only for the first group since all groups are tested consecutively. In this example, the test results of the second and $(g-1)$th groups are found to be erroneous. Thus, the error bits would need to be identified only in these two groups during the second phase. The second phase identifies erroneous bits in the erroneous pattern groups identified in the first phase. The seeds and the expected responses of the erroneous pattern groups, which are identified in the first phase, are loaded onto the tester and by using error identification procedure proposed in Section II all errors, in each error group, can be identified. For the example shown in Fig. 4, test data of the second group and the $(g-1)$th group are loaded onto the tester as shown in Fig. 5 and all responses are observed to identify the errors.

Note that the test data size for error identification is much smaller than the prior approach explained in Section II since only the test data of erroneous groups are loaded onto the tester.

Some erroneous responses may be dropped due to aliasing of signature analyzers in the first phase; however, any error-free response will not be identified as erroneous, thus satisfying the constraints specified in the problem formulation.

In the first phase, larger groups result into a larger reduction in the test data, which reduces test time including tester loading time. But, larger groups in the first phase imply increased error

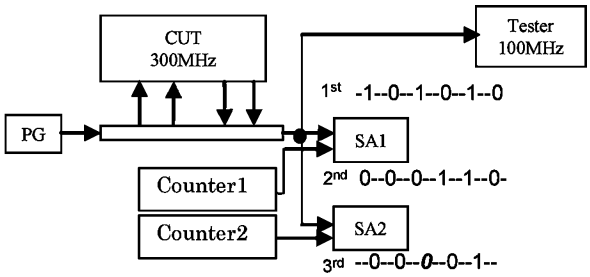

Fig. 6. Diagnosis with error detector signature analyzers.

probability for each pattern group. This, in turn, will cause larger test data in the second phase Therefore, there is an optimal group size for minimum test time. Later in this paper, we deduce the optimal group size for the previous two phase error identification method.

2) Reducing BIST Iterations: Fig. 6 shows a BIST structure with the error detectors proposed in [19]. While the tester observes the response of the first iteration of the BIST sequence, signature analyzers compact the responses which are to be observed by the tester in the second and the third BIST iterations. A counter is associated with each SA to select the responses to be observed by the signature analyzer. If a signature is not erroneous, we can skip the corresponding iteration. For example, if the signature analyzer SA1, detects no error and the signature analyzer SA2, detects an error, the tester skips the second iteration and observes the third iteration. Also, during the third iteration the signature analyzers compact the responses which are to be observed by the tester in the fourth and fifth iterations, and so on. The hardware overhead of the scheme consists of these signature analyzers and counters. If no signature analyzers were used and we wish to observe all responses with a slow tester, we could achieve this by repeating the BIST sequence $P = f_c/f_t$ times. The number of BIST iterations, P_{SA}, using signature analyzers can be estimated as follows. Assuming that we can skip $E(\mathrm{skip})$ iterations using signature analyzers, then

$$P_{\mathrm{SA}} = P - E(\mathrm{skip}). \tag{1}$$

In [19], it is shown that the expected value of $E(\mathrm{skip})$ can be computed as follows:

$$E(\mathrm{skip}) = \sum_{m=1}^{P} m \cdot \Pr\{m\,\mathrm{skip}\} = P \cdot \Pr\{1\,\mathrm{skip}\} \tag{2}$$

where $\Pr\{\mathrm{mskip}\}$ is the probability that m iterations are skipped. Further, it is also shown in [19] that $\Pr\{1\,\mathrm{skip}\}$ is nearly equal to the probability of no errors ($\Pr\{\mathrm{no\ error}\}$), provided more than three signature analyzers are used. A signature analyzer compacts w/P responses in one BIST iteration, where w is the size of a pattern group. Therefore, the probability of skipping one BIST iteration is

$$\Pr\{1\,\mathrm{skip}\} \approx \Pr\{\mathrm{no\ error}\} = (1 - \Pr\{1\,\mathrm{bit\ error}\})^{\frac{w}{P}}.$$

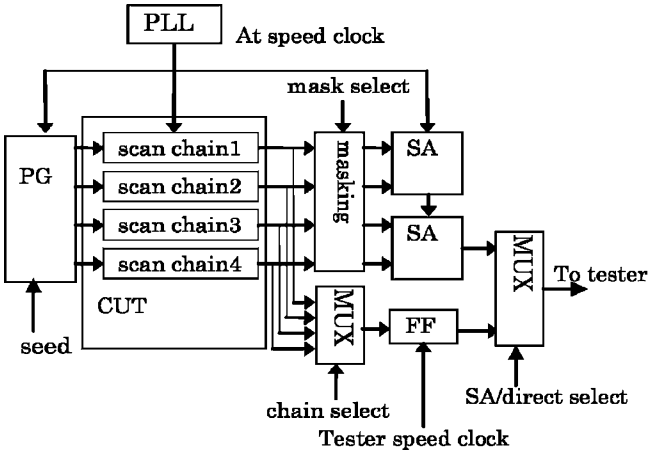

Fig. 7. BIST architecture for two-phase error identification.

The number of BIST iterations P_{SA}, using signature analyzer(s) is

$$\begin{aligned} P_{\mathrm{SA}} &= P(1 - \Pr\{\mathrm{no\ alias}\}\Pr\{1\,\mathrm{skip}\}) \\ &\approx \frac{f_c}{f_t}\left(1 - (1 - 2^{-S_{\mathrm{SA}}})(1 - \Pr\{1\,\mathrm{bit\ error}\})^{\frac{f_t w}{f_c}}\right) \end{aligned} \tag{3}$$

where S_{SA} is the size of each signature analyzer. If S_{SA} is large then the previous expression can be approximated as

$$P_{\mathrm{SA}} \approx \frac{f_c}{f_t}\left(1 - (1 - \Pr\{1\,\mathrm{bit\ error}\})^{\frac{f_t w}{f_c}}\right) \tag{4}$$

This equation will be used later for optimizing the group size.

