Parallel Signature Analysis Design with Bounds on Aliasing

Nirmal R. Saxena, *Member*, *IEEE*, and Edward J. McCluskey, *Fellow*, *IEEE*

Abstract—This paper presents parallel signature design techniques that guarantee the aliasing probability to be less than $2/L$, where L is the test length. Using y signature samples, a parallel signature analysis design is proposed that guarantees the aliasing probability to be less than $(y/L)^{y/2}$. Inaccuracies and incompleteness in previously published bounds on the aliasing probability are discussed. Simple bounds on the aliasing probability are derived for parallel signature designs using primitive polynomials.

Index Terms—Signature analysis, aliasing probability bounds, random testing, linear feedback shift registers, parallel signature designs, multiple input signature registers (MISR).

1 Introduction

The subject area of this paper is *signature analysis* design. There are two main contributions in this paper. One, a simple relationship between the aliasing probability, the number of signature register samples, and the length of test stimulus is derived. This simple relationship provides a guideline to select the parallel signature design parameters. Two, inaccuracies in prior work on signature analysis are reported for the first time. These inaccuracies have somehow escaped the scrutiny, for almost a decade, of an otherwise vigilant research community in this subject area.

The organization of this paper is as follows:

- Section 1 introduces the subject area of this paper.
- Section 2 presents definitions and notational framework for this paper. Previous research on signature analysis design is also critically reviewed in this section.
- Section 3 presents the motivation behind the derivation of simple bounds for the aliasing probability in parallel signature analysis.
- Section 4 is a presentation of an elaborate mathematical derivation of simple bounds for the aliasing probability in parallel signature analysis.
- Section 5 is a summary and conclusion of this work.

1.1 Signature Analysis and Aliasing

The problem of testing a system is twofold: test application and response verification. Test application involves the careful selection and generation of test stimuli to activate defects in the system under test. Response verification involves the comparison of the response with the expected response from a defect-free system using the same stimuli. It is generally assumed that the test stimulus will be applied by a tester that can store test patterns and the corresponding defect-free responses. Such testers are expensive. Tester cost is not the only difficulty encountered in testing. The number of response patterns is also becoming too large to be handled efficiently by the tester hardware. Due to the limited visibility and accessibility of *very large scale integrated* (VLSI) chips, test application and response analysis are difficult problems. *Built-in self test* (BIST) is an approach that addresses the testing problem in VLSI chips. In a BIST structure, a test generation circuit and a response compaction circuit can be fabricated on the same VLSI chip as the circuit under test. In most BIST applications, the test stimulus is derived from pseudorandom test pattern generators. A data compaction circuit using a *linear feedback shift register* (LFSR) structure is illustrated in Fig. 1. This method of compaction [22], [23] is called *signature analysis.* The term *parallel signature analysis* is used for LFSR structures that compact the response from multiple circuit outputs [2], [3]; and, the term *serial signature analysis* is used for LFSR structures that compact response from a single circuit output.

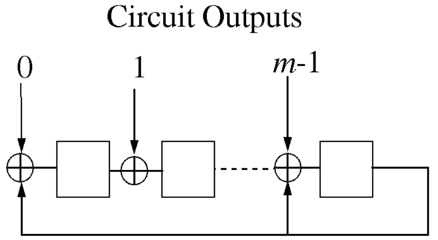

Fig. 1. Parallel signature analysis.

The final state of the linear feedback shift register, after the response has been compacted, is called the *signature* of the response. The signature of the response from a circuit under test is compared against a reference signature of the response from a defect-free circuit. The compaction process reduces a large amount (for example, a million bytes) of

- *N. Saxena is with Silicon Graphics, Inc., E-mail: nsaxena@abel.mti.sgi.com.*
- *E.J. McCluskey is with the Center for Reliable Computing, Gates 236, Stanford University, Stanford, CA 94305.*

Manuscript received 14 Sept. 1995; revised 25 Oct. 1996.
For information on obtaining reprints of this article, please send e-mail to: transcom@computer.org, and reference IEEECS Log Number C96332.

response data to a small amount of signature data. It is possible that the signature of some faulty response data to be the same as that of the defect-free response data. This phenomenon is called *aliasing.* Aliasing causes test escapes and lowers the quality of shipped parts. A desirable design goal, therefore, is to eliminate the incidence of aliasing.

Aliasing depends on the response data, the compaction function, and the error characteristics of the system under test. Error characteristics are usually described by probability models, and the phenomenon of aliasing is often quantified by the aliasing probability. The *aliasing probability* is the probability that the data compaction method produces the defect-free signature when the response data is in error. Probability models are used because there is some uncertainty about the error characteristics. For the aliasing probability measure to be relevant, it is important to have justifiable and validated probability models. The aliasing probability in parallel signature analysis is studied in the following context:

1) Test stimulus is derived from random pattern generators. Although in reality only pseudorandom pattern generators are used; their characteristics can be approximately modeled by the characteristics of a random pattern generator [11].
2) The circuit under test is combinational.
3) Faults in the circuit under test are combinational. Faults that preserve the combinational nature of a circuit are called *combinational faults.*

The results for serial signature analysis in [14], [15], [16] give the following simple relationship between random pattern test length L, the signature register period, and the aliasing probability:

- The aliasing probability is less than $1/L$ for random pattern test length L and signature register period greater than L.

For example, if the system designer needs to guarantee aliasing probability in serial signature analysis to be less than one in a 1,000,000, then a random pattern test of length 1,000,000 and a primitive signature register size of 20 would suffice.

This paper addresses design issues related to parallel signature analysis. The results in this paper present a simple relationship between the number of outputs m in the circuit under test, the random pattern test length L, the primitive signature register size r, the number of signature register samples y, and the aliasing probability. It is shown that the aliasing probability is less than $(y/L)^{y/2}$ if $r \geq m$ and $2^r - 1 > L/y$. For example, if $m = 32$ then using two signatures samples ($y = 2$), a test length $L = 2{,}000{,}000$, a primitive signature register of size $r = 32$ will guarantee the aliasing probability to be less than one in a 1,000,000. If only one signature sample ($y = 1$) is used then the aliasing probability is guaranteed to be less than one in 1,414. Readers can note that for $y = 1$, the results in this paper show that the aliasing probability is less than $1/\sqrt{L}$ for a random pattern test length L and a primitive signature register period greater than L. This is not as tight as the $1/L$ bound for serial-signature analysis. Researchers are encouraged to improve this bound. Clearly, serial-signature bounds are not applicable to parallel signature analysis. Fig. 2 demonstrates this by comparing the experimental estimates of aliasing probability for parallel signature analysis with the $1/L$ bound for serial-signature analysis. Fig. 2 shows that in the test length interval 200-250 the experimental aliasing probability estimates far exceed the $1/L$ bound. The details of this experiment are described in Section 2.1.

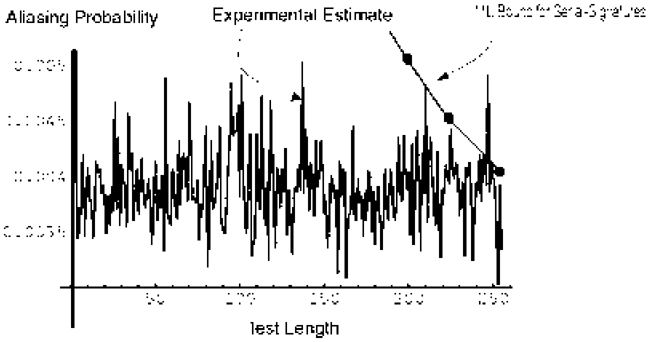

Fig. 2. Serial-signature aliasing bound vs. aliasing estimate for parallel signatures.

The key differences between the design guidelines for parallel signatures and serial signatures are:

- Multiple signature samples may be required for parallel signatures to accomplish acceptable quality levels. For serial-signature analysis a single sample may suffice if an appropriate test length and a signature register period are chosen.
- The signature register polynomials for parallel signatures are required to be primitive; whereas, for serial-signature analysis this requirement is not necessary. The only requirement that should hold good for serial-signature registers is that their period be greater than the test length; and, nonprimitive polynomials can also satisfy this requirement.
- For parallel signature designs the signature register size must be greater than or equal to the number of circuit outputs. There is no such constraint for serial-signature designs.

2 Error Models, Aliasing Probability, and Previous Research

For the derivation of the results on parallel signature analysis the following assumptions were made:

- The test stimulus is generated by random pattern generators,
- The circuit under test is combinational, and
- The faults in the circuit under test are combinational (i.e., faults do not cause memory behavior in the circuit).

A reasonable error model for single output circuits under these assumptions is the Bernoulli error model. The Bernoulli error model has been widely used by several researchers, [5], [8], [10], [11], [14], [15], [16], [18], [19], [24]. In the Bernoulli error model, output errors are assumed to occur with probability *p*, called the *detection probability* [11], in the presence of a fault. An *output error* for a particular fault is an event where the output of the faulty circuit is different from the output of the fault free circuit. The term, *output errors*, denotes a sequence of output error events for a sequence of test patterns. For a random test pattern these events are independent. For serial signatures, algorithms to calculate the exact aliasing probability [5], [8], [10], [16], [24] and simple bounds [14], [15], [16] on the aliasing probability have been derived.

A proposed extension [20], [21] of this model to multiple-output circuits is to assume independent single output combinational circuits. This allows easy extensions of the results derived for serial signature analysis [14], [15], [16] to parallel signature analysis. In [24], an exact aliasing formula for the independent error model is presented for certain test lengths. The closed-form formula in [24] relies on the independent error model and uses binary weight distribution of Reed-Solomon codes [1]. Although the independent error model does not preclude the possibility of allowing the Bernoulli error model, for each output to be characterized by a different value of *p*, the assumption of multiple independent Bernoulli error models is questionable. There is a simple reason for this. Outputs in a multiple-output circuit can share common circuitry. For faults in this common circuitry, the error behavior in the outputs will be highly correlated (see Fig. 3 and Table 1).

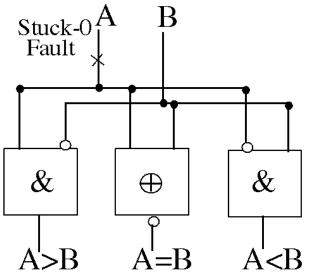

Fig. 3. One bit comparator.

TABLE 1
ONE BIT COMPARATOR'S FAULT-FREE AND FAULTY OUTPUT RESPONSES

AB	Fault-Free Output	Faulty Output	Error Vector	Integer Value *i*
00	010	010	000	0
01	001	001	000	0
10	100	010	110	6
11	010	001	011	3

The *error vector* is the bit-wise exclusive-or of the faulty and fault-free response vectors. For example, the error vector for the fault shown in Fig. 3 for input pattern AB = 11 is 011 (see Table 1). For compactness in notation we will denote error vectors by integer values (symbol *i* in Table 1).

2.1 Q-ary Error Model Review

The *q-ary error model*, [10], [24], assumes that all error vectors occur with the same probability $q = p / (2^m - 1)$, where *p* is the detection probability of a given fault and *m* is the number of circuit outputs. Although the *q*-ary model allows the possibility of correlated errors in the multiple-outputs, it cannot always be justified. Following are some scenarios where the *q*-ary model may not be applicable:

- Faults that affect only a subset of the *m*-outputs cannot produce all possible $2^m - 1$ error vectors.
- When the number of primary inputs *n* to a combinational circuit is strictly less than *m*, the number of primary outputs, the number of distinct error vectors is less than or equal to $2^n - 1$ and therefore is strictly less than $2^m - 1$. The example in Fig. 3 illustrates this.