C. BIST Architecture

Fig. 7 shows a BIST architecture with a single output and the logic required for diagnosis. The BIST architecture of Fig. 7 is based on the scan-based BIST which is one of the most commonly used architectures. The BIST pattern generator (PG) provides scan inputs and the signature analyzers (SAs) compact the responses. The CUT and the BIST circuits operate at-speed, the clocks are usually generated by a PLL circuit. Multiple input signature registers can be used as signature analyzers during testing. During diagnosis, a masking circuit allows only one scan chain to feed a signature analyzer which is selected by input "mask select."

As shown in Fig. 7, scanouts are connected to an output port via a multiplexer during diagnosis as in the noncompaction-based approach [19]. A Register FF is inserted at the multiplexer output to synchronize the scan chain with the tester since the CUT test frequency may be higher than the tester frequency. The FF samples the signal produced by the scan chain and holds the value during one tester period.

In the first phase of the identification procedure, each signature analyzer compacts responses of one scan chain which is selected by masking circuit. If there are n_{SA} signature analyzers, n_{SA} pattern groups in different scan chains can be simultaneously tested.

On the other hand, in the second phase, only one group in one scan chain, selected by the multiplexers, can be tested since we use noncompaction-based approach in the second phase.

D. Procedure of Error Identification

Based on Sections III-A–III-C, the procedure for identifying error occurrences is as follows.

Given condition.
Test frequency of CUT: f_c.
Tester frequency limitation: f_t.
Number of signature analyzers: n_{SA}.
BIST test length: N.
Number of pattern groups: g.
Number of scan out clock cycles including capture operation: L.
Target number of errors identified: E.

Step 1. identify erroneous pattern groups.
Step 1.1. Select untested pattern groups for each signature analyzer.
Step 1.2. Apply BIST sequence and identify signature analyzers containing erroneous signatures, i.e., erroneous pattern groups.
Step 1.3. Repeat step 1.1–1.2 until all pattern groups are tested. Total number of BIST iterations is: g/n_{SA}

Step 2-wsa. Identify erroneous scan cell and pattern (without using signature analyzer).
This step is followed if signature analyzers are not used during the error identification phase.
Step 2-wsa.1. Select an erroneous group which is identified in Step 1. The size of pattern group is $w = N/g$.
Step 2-wsa.2. Set observation time period $P = f_c/f_t$. Adjust the BIST test length to $w' = w + \alpha$ such that w' and P are co-prime. Note that α is small and can be found easily as shown in Appendix II. Reset the BIST iteration counter $r = 0$ and tester observation counter $i = 0$.
Step 2-wsa.3. Apply $P \cdot w'$ clocks to BIST pattern generator, while observing one scan output every P test cycles.
Step 2-wsa.4. If an error is detected at the $(i+1)$th observation, then:
Failing scan pattern = $\lfloor iP \bmod w'/L \rfloor$;
Erroneous scan cell = $(iP \bmod w') \bmod (L)$.
Step 2-wsa.5. Repeat Step 2-wsa.1 to Step 2-wsa.4 until at most E errors are identified.

Step 2-sa. Identify erroneous scan cell and pattern (with signature analyzer).
This step is followed if signature analyzers are used during the error identification phase.
Step 2-sa.1. Select an erroneous group which is identified in Step 1. The size of pattern group is $w = N/g$.
Step 2-sa.2. Set observation time period $P = f_c/f_t$. Adjust the BIST test length to $w' = w+\alpha$ such that w' and P are co-prime. Note that α is small and can be obtained as shown in Appendix II. Reset the BIST iteration counter $r = 0$ and tester observation counter $i = 0$.
Step 2-sa.3. Apply w' clocks to CUT, while observing a scan output every P test cycles by the tester. Note that each of the n_{SA} signature analyzers compacts the scan output which will be observed by the tester in $r+1, r+2, \ldots, r+n_{\mathrm{SA}}$ BIST iteration, respectively. The counter i is incremented by each tester observation.
Step 2-sa.4. Observe each signature analyzer by the tester. Set the next BIST iteration counter r_{next} as follows:
if no signature analyzer detects errors then, set $r_{\mathrm{next}} = r + n_{\mathrm{SA}} + 1$;
else, set $r_{\mathrm{next}} = j_{\min}$, where $j_{\min}$ is the minimum BIST iteration number which corresponds to the signature analyzer with erroneous result.
Step 2-sa.5. Adjust the BIST test length to $w' = w+\alpha+\beta$, where β is the number of iterations to be skipped

$$\beta = (r_{\mathrm{next}} - r - 1)(N + \alpha) \bmod P.$$

Set $r = r_{\mathrm{next}}$ and tester observation counter i to

$$i = i + \left\lfloor \frac{(r_{\mathrm{next}} - r - 1)(N + \alpha)}{P} \right\rfloor.$$

Step 2-sa.6. If an error is detected at the $(i + 1)$th observation, then:
Failing scan pattern = $\lfloor iP \bmod w'/L \rfloor$;
Erroneous scan cell = $(iP \bmod w') \bmod (L)$.
Step 2-sa.7. Repeat Step2-sa.3 to Step2-sa.5 while $r \leq P$.
Step 2-sa.8. Repeat Step2-sa.1 to Step2-sa.7 until at most E errors are identified.