In an *n*-input and *m*-output combinational circuit, for a given fault and for a random test pattern, the probability, p_i, of producing an error vector with integer value *i* is equal to the number of distinct test patterns, k_i, that produce these error vectors divided by the total number, $N = 2^n$, of test patterns. That is, $p_i = k_i / N$. The detectability *k* [11] for a fault is the summation of k_i for that fault for all *i*, $0 < i < 2^m$. Likewise, the detection probability *p* of a fault is the summation of p_i for that fault for all nonzero i. Table 1 illustrates the characterization of the error model for an example circuit with two-input random test patterns. For example, the error vector probabilities for Table 1 are: $p_0 = 0.5$, $p_6 = 0.25$, $p_3 = 0.25$, all other $p_i = 0$; $p = 0.5$. The assumption that all error vectors have the same probability is shown to be not true by this example.

The aliasing probability results derived using the *q*-ary model, however, can still be useful if they provide an upper-bound on the actual aliasing probability. The probability of aliasing, P_{al}, in the *q*-ary error model for test length *L*, detection probability *p*, and number of outputs *m* is given in [24, equation (12)] as

$$P_{al}(p,m,L) = 2^{-m}\left(1 - 2^m(1-p)^L + \left(2^m - 1\right)\left(1 - \frac{2^m p}{2^m - 1}\right)^L\right)$$

Using the following fact

$$\forall p \quad (1-p) > \left(1 - \frac{2^m p}{2^m - 1}\right)$$

For $p \leq \frac{2^m - 1}{2^m}$, we have

$$P_{al}(p,m,L) < 2^{-m}\left(1 - 2^m(1-p)^L + \left(2^m - 1\right)(1-p)^L\right)$$

Simplifying

$$P_{al}(p,m,L) < 2^{-m}\left(1 - (1-p)^L\right) < 2^{-m}$$

Also for $p > \frac{2^m - 1}{2^m}$, the absolute value $\left|1 - \frac{2^m p}{2^m - 1}\right| \leq \frac{1}{2^m - 1}$.

Using the foregoing inequality we have the following bound for $p > \frac{2^m - 1}{2^m}$

$$P_{al}(p,m,L) < 2^{-m}\left(1 + \left(\frac{1}{2^m - 1}\right)^{L-1}\right)$$

The above equation is a bound on aliasing probability for

q-ary model that depends only on test length L and the number of outputs m. To test the effectiveness of the q-ary model in predicting an upper-bound on aliasing probability, the following experiment was conducted. A larger example (ALU181 circuit with a multiple stuck-at fault, $n = 14$ and $m = 8$ for this circuit) was considered. The detectability profile for this fault was calculated through simulation and is shown in Table 2.

TABLE 2
DETECTABILITY PROFILE FOR A MULTIPLE STUCK-AT FAULT IN ALU181

Error Vector i	k_i	p_i	Error Vector i	k_i	p_i
1	620	0.0378	**195**	312	0.0190
2	680	0.0415	**196**	336	0.0205
3	252	0.0154	**201**	124	0.0075
4	752	0.0459	**203**	240	0.0146
137	176	0.0107	**204**	404	0.0246
139	264	0.0161	**209**	188	0.0114
140	456	0.0278	**211**	240	0.0146
145	240	0.0146	**212**	356	0.0217
147	264	0.0161	**217**	40	0.0024
148	384	0.0234	**219**	312	0.0190
153	88	0.0053	**220**	528	0.0322
155	228	0.0139	**225**	172	0.0104
156	460	0.0281	**227**	288	0.0175
161	264	0.0161	**228**	316	0.0192
163	264	0.0161	**233**	60	0.0036
164	352	0.0214	**235**	324	0.0198
169	128	0.0078	**236**	504	0.0308
171	228	0.0139	**241**	84	0.0051
172	428	0.0261	**243**	372	0.0227
177	176	0.0107	**244**	440	0.0268
179	228	0.0139	**250**	680	0.0415
180	364	0.0222	**251**	72	0.0044
187	336	0.0205	**252**	192	0.0117
188	560	0.0342	**254**	1,360	0.0830
193	248	0.0151			

Table 2 lists only those error vectors that have $p_i > 0$. This is contrary to the q-ary model because not all p_i are equal. The eight outputs of the ALU181 circuit were compacted using parallel signature design with primitive polynomial $x^8 + x^5 + x^3 + x^2 + 1$. The 14-input random test stimulus was derived by sampling 14-bits from a 32-bit primitive LFSR. The aliasing probability was estimated by 50,000 different random pattern runs of test length up to 254. That is, each experiment consisted of starting with a different seed for the 32-bit LFSR pattern generator and running up to test length 254. Within each run, aliasing events were recorded for the test length range one to 254. From the ensemble of 50,000 different experiments, the aliasing probability was estimated. Fig. 4 shows the plot of experimentally estimated aliasing versus the test length. Also shown in this figure is the upper-bound on aliasing probability (approximately 0.004 in this set-up) derived from the q-ary model closed-form expression in [24]. The standard deviation error in the estimated aliasing probability for 50,000 experiments under the hypothesis that the q-ary model aliasing probability is correct is approximately square root of 0.004 divided by square root of 50,000. This amounts to 0.0002. The experimental estimates, for many test lengths, differ by more than three standard deviations. This clearly indicates that q-ary model is not a good predictor of aliasing probability behavior in parallel signature analysis.

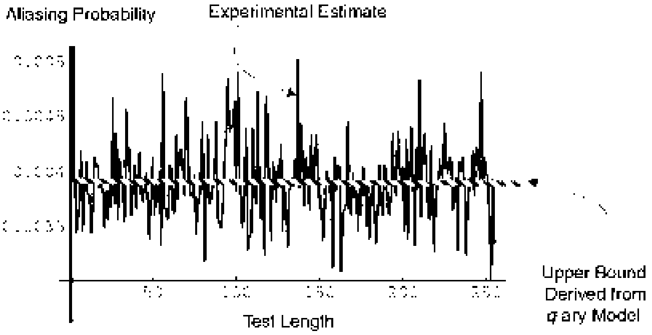

Fig. 4. Inadequacy of q-ary model in bounding aliasing.