As shown in Appendix I. The maximum resolution is achieved if and only if $\gcd(N, P) = 1$ and the number of BIST iterations is at least P. To satisfy the condition $\gcd(N, P) = 1$, it is sufficient to adjust only N by inserting dummy clocks for the minimum test application time. However, it has not been shown yet how to decide the size of group w, i.e., the number of groups g. Clearly the group size should be chosen to minimize test time, the problem formulation and solution related to this aspect is given in Section IV.

IV. Optimizing Group Size

A. Tester Loading Rate

Test data is prepared for each faulty chip in the fault diagnosis phase. Therefore, tester loading time cannot be ignored especially in view of the fact that the tester may not have sufficient memory for all test data. In this case, the tester should load the test data several times.

The tester loading time will normally be proportional to the test data volume and the overhead associated with each tester loading. Thus the tester loading time T_{Load} can be expressed as follows using two constants L_C and L_V:

$$T_{\mathrm{Load}} = L_C \cdot (\#\text{of tester loads}) + L_V \cdot V \tag{5}$$

where V is the test volume. Let M be the tester memory size, then the number of tester loading is V/M. Therefore

$$T_{\mathrm{Load}} = V \left(\frac{L_C}{M} + L_V \right). \tag{6}$$

Note that L_C, L_V, and M are parameters associated only with the tester. Denote $r_{\mathrm{Load}} = M/(L_C + ML_V)$, the tester loading time is

$$T_{\mathrm{Load}} = \frac{V}{r_{\mathrm{Load}}}. \tag{7}$$

In this paper, we use r_{Load} as the bit rate of tester loading. Note that the parameter r_{Load} reflects tester memory limitation also.

B. Analysis of Test Time

In this subsection, we estimate the test time for the error identification procedure which is proposed in Section III-D.

We use following notation in the analysis that follows:

N	total length of the BIST sequence;
w	size of the expected response of one group;
g	number of groups of all scan chains ($g = N/w$);
S_{SA}	bit size of signature;
S_{PG}	bit size of pattern generator;
n_{SA}	number of signature analyzers;
r_{Load}	bit rate for loading to the tester;
f_c	CUT test frequency;
f_t	tester frequency;
g_e	number of groups identified erroneous in Step 1;
P_{SA}	the average number of BIST iterations using signature analyzer in Step 2;
E	target number of errors to be identified.

1) Analysis of Step 1: The error identification procedure Step 1 identifies erroneous pattern groups.

The total test time including tester loading time is Total test time = test application time + tester loading time. The test application time of Step 1 can be estimated as follows. In Step 1 we test n_{SA} groups simultaneously using n_{SA} signature analyzers. Therefore, the BIST pattern should be applied in g/n_{SA} times. At the end of the BIST session, we read out the signature. We assume that we read out signatures on one output port as shown in Fig. 7.

Therefore, the test application time of Step 1 is

$$\mathrm{TAT}_{\mathrm{step1}} = g\left(\frac{w}{n_{\mathrm{SA}} f_c} + \frac{S_{\mathrm{SA}}}{f_t}\right) + \frac{S_{\mathrm{PG}}}{f_t}. \tag{8}$$

The total test volume for Step 1 is

$$V_{\mathrm{Step1}} = gS_{\mathrm{SA}} + S_{\mathrm{PG}}. \tag{9}$$

Therefore, total test time including loading time is

$$T_{\mathrm{step1}} = \mathrm{TAT}_{\mathrm{Step1}} + V_{\mathrm{Step1}}/r_{\mathrm{Load}}. \tag{10}$$

Example 1: Let the total length of a BIST sequence be 500-M clocks which is divided into 5-M groups. Let the size of signature analyzer and pattern generator be 64 bits. Let the CUT test frequency be 800 MHz, the tester frequency be 40 MHz, and tester loading rate be 150 Mbit/s. The test time for Step 1 is computed as follows:

We have the following parameters:

$$\begin{aligned} N &= 5 \times 10^8 \\ g &= 5 \times 10^6 \\ w &= N/g = 10^2 \\ S_{\mathrm{SA}} &= S_{\mathrm{PG}} = 64 \\ r_{\mathrm{Load}} &= 1.5 \times 10^8 (\mathrm{bps}) \\ f_c &= 8 \times 10^8\,\mathrm{Hz} \\ f_t &= 4 \times 10^7\,\mathrm{Hz} \end{aligned}$$

the test application time of Step 1 is

$$\begin{aligned} \mathrm{TAT}_{\mathrm{step1}} &= 5 \times 10^6 \left(\frac{10^2}{64 \cdot 8 \times 10^8} + \frac{64}{4 \times 10^7}\right) + \frac{64}{4 \times 10^7} \\ &= 16.1(s) \end{aligned}$$

Total test time including loading time is

$$T_{\mathrm{step1}} = 16.1 + \frac{5 \times 10^6 \cdot 64 + 64}{1.5 \times 10^8} = 20.4(s).$$

2) Analysis of Step 2-wsa (Without Signature Analyzer): We identify erroneous responses in the error identification procedure Step 2. The error identification procedure will be finished when we identify predetermined E errors. And since only erroneous groups are tested in Step 2, the expected number of erroneous bits in one group is

$$\begin{aligned} E_w &= \frac{w \Pr\{1\,\text{bit error}\}}{\Pr\{1\,\text{group error}\}} \\ &= \frac{w \Pr\{1\,\text{bit error}\}}{1 - (1 - \Pr\{1\,\text{bit error}\})^w}. \end{aligned} \tag{11}$$

Therefore, the expected number of groups contains E errors is

$$g'_e = \frac{E}{E_w} = \frac{E(1 - (1 - \Pr\{1\,\text{bit error}\})^w)}{w \Pr\{1\,\text{bit error}\}}. \tag{12}$$

If we do not use signature analyzers in Step 2, we must apply BIST sequence f_c/f_t times for erroneous groups. Therefore, the test application time of Step 2 without signature analyzer is

$$\mathrm{TAT}_{\mathrm{step2}} = \frac{f_c g'_e}{f_t}\left(\frac{w}{f_c} + \frac{S_{\mathrm{PG}}}{f_t}\right) = \frac{g'_e}{f_t}\left(w + \frac{f_c S_{\mathrm{PG}}}{f_t}\right). \tag{13}$$

The test volume is

$$V_{\mathrm{Step2}} = g'_e(w + S_{\mathrm{PG}}). \tag{14}$$

Total test time including loading time is

$$T_{\mathrm{step2}} = \mathrm{TAT}_{\mathrm{Step2}} + V_{\mathrm{Step2}}/r_{\mathrm{Load}}. \tag{15}$$

Example 2: Let all the parameters be the same as in Example 1. Let the probability of 1 bit error occurrence be 0.1% and we need to identify 1000 errors. In this case the test time for Step 2 is computed as follows.

We use (13) and (15) to obtain

$$\begin{aligned} \mathrm{TAT}_{\mathrm{step2}} &= 0.0662(s) \\ T_{\mathrm{step2}} &= 0.0677(s). \end{aligned}$$

3) Analysis of Step 2-sa (With Signature Analyzer): When we use signature analyzers in Step 2 as introduced in Section III-B, we have to read out signature at the end of each BIST session. Let P_{SA} be the average number of BIST iterations using signature analyzer which is deduced in Section III-D, the test application time without reading signature analyzer and setting pattern generator is

$$\mathrm{TAT}_{\mathrm{step2}} = g'_e P_{\mathrm{SA}}\left(\frac{w}{f_c} + \frac{S_{\mathrm{PG}} + n_{\mathrm{SA}} S_{\mathrm{SA}}}{f_t}\right). \tag{16}$$

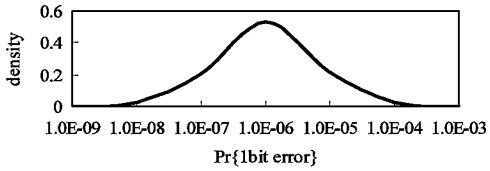

Fig. 8. Distribution of error probability.

Since we do not know which BIST iteration can be skipped beforehand, we have to load all BIST sequences to the tester memory. Therefore, the test volume is

$$V_{\mathrm{Step2}} = g'_e \left(w + S_{\mathrm{PG}} + \frac{S_{\mathrm{SA}} f_c}{f_t} \right). \tag{17}$$

Again, The total test time including loading time is

$$T_{\mathrm{step2}} = \mathrm{TAT}_{\mathrm{Step2}} + V_{\mathrm{Step2}} / r_{\mathrm{Load}}. \tag{18}$$

Example 3: Consider again the case described in Example 2 with using five signature analyzers. By using (16) and (18), we get

$$\mathrm{TAT}_{\mathrm{step2}} = 1.92 \times 10^{-4}(s)$$
$$T_{\mathrm{step2}} = 0.0187(s).$$

4) Optimizing the Group Size: In this section, we analyze the total test time including tester loading time for Step 1 and Step 2 with and without signature analyzers. We can find the optimal grouping size $w\,(1 \le w \le N)$, which minimizes

$$T_{\mathrm{Step1}} + T_{\mathrm{Step2}} \tag{19}$$

by using common solver tools. However, there is an unknown parameter $\Pr\{1\mathrm{bit\ error}\}$ in (19) which depends on the existence of fault(s) in the chip and its manifestation as an error. In Section V, we will do some experiments to show the relationship between error probability and optimal group size w, and propose a practical group size w.

V. Experimental Results

A. Optimal Group Size

1) Distribution of Error Probability: We have shown that the test time of error identification depends on the error probability. In Fig. 8, we show the distribution of the error probability for a large industrial circuit which is obtained by simulating randomly selected 100 single stuck-at faults. The experimental circuit is a part of an SoC developed at NEC Electronics Co. and details of the circuit are provided in Table I. The BIST architecture shown in Fig. 7 is used in this experimental circuit. The PG and SA are general LFSR and MISR, respectively. In the chip testing phase, each scan operation needs a capture cycle, therefore, the BIST test length is $(54505+1) \times 1000$. Note that in the diagnosis phase, the total BIST test length will be much longer than the testing phase, since the BIST operation is adjusted and repeated several times as described in Section III-D. This total test time will be analyzed by experiments in this section. An 800-MHz scan clock generated by PLL and provided to each scan FF is used. However, tester can observe scan out at 40 MHz by a tester synchronization circuit as described in Section III-C.

TABLE I
Experimental Circuit

No. of gates	6M gates
No. of FFs	54505
No. of external ports	317
No. of Scan chains	65
Size of PG and SA	64 bit LFSR
No. of SAs	5
clock frequency of CUT	800MHz
clock frequency of tester	40MHz
bit rate of tester loading	140Mbps
Length of Test pattern (Length of the BIST sequence)	1000 (54,506,000)

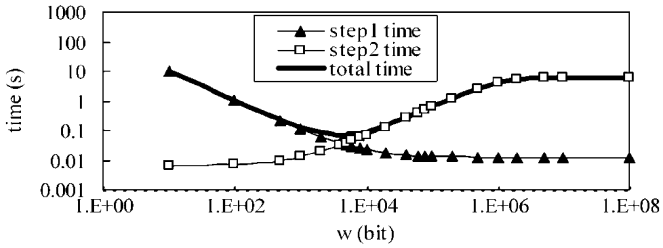

Fig. 9. Test time as a function of group size (at typical error probability).