The previous example shows that ignoring the fact that the error vectors have unequal probabilities may lead to inaccurate predictors on the aliasing probability behavior. A general error model proposed in [30] allows for the possibility of the error vectors having unequal probabilities. It must be pointed out that the q-ary model is very useful in cases where the faults in the combinational circuit satisfy the model's assumptions. This is because the aliasing probability can be studied using a simple closed-form formula derived in [24] for special test lengths and for special signature register polynomials. For general error models, arbitrary test lengths, and arbitrary signature polynomials, the analysis of aliasing is not an easy problem. Some other approaches are needed.

2.2 Simple Bounds and Their Justification

In previous work [14], [15], [16], simple bounds on the aliasing probability were derived for single-output combinational circuits, for arbitrary test lengths, and for arbitrary signature polynomials. These bounds were validated by experiments. Experimental validation is necessary because aliasing probability bounds are derived based on the assumption of random test stimulus; however, in actual test environments pseudorandom patterns are used. This paper complements the previous work on serial signature design guidelines by adding a new set of guidelines for parallel signature design. The following paragraphs justify the motivation for this work. These paragraphs very briefly examine each of the several approaches to estimate aliasing probability. These approaches fall into three categories:

- estimating aliasing probability through fault simulation,.
- deriving an analytical formula to calculate the aliasing probability,
- deriving bounds for the aliasing probability.

These are reviewed in the following sections.

2.2.1 Estimating Aliasing Using Fault Simulation

Fault simulation experiments can be performed to estimate the aliasing behavior and in most cases designs can be chosen that eliminate aliasing for the modeled faults [25]. Using simulation to experimentally estimate aliasing probability is only useful for simulated faults (usually single stuck-at faults). There is an exponential number of combinational faults like multiple stuck-at and bridging faults that also need to be simulated. There are three problems with this:

1) simulation time may be too long, even for single stuck-at faults,
2) commercial tools may not exist that simulate faults other than single stuck-at faults, and
3) simulation results do not give any insight into the signature design in terms of what signature design characteristics are suitable to reduce aliasing.

2.2.2 Analytical Formula for Aliasing Probability

Earlier papers [5], [8], [10], [16] presented methods to calculate the exact aliasing probability in serial signature analysis. In [24], closed-form formula for the aliasing probability for parallel signature analysis was derived using the q-ary error model. Section 2.1 has reviewed this in detail. A recent paper[1] [31] acknowledged the limitations of q-ary model and proposed a modified version of independent error model [24] to calculate the aliasing probability for correlated errors. Each output i of a multioutput combinational circuit is characterized by a detection probability value ϕ_i for a combinational fault. This still assumes that the error behavior at each circuit output is statistically independent. The authors in [31] derive the aliasing probability formula through coding theory techniques and arrive at the same result as in [21]. The formula of aliasing probability using this independent error model is

$$Pal = \frac{1}{2^m} + \sum_{i=1}^{2^m-1}\left(\prod_{j=0}^{m-1}\left(1-\phi_j\right)^{w_j(i,L)}\right) - p_0^L$$

$w_j(i, L)$ is the number of ones seen by jth stage during L state transitions for an autonomous LFSR starting at state i. A program was written by the first author to calculate the aliasing probability under independent error model. To validate that the formula was correctly implemented by the program, the results in [21] and [31] are reproduced in Fig. 5 and Fig. 6, respectively. Fig. 7 calculates the aliasing probability for the ALU experiment described in Section 2.1. The values of ϕ_0 through ϕ_7 are given in Fig. 7. Again, it clearly shows that the aliasing probability predicted by the independent error model differs significantly from the experimental estimates shown in Fig. 4.

The exact calculation of aliasing probability for parallel signature analysis is a more difficult problem than that for serial signature analysis. This is because serial signature analysis is a very special case of parallel signature analysis. In addition, the calculation of detection probability is also an NP-Hard problem [6], [14].

2.2.3 Previously Published Aliasing Probability Bounds

Previously published bounds [5], [8], [19] depend on the detection probability and therefore do not eliminate the NP-Hard problem associated with the calculation of detection probability. Also, the bounds presented in these papers have incomplete proofs. This is discussed in the following sections.

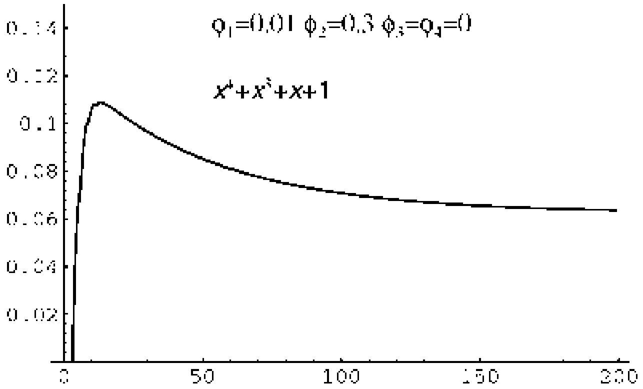

Fig. 5. Reproduction of aliasing probability plot in [21].

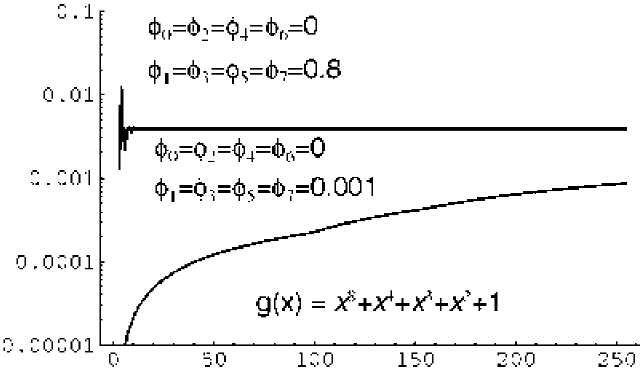

Fig. 6. Reproduction of aliasing probability plot in [31].