As shown in Fig. 8, the error probability is almost in the range of $10^{-8} < \Pr\{1\,\mathrm{bit\ error}\} < 10^{-4}$, and typically near 10^{-6}. In this section, we first investigate the typical case, $\Pr\{1\,\mathrm{bit\ error}\} = 10^{-6}$, and then we investigate both lower and higher error probability cases.

We first conduct an experiment without signature analyzers in Step 2 to figure out the relationship between group size and the error probability. Later, in this section, we will show the effect of using signature analyzers in Step 2.

2) Typical Error Probability Case: We plot the total test time including tester loading time using equations deduced in Section IV. The parameters reflect the experimental circuit shown in Table I and we assume error probability is typical $(\Pr\{1\,\mathrm{bit\ error}\} = 10^{-6})$. Fig. 9 shows the Step 1, Step 2 (without signature analyzer) and total test time as a function of group size w, to identify 200 errors which we believe is sufficient error information for diagnosing scan based design [3]–[5].

Fig. 9 shows that the test time of Step 1 is significantly increased when group size becomes very small. It is because very small groups need too much expected signature data to be loaded onto the tester which causes many tester reloadings, resulting in more time being spent in observing signatures. The test time of Step 1 becomes constant when we use sufficiently large group size ($w > 5000$ bits).

Fig. 9 also shows that smaller group size is efficient to minimize the test time of Step 2. It is because we perform Step 2 only for groups identified as erroneous in Step 1, and more error-free responses can be pruned when the size of group is smaller.

Therefore, there is an optimal point of group size to minimize total test time. In this experimental case, 5860 bits is the optimal group size.

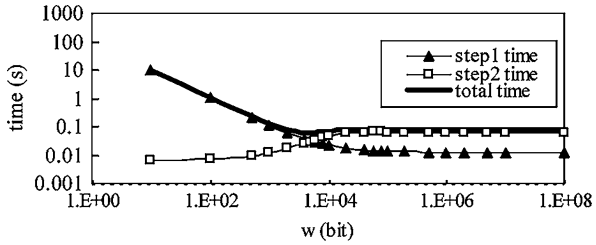

Fig. 10. Test time as a function of group size (at high error probability).

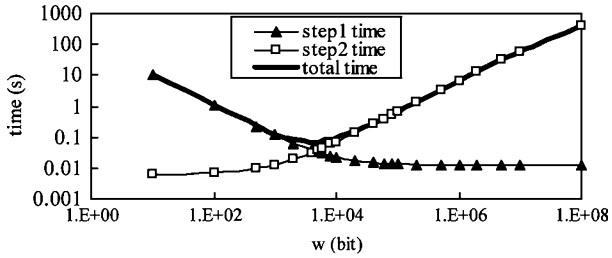

Fig. 11. Test time as a function of group size (at low error probability).

3) High Error Probability Case: In the next experiment, we assume the error probability is very high ($\Pr\{1\,\text{bit error}\} = 10^{-4}$). Fig. 10 shows the Step 1, Step 2 (without signature analyzer) and total test time as a function of group size w, to identify 200 errors.

Fig. 10 shows that the total test time is almost dominated by Step 1 when the error probability is very high. The test time of Step 2 is much shorter than the typical case since the Step 2 procedure is terminated by finding $E = 200$ errors.

Therefore, the optimal grouping strategy for high error probability case may be as large as possible, i.e., no-grouping is needed. However, Fig. 10 also shows that when the group size is larger than 5000, which is around optimal solution for typical case, test time is almost constant. Therefore, the solution for the optimal group size under typical error probability is also practically suitable for higher error probability case.

4) Low Error Probability Case: Next, we plot the test time for the case when the error probability is very low ($\Pr\{1\,\text{bit error}\} = 10^{-8}$). Fig. 11 shows the Step 1, Step 2 (without signature analyzer) and total test time as a function of group size w, to identify 200 errors. Figs. 9 and 11 show that the test time with optimal size group is almost identical for both typical and low error probability cases, however, the test time grows significantly longer in lower error probability case when we use larger group size. Therefore, optimizing group size is much more important for low error probability case.

5) Optimized Group Size: The next question is the relationship between the optimized group size and the error probability. Fig. 12 shows the optimal group size as a function of the error probability.

Fig. 12 shows that the optimal group size is almost constant when the error probability is lower than 10^{-5}. The optimal group size much larger for higher error probability case, however, we have already shown in Section V-A3 that test time is not so influenced.

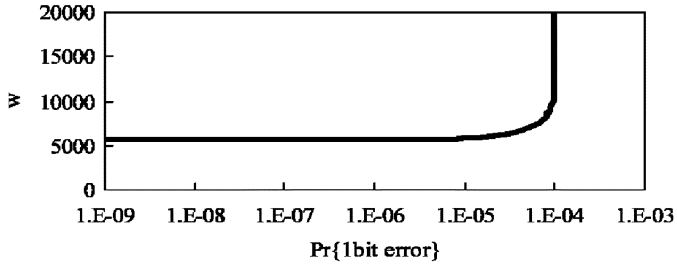

Fig. 12. Optimized group size as a function of the error probability.

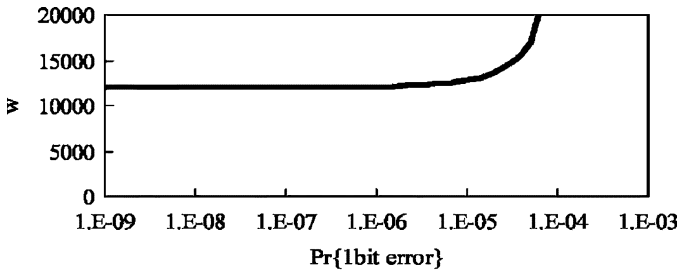

Fig. 13. Optimized group size as a function of the error probability (using signature analyzer in Step 2).

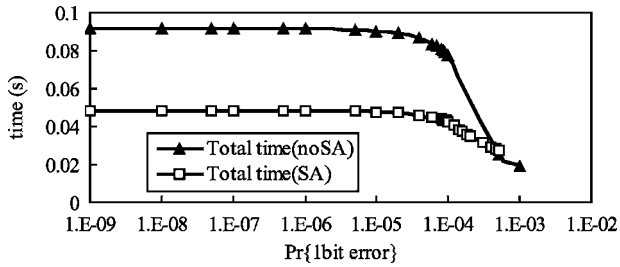

Fig. 14. Test time with and without signature analyzers.

Therefore, we conclude that the optimal size which minimizes test time can be obtained by solving (19) under the assumption of $\Pr\{1\,\text{bit error}\}$ to be fairly low.

B. Effect of Using Signature Analyzers in Step 2-sa

In Section V-A, we discussed the optimal group size for the case that we do not use signature analyzers in Step 2 identification procedure. As shown in Section III-B we can use signature analyzers in Step 2 to reduce the BIST iterations.

Fig. 13 shows the optimal group size as a function of $\Pr\{1\,\text{bit error}\}$. As shown in Figs. 12 and 13, the optimal group size is larger than the case signature analyzers are not used, however, the relation between the error probability and the optimal size group is quite similar. Therefore, the optimal group size of the case using signature analyzers in Step 2 also can be obtained by solving (19) assuming $\Pr\{1\,\text{bit error}\}$ is sufficiently low. Fig. 14 compares the test time with and without signature analyzers. As shown in Fig. 14, by using signature analyzers in Step 2, test time will reduce $1/2$ for practical range of error probability ($10^{-8} < \Pr\{1\,\text{bit error}\} < 10^{-4}$). When we use signature analyzers for error identification, it is also important to determine the proper signature size to minimize test time as argued in the following. By using smaller size of signature, test time can be reduced. However, aliasing

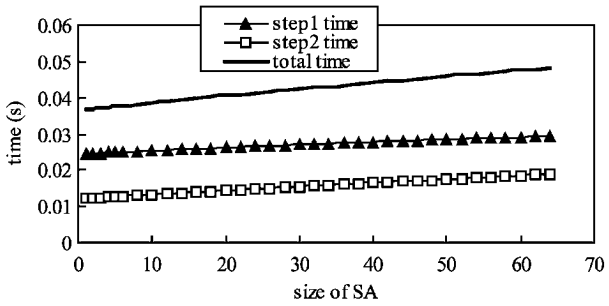

Fig. 15. Test time as a function of size of signature analyzers.

probability will be increased. When a signature indicates no errors for erroneous responses, we have to try other groups to identify more errors and it may increase test time. Fig. 15 shows that test time as a function of size of signature analyzers with optimized sized groups and typical error probability ($\Pr\{1\,\text{bit error}\} = 10^{-6}$). In this experiment, the aliasing of signature analyzers is also taken into account by using (4). As shown in Fig. 15, aliasing does not impact the test time even if the signature size is very small. However, using different signature size will cost more hardware overhead and Fig. 15 shows the test time for large signature and small signature is actually not so much different. Therefore, we conclude that the size of signature is not important for error identification procedure. The signature size when used in production testing is also suitable for diagnosis.

VI. Conclusion

In this paper, we proposed a method for identifying erroneous responses for the BIST architecture in minimum test time. Our approach is efficient even if the CUT test clock frequency is much higher than the tester frequency. Our approach requires only multiplexer and masking circuit for diagnosis which is quite negligible hardware overhead.

We also proposed how to decide the size of pattern group to minimize test time. Our method takes into consideration the tester loading time. The proposed equations that are used to decide the optimal group size include the error probability. However, the experimental results show that the group size which is obtained under assumption of low error probability is also optimal for higher error probability cases. Experimental results also show that our method reduces the test time by a factor of 10 compared to the methods that do not employ the signatures analyzers during diagnosis. Therefore, we conclude that our method can identify sufficiently many errors for diagnosis with minimum test application time with very little hardware overhead.

Appendix I
Conditions to Achieve Maximum Resolution

We use the following terms.

Absolute Time: The number of a scan clock cycle starting from the beginning of the first BIST iteration.

Relative Time: The number of a scan clock cycle starting from the beginning of the current BIST iteration.

We use the following notation:

- N length of the BIST test sequence;
- P period of the tester relative to the CUT test clock period. We assume that P is an integer and that $1 < P < N$;
- $R_{\min}$ minimum number of BIST iterations to observe every response;
- $M(i)$ relative time at $(i+1)$th observation.

The range of $M(i)$ is $0 \le M(i) < N$ and $M(i) = iP - kN$ assuming the response at $M(i)$ is observed during the $(k+1)$th BIST iteration.

We derive the relationship between P and N to achieve the maximum resolution of observation.

Theorem 1: The maximum resolution is achieved if and only if $\gcd(N, P) = 1$ and the number of BIST iterations is at least P.