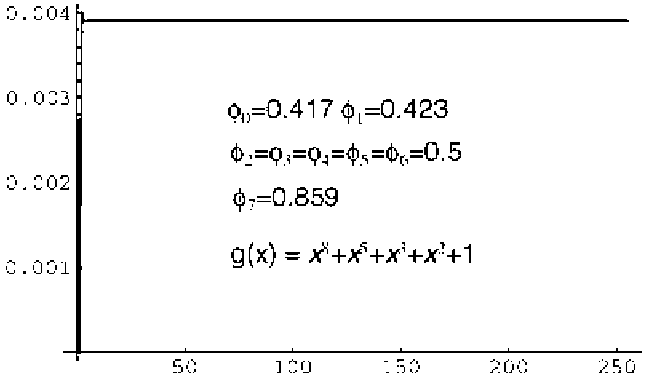

Fig. 7. Aliasing probability prediction for the ALU experiment using independent error model.

2.2.4 On the Aliasing Probability Bound in [8], [15]

In [8], an upper bound on the aliasing probability for serial signatures was derived. An attempt to clarify the proof of this bound was presented in [15]. The main approach in deriving this bound for aliasing probability was to map the problem of computing aliasing probability to the problem of computing signal probability. The method used was to show that the computation of aliasing probability is a signal probability problem involving overlapping sets of xor-trees whose outputs feed to a single or-gate. The probability of

[1] Brought to the attention of the authors during the review process.

aliasing was shown to be the same as the probability of getting signal zero at the output of the or-gate [8], [15].

By using the cutting algorithm [9], a bound on signal probability and therefore a bound on aliasing probability was derived in [8], [15]. The fallacy was in the application of the cutting algorithm. The cutting algorithm can be used in propagating probability bounds only for unate functions and it is not applicable for nonunate functions (like xor gates). It must be pointed out that fallacy in the proof does not imply that the bound derived in [8], [15] is incorrect. It only suggests that either a correct proof be presented or a counter example provided. At this time, the authors are not aware of a correct proof or a counter example.

However, the fallacy in [15] has been corrected in [16] by deriving a provably correct bound that differs from that derived in [8] but shares some essential characteristics that were necessary to derive simple bounds on aliasing in [16].

2.2.5 On the Bound Presented in [19]

The bound on aliasing probability presented in [19] although empirically appears to be valid has no valid proof backing it. Unfortunately, this bound has made its way in the text book [4, p. 445].

2.2.6 On the 4/L Bound [26]

The bounds presented in [14], [15], [16] eliminate the exponential complexity associated with the exact calculation of aliasing probability; however, they have been derived only for serial signature analysis. The simple bound, $4/L$, presented in [26], applies to parallel signature analysis but lacks a complete proof. If the results in [26] are provably correct, then this paper would not be necessary.

3 Characterizing Aliasing Probability and Simple Bounds

Based on the characterization of the error model in the previous section, the aliasing probability for parallel signature analysis can be fully characterized by the function

- $P_{al}(p_1, \dots, p_{2^m-1}, L, U(X))$, and
- the implementation of parallel signature analysis (Fig. 8, Fig. 9, and Fig. 10.

$U(X)$ is the polynomial characterizing the signature register [14], L is the test length, and p_i is the detection probability of error vectors with integer value i for a particular fault. Fig. 8, Fig. 9, and Fig. 10 give three possible implementations of parallel signature analysis. The implementation in Fig. 8, called the *xor-tree* implementation, can be easily analyzed using the results from serial signature analysis [16]. This is because the *xor*-tree in Fig. 8 converts the multioutput combinational circuit into a single-output combinational circuit.

The problem with the *xor*-tree implementation is that it cannot detect faults in the combinational circuit that always affect even number of outputs (for example, the stuck-at fault in Fig. 3). In this paper, only the implementations of Fig. 9 and Fig. 10 will be considered. In the Fig. 9 implementation, called the *parallel-load* implementation, the outputs are synchronously loaded into the feedback register stages through the *xor* gates. In the parallel-load implementation, the signature computation takes L clock cycles for L test input patterns. In the Fig. 10 implementation, called the *scan* implementation, the m-outputs are first loaded into a scan-register and then the contents of the scan-register are synchronously clocked into the signature register in m cycles.

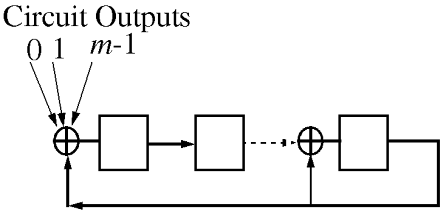

Fig. 8. XOR-tree implementation.

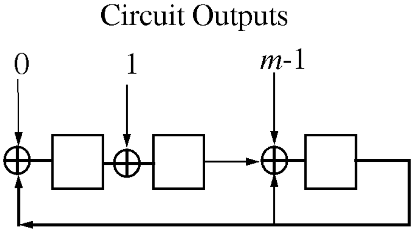

Fig. 9. Parallel-load implementation.

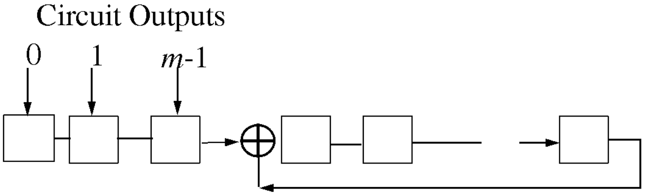

Fig. 10. Scan implementation.

For L test input-patterns, the scan implementation takes mL cycles for signature computation. However, it can be shown that for every scan implementation there exists an equivalent parallel-load implementation that requires only L cycles for signature computation. This is illustrated in Fig. 11. The proof of this equivalence is not the main subject of this paper and will be published in a separate report. The proof trivially follows from matrix algebra and this is sketched, by way of example, in Fig. 11. Matrix algebraic treatment of computations similar to signature computation can be found in [1], [27], [28].

3.1 Motivation for Simple Bounds

The calculation of exact aliasing probability for serial signature analysis was shown to be NP-Hard [14]. For a discussion of NP-Hard problems see [6]. The best known algorithms [16] require a complexity of $O(L2^r)$ for test length L and signature register size r. The calculation of the exact aliasing probability for parallel signature analysis is a much harder problem. This is because serial signature analysis (for m = 1) is a very special case of parallel signature analysis.

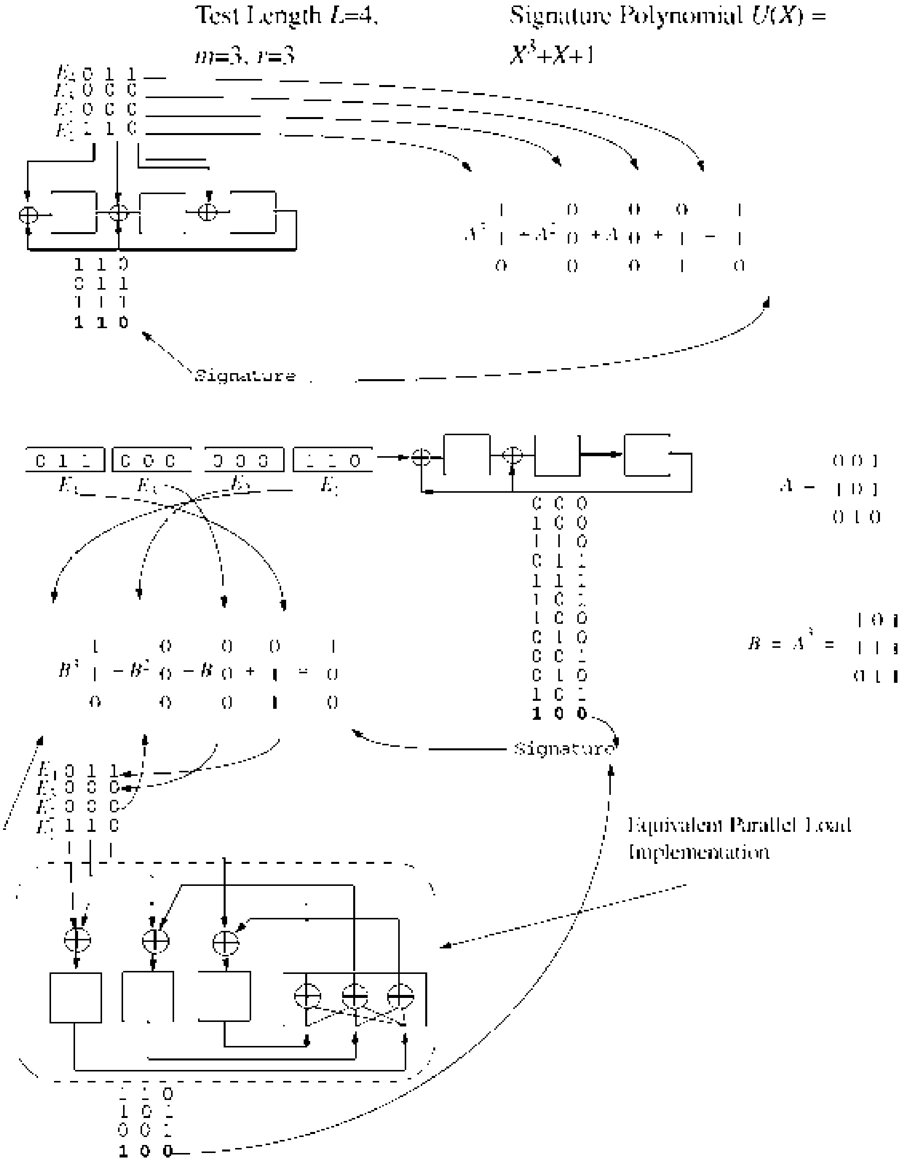
Test Length L=4, m=3, r=3
Signature Polynomial U(X) = X^3+X+1
E1
E2
E3
E4
Signature
A =
B = A^3 =
Signature
Equivalent Parallel Load Implementation

Fig. 11. Scan and parallel-load implementation equivalence.

A practical approach to quantify aliasing in parallel signature analysis is to derive some bounds on the aliasing probability denoted by $P_{al}(p_1, \ldots, p_{2^m-1}, L, U(X))$.

Associated with every combinational fault is a distribution of detection probability p_i. Given a fault, the calculation of p_i is also an NP-Hard problem because it requires the enumeration of NP-Complete decision problems on Boolean satisfiability [6]. Therefore, we need bounds

$$P_{al}\left(p_1, \ldots, p_{2^m-1}, L, U(X)\right) \le f\left(L, U(X)\right)$$

for the aliasing probability that are independent of the distribution of p_i, and only depend on the test length L and the signature register characterized by the polynomial $U(X)$. Such bounds are called *simple bounds*. Simple bounds are useful in that

- they do not require any exponential complexity in calculating aliasing probability,
- they do not require any a priori assumption of the type of combinational fault,
- they provide simple guidelines in the design of signature analysis.

4 Simple Bounds

Simple bounds on the aliasing probability for parallel signature analysis are derived in this section. The following notation and definitions are used in deriving some intermediate results. These results are used in the final derivation of the simple bounds.

4.1 Notation.

Let us denote m-bit error vectors for a response length L by $E_1, \ldots, E_L$. Let v_j denote the number of error vectors with integer value j. For an m-output circuit, j is in the range $[0, 2^m - 1]$. For example, the error vectors in Table 1 (for response length

TABLE 3
SIGNATURES FOR ERROR SEQUENCE PERMUTATIONS
USING SIGNATURE POLYNOMIAL $X^3 + X + 1$

Sequence <E>	Time 1	Time 2	Time 3	Time 4	Signature S
$<E>_1$	000	000	011	110	001
$<E>_2$	000	011	110	000	110
$<E>_3$	000	011	000	110	011
$<E>_4$	011	000	000	110	010
$<E>_5$	011	000	110	000	111
$<E>_6$	011	110	000	000	011
$<E>_7$	000	000	110	011	000
$<E>_8$	000	110	011	000	000
$<E>_9$	000	110	000	011	100
$<E>_{10}$	110	000	000	011	110
$<E>_{11}$	110	000	011	000	010
$<E>_{12}$	110	011	000	000	000

L = 4) are E_1 = 000, E_2 = 000, E_3 = 110, and E_4 = 011 (assuming the first row in the table corresponds to the first response pattern). The values for v_j in this example are $v_0 = 2$, $v_1 = 0$, $v_2 = 0$, $v_3 = 1$, $v_4 = 0$, $v_5 = 0$, $v_6 = 1$, and $v_7 = 0$.

It follows from the notation that

$$\sum_{j=0}^{2^m-1} v_j = L.$$

Given a population distribution (characterized by v_j) of error vectors E_1, ..., E_L; in a random test environment, all permutations of these error sequences are equally likely. For example, Table 3 enumerates all permutations of error vectors in the foregoing example.

DEFINITION 1. *Function* $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$ *is the number of permutations of the error vectors characterized by population distribution* v_j *that have the same signature* S *using signature polynomial* $U(X)$.

For example, Table 4 illustrates the calculation of the value of this function for $L = 4$, $U(X) = X^3 + X + 1$ (Fig. 12, parallel-load implementation), and the error vector population distribution $v_0 = 2$, $v_1 = 0$, $v_2 = 0$, $v_3 = 1$, $v_4 = 0$, $v_5 = 0$, $v_6 = 1$, and $v_7 = 0$. Using the signature values for the various sequences in Table 3, we can enumerate the values of function $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$. This is presented in Table 4.

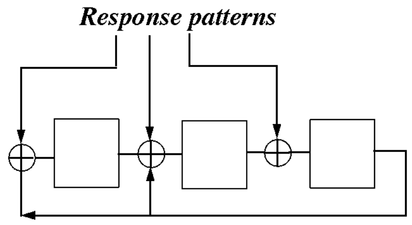

Fig. 12. Signature register using polynomial $X^3 + X + 1$.

4.2 Plan of Attack in Deriving Simple Bounds

This section sketches the methodology used in deriving simple bounds on the aliasing probability. The function $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$ enumerates the weight distribution of linear codes in GF(2^m) [1]. The aliasing probability in parallel signature analysis is the same as the probability of undetected errors in GF(2^m). The formulation of aliasing probability in terms of weight distribution also appears in [30].

Using the function $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$, the probability of aliasing, $P_{al}(p_1, \ldots, p_{2^m-1}, L, U(X))$, is given by

$$\left(\sum_{v_0+v_1+\ldots+v_{2^m-1}=L} N\left(v_0, v_1, \ldots, v_{2^m-1};\ 0, U(X)\right) p_0^{v_0} \ldots p_{2^m-1}^{v_{2^m-1}} \right) - \left(1-p\right)^L \qquad (1)$$

This is similar to the aliasing probability expression derived in [14], [15] for $GF(2)$. The probability of a particular error vector sequence with distribution $(v_0, v_1, \ldots, v_{2^m-1})$ is $p_0^{v_0} \ldots p_{2^m-1}^{v_{2^m-1}}$.

By definition there are $N(v_0, v_1, \ldots v_{2^m-1};\ 0, U(X))$ sequences with distribution $(v_0, v_1, \ldots, v_{2^m-1})$ that have

TABLE 4
FUNCTION VALUES FOR N

S	$N(2, 0, 0, 1, 0, 0, 1, 0; S, X^3 + X + 1)$
000	3
001	1
010	2
011	2
100	1
101	0
110	2
111	1

signature $S = 0$. Therefore, the probability of an error sequence with distribution $(v_0, v_1, \ldots, v_{2^m-1})$ and the signature $S = 0$ is the product $N(v_0, v_1, \ldots, v_{2^m-1};\ 0, U(X))p_0^{v_0} \ldots p_{2^m-1}^{v_{2^m-1}}$. The probability of any error sequence having signature $S = 0$ is obtained by summing the product,

$$N(v_0, v_1, \ldots, v_{2^m-1};\ 0, U(X))p_0^{v_0} \ldots p_{2^m-1}^{v_{2^m-1}},$$

over all distributions $(v_0, v_1, \ldots, v_{2^m-1})$ such that $v_0 + v_1 + \ldots + v_{2^m-1} = L$. This summation includes the probability, $(1 - p)^L$, of an all-zero error sequence. Subtracting the all-zero error probability from the above summation we get (1).

Theorem 1 derives an upper bound on the function $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$. The upper bound derived in Theorem 1 uses a *sphere packing bound* [28] approach. The basic idea is to use permutations of error sequences that produce a specific signature S with population distribution specified by v_j. Now each permutation, denoted by **<E>**, is used in generating new error sequences that differ in exactly one error vector from **<E>**. Consider that these new error sequences surrounding **<E>** are enclosed in a sphere. Theorem 1 proves (an intermediate result) that all the spheres produced by considering all the permutations of error sequences that produce a specific signature S are nonintersecting. That is, the spheres do not have any common error sequence. This intermediate result is used by Theorem 1 to establish an upper bound on $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$.

Theorem 2 uses the bound derived by Theorem 1 and the aliasing probability expression (given in (1)) in deriving a closed-form aliasing probability bound. Theorem 3, in turn, uses the closed-form bound derived by Theorem 2 and another closed-form bound on the aliasing probability to derive a family of simple bounds.

4.3 Derivation of Simple Bounds

THEOREM 1. *Given a degree r primitive polynomial and test length* $L < 2^r - 1$, *consider a sequence of L, m-bit error vectors* $E_1, \ldots, E_L$, *such that there are* v_j *vectors with integer value j in the population of these L vectors and*

$$\sum_{j=0}^{2^m-1} v_j = L.$$

The number of permutations $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$ *of these L vectors that produce a particular signature S*

- *in the parallel-load implementation for* $r \geq m$
- *in the scan implementation for* $r \geq m$ *and* $2^r - 1$ *relatively prime to m*

is bounded above by

$$min\left\{\frac{1}{v_i+1}\binom{L}{v_0, v_1, \ldots v_{2^m-1}}\right\}\left(\forall j\left(v_j > 0\right)\forall(i \neq j) \in \left[0,\ 2^m - 1\right]\right)$$

where the multinomial coefficient

$$\binom{L}{v_0, v_1, \ldots, v_{2^m-1}} = \frac{L!}{v_0!\,v_1!\ldots!\,v_{2^m-1}!}$$

is the number of different length L sequences with the error vector distribution specified by $(v_0, v_1, \ldots, v_{2^m-1})$.

PROOF. Let A be the $r \times r$ matrix characterizing the parallel-load implementation of the primitive signature polynomial $U(X)$. The signature of the error vector sequence $E_1, \ldots, E_L$ is given by the matrix polynomial $A^{L-1}E_1 + A^{L-2}E_2 + \ldots + E_L$ (see Fig. 11 for illustration), if E_1 is the first vector in the sequence. Let us assume that a particular error vector sequence $E_1, \ldots, E_L$ produces a specific signature $S = A^{L-1}E_1 + A^{L-2}E_2 + \ldots + E_L$.

By definition there are $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$ distinct permutations of this length L error vector sequence that produce the same signature.

Pick a particular sequence **<E>** = $E_1, \ldots, E_L$ that produces signature S. Choose an integer j such that the v_j is greater than zero. Arbitrarily pick an integer i (not equal to j) that corresponds to an error vector. By definition we are given that there are v_i such error vector terms. Now, replace one of the error vectors in the sequence **<E>** having integer value j with an error vector having integer value i. This generates a new sequence that differs from **<E>** in one vector. Repeating this procedure for all locations corresponding to error vector j will generate v_j new distinct error sequences from the sequence **<E>**. The new error sequences will have $v_i + 1$ error terms with integer value i and $v_j - 1$ terms with integer value j. Table 5 (derived from the example presented in Table 3) illustrates this replacement procedure for $j = 0$, $i = 7$, and $S = 0$. The first column lists all the error vector sequence permutations, with $v_0 = 2$, $v_3 = v_6 = 1$, $v_1 = v_2 = v_4 = v_5 = v_7 = 0$, (each error vector is denoted by its integer value) that produce signature $S = 0$. Columns 2 and 3 in Table 5 are obtained by replacing the first and second occurrence of error vectors respectively with value $j = 0$ in column 1 by error vector with value $i = 7$.

TABLE 5
GENERATING NEW SEQUENCES THROUGH REPLACEMENT FOR $j = 0$, $i = 7$, $S = 0$

<E>	replace first	replace second
0, 6, 3, 0	7, 6, 3, 0	0, 6, 3, 7
6, 3, 0, 0	6, 3, 7, 0	6, 3, 0, 7
0, 0, 6, 3	7, 0, 6, 3	0, 7, 6, 3

Next, we prove that all the new sequences generated by the above procedure are distinct. Let us consider a sequence **<E*>** ≠ **<E>** that is a permutation of sequence **<E>** and has the same signature S. Now we will show that replacing error vectors with integer value j in **<E*>** by the error vector that has value i cannot produce a sequence that is same as that generated by considering error sequence **<E>** (Table 5 illustrates this). The proof is by contradiction. Let us assume that it is possible to produce the same sequence from error sequences **<E>** and **<E*>**. Let us assume that in sequence **<E>** the location t corresponding to the error vector E_t with integer value j is replaced by the error vector with integer value i. Likewise assume that in sequence **<E*>** the location k corresponding to the error vector E_k with integer value j is replaced by the error vector with integer value i. This is shown in Fig. 13. Assume that they generate the same new sequence. Notice that k cannot be equal to t because this implies that both sequences **<E>** and **<E*>** are the same. The case of interest is k not equal to t. Since we assumed that the new sequences are the same it implies (follows from the illustration in Fig. 13) that location t in **<E*>** has an error vector that has integer value equal to i. Likewise, it follows that location k in **<E>** has an error vector with value i.

This means $E_t = E_k^*$ with integer value j and $E_k = E_t^*$ with integer value i in the sequences **<E>** and **<E*>**, respectively. Since sequences **<E>** and **<E*>** have the same signature S, by linearity this implies that the signature of the sequence generated by the bit-by-bit exclusive-or sum of the error sequences **<E>** and **<E*>** will be zero. This bit-by-bit *xor* operation will result in only two nonzero terms because the error vectors in the shaded areas of the illustration (in Fig. 13) are identical both in value and location. For $C = E_t + E_t^* = E_k + E_k^*$ (C is not equal to zero as the error vectors i, j are distinct by definition), in matrix polynomial terms this means $A^{L-t}C + A^{L-k}C = 0 \Rightarrow A^{k-t}C = C$ (Algebra is in modulo 2 arithmetic). □

LEMMA 1. *If A is a matrix representing a degree r primitive polynomial and C is a nonzero vector, then for $z > 0$, $A^zC = C$ if and only if z is a nonzero multiple of* $