Proof: Let us assume that the maximum resolution is achieved. Then the response at the relative time 1 is observable and, therefore, $M(i) = iP - kN = 1$ has a solution (i, k), which shows that $\gcd(N, P) = 1$. Furthermore, $\lfloor N \cdot R/P \rfloor$ bits are observed in R BIST sequences. Therefore, the maximum resolution implies $\lfloor N \cdot R/P \rfloor \ge N$ or $R \ge P$.

Let us assume that the number of BIST iterations is P and $\gcd(N, P) = 1$, then the equation $M(i) = t$ cannot have more than one solution, otherwise $M(i)$ can be expressed in two ways using two solutions i_1, i_2

$$M(i) = i_1P - k_1N = i_2P - k_2N$$
$$(i_1 - i_2)P = (k_1 - k_2)N.$$

This shows that P divides $k_1 - k_2$ and since $0 \le |k_1 - k_2| < P$, it implies $k_1 = k_2$ and $i_1 = i_2$.

The number of observations in P BIST iterations is $\lfloor N \cdot P/P \rfloor$, i.e., N. Since $M(i)$ for every $0 \le i < N$ are different, the set $\{M(i) : 0 \le i < N\}$ has to be $\{0, 1, 2, \ldots, N-1\}$, i.e., the observing resolution is maximum. ■

Appendix II
Adjusting N or P to Achieve Maximum Resolution of Observation and Minimum TAT

The adjustment of N and/or P is chosen to minimize the test application time. Let $N' = N + i$ be the adjusted length of the BIST sequence and $P' = P + j$ be the adjusted tester observing period. The test application time is

$$\text{TAT} = \frac{N'P'}{f_c} = \frac{1}{f_c}(NP + iP + jN + ij). \qquad (20)$$

The problem is to find a pair (i, j) that minimizes $iP + jN + ij$, with $N + i$ and $P + j$ co-prime.

Theorem 2: If $N \ge P(P-1)$, The solution (i, j) that minimizes (20) with $N + i$ and $P + j$ co-prime is such that $j = 0$.

Proof: Since $\gcd(\alpha P + 1, P) = 1$ for any integer $\alpha \ge 0$, there exists a co-prime of P in any consecutive P integers.

Therefore, the range of i in (20) is $0 \leq i < P$. Similarly, the range of j is $0 \leq j < N$. First, we consider the case where $j = 0$. The worst case of minimum $iP + jN + ij$ is the case where $i = P - 1$, therefore

$$iP + jN + ij = (P - 1)P.$$

Next, we consider the case when $j \neq 0$. The best case of minimum $iP + jN + ij$ is the case where $i = 0$, therefore

$$iP + jN + ij = jN \geq N.$$

Therefore, if $N > P(P - 1)$, $j = 0$ is the solution that minimizes test application time. ■

REFERENCES

[1] P. H. Bardell, W. H. McAnney, and J. Savir, *Built-in Test for VLSI: Pseudorandom Techniques*. New York: Wiley Interscience, 1987.

[2] H. Y. Chen, E. Manning, and G. Mets, *Fault Diagnosis of Digital Systems*. New York: Wiley, 1970.

[3] J. A. Waicukauski and E. Lindbloom, "Failure diagnosis of structured VLSI," *IEEE Design Test*, vol. 6, no. 4, pp. 49–60, Aug. 1989.

[4] M. Abramovici and M. A. Breuer, "Multiple fault diagnosis in combinational circuits based on effect-cause analysis," *IEEE Trans. Comput.*, vol. C-29, no. 6, pp. 451–460, Jun. 1980.

[5] K. Shigeta and T. Ishiyama, "An improved fault diagnosis algorithm based on path tracing with dynamic circuit extraction," in *Proc. Int. Test Conf.*, 2000, pp. 235–244.

[6] J. Rajski and J. Tyszer, "Fault diagnosis in scan-based BIST," in *Proc. Int. Test Conf.*, 1997, pp. 894–902.

[7] I. Bayraktaroglu and A. Orailoglu, "Improved fault diagnosis in scan-based BIST via superposition," in *Proc. Des. Autom. Conf.*, 2000, pp. 55–58.

[8] I. Bayraktaroglu and A. Orailoglu, "Deterministic partitioning techniques for fault diagnosis in scan-based BIST," in *Proc. Int. Test Conf.*, 2000, pp. 273–282.

[9] J. Ghosh-Dastidar and N. A. Touba, "A rapid and scalable diagnosis scheme for BIST environments with a large number of scan chains," in *Proc. VLSI Test Symp.*, 2000, pp. 73–78.

[10] C. Liu and K. Chakrabarty, "A partition-based approach for identifying failing scan cells in scan-BIST with application to system-on-chip fault diagnosis," in *Proc. Des. Autom. Test Eur. Conf. Exhibition*, 2003, pp. 230–235.

[11] J. G. Dastidar, D. Das, and A. Touba, "Fault diagnosis in scan-based BIST using both time and space information," in *Proc. Int. Test Conf.*, 1999, pp. 95–102.

[12] J. Savir and W. H. McAnney, "Identification of failing tests with cycling registers," in *Proc. Int. Test Conf.*, 1988, pp. 322–328.

[13] T. R. Damarla, C. E. Stroud, and A. Sathaye, "Multiple error detection and identification via signature analysis," *J. Electron. Testing: Theory Appl.*, vol. 7, pp. 193–207, 1995.

[14] Y. Wu and S. Adham, "BIST fault diagnosis in scan-based VLSI environments," in *Proc. Int. Test Conf.*, 1996, pp. 48–57.

[15] J. Savir, "Salvaging test windows in BIST diagnostics," in *Proc. VLSI Test Symp.*, 1997, pp. 416–425.

[16] T. Clouqueur, O. Ercevik, K. K. Saluja, and H. Takahashi, "Efficient signature-based fault diagnosis using variable size windows," in *Proc. VLSI Des.*, 2001, pp. 391–396.

[17] C. Liu and K. Chakrabarty, "Failing vector identification based on overlapping intervals of test vectors in a scan-BIST environment," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 22, pp. 593–604, 2003.

[18] P. Wohl, J. A. Waicukauski, S. Patel, and G. Maston, "Effective diagnostics through interval unloads in a BIST environment," in *Proc. 39th Des. Autom. Conf.*, 2002, pp. 10–14.

[19] Y. Nakamura, T. Clouqueur, K. K. Saluja, and H. Fujiwara, "Error identification in at-speed scan BIST environment in the presence of circuit and tester speed mismatch," *IEICE Trans. Inf. Syst.*, vol. E-89-D, no. 3, pp. 1165–1172, 2006.

[20] The Semiconductor Industry Association (SIA), San Jose, CA, "International technology roadmap of semiconductors," 2005.

[21] Y. Nakamura, T. Clouqueur, K. K. Saluja, and H. Fujiwara, "Diagnosing at-speed scan BIST circuits using a low speed and low memory tester," in *Proc. 15th IEEE Asian Test Symp. (ATS'06)*, 2006, pp. 409–414.

Yoshiyuki Nakamura (S'04–M'06) received the B.E. and M.E. degrees in electronics and communication engineering from Meiji University, Kawasaki, Japan, in 1988 and 1990, and the Ph.D. degree in information science from Nara Institute of Science and Technology, Nara, Japan, in 2006.

He is currently a Senior Engineer at NEC Electronics Corporation, Kawasaki, Japan. In 1990, he joined NEC Corporation, Kawasaki, Japan, and has been engaged in the development of electronic design automation (EDA) for testing. His research interests include design for test, including SCAN, built-in self-test, and SOC testing.

Dr. Nakamura is a member of the Institute of Electronics, Information and Communication Engineers of Japan (IEICE) and a member of the Information Processing Society of Japan (IPSJ).

Thomas Clouqueur (S'01–M'04) received the Engineering degree in electrical engineering from the Ecole Superieure d'Electricite, Gif-sur-Yvette, France, in 1999, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Wisconsin, Madison, in 1999 and 2003, respectively.

He is currently with Advanced Micro Devices Inc., Boxborough, MA. He was a COE Postdoctoral Fellow at the Nara Institute of Science and Technology, Nara, Japan, in 2004–2005. His research interests include VLSI design and testing and fault-tolerant computing.

Kewal K. Saluja (S'70–M'73–SM'89–F'95) received the Bachelor of Engineering (BE) degree in electrical engineering from the University of Roorkee, Roorkee, India, in 1967, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Iowa, Iowa City, in 1972 and 1973, respectively.

He is currently a Professor with the Department of Electrical and Computer Engineering, University of Wisconsin-Madison, where he teaches courses in logic design, computer architecture, microprocessor-based systems, VLSI design and testing, and fault-tolerant computing. Prior to this, he was at the University of Newcastle, Newcastle, Australia. He has held visiting and consulting positions at various national and international institutions including University of Southern California, Hiroshima University, Nara Institute of Science and Technology, and the University of Roorkee. He has also served as a consultant to the United Nations Development Program. He was the general chair of the 29th FTCS. He is currently the Associate Editor for the letters section of the *Journal of Electronic Testing: Theory and Applications (JETTA)*. He has authored or co-authored over 250 technical papers that have appeared in conference proceedings and journals.

Prof. Saluja served as an Editor of the IEEE TRANSACTIONS ON COMPUTERS (1997–2001). He is a member of Eta Kappa Nu, Tau Beta Pi, and a fellow of the JSPS.

Hideo Fujiwara (M'74–SM'83–F'89) received the B.E., M.E., and Ph.D. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1969, 1971, and 1974, respectively.

Presently, he is a Professor at the Graduate School of Information Science, Nara Institute of Science and Technology, Nara, Japan. He was with Osaka University from 1974 to 1985 and Meiji University from 1985 to 1993, and joined Nara Institute of Science and Technology in 1993. In 1981, he was a Visiting Research Assistant Professor at the University of Waterloo, Waterloo, ON, Canada, and, in 1984, he was a Visiting Associate Professor at McGill University, Montreal, QC, Canada. His research interests include logic design, digital systems design and test, VLSI CAD, and fault tolerant computing, including high-level/logic synthesis for testability, test synthesis, design for testability, built-in self-test, test pattern generation, parallel processing, and computational complexity. He is the author of *Logic Testing and Design for Testability* (MIT Press, 1985). He served as an Associate Editor of the *Journal of Electronic Testing: Theory and Application*, an Associate Editor of the *Journal of Circuits, Systems and Computers*, an Editor of *VLSI Design: Application Journal of Custom-Chip Design, Simulation, and Testing*, and several guest editors of special issues of *IEICE Transactions*. He also served as an Advisory Member of IEICE Editorial Board and is currently an Advisory Member of the IEICE *Transactions on Information and Systems*.

Dr. Fujiwara was a recipient of an IECE Young Engineer Award in 1977, an IEEE Computer Society Certificate of Appreciation Award in 1991, 2000, and 2001, an Okawa Prize for Publication in 1994, an IEEE Computer Society Meritorious Service Award in 1996, and an IEEE Computer Society Outstanding Contribution Award in 2001. He served as an Editor of the IEEE TRANSACTIONS ON COMPUTERS. He is a Golden Core member of the IEEE Computer Society, a fellow of the Institute of Electronics, Information and Communication Engineers of Japan (IEICE), and a fellow of the Information Processing Society of Japan (IPSJ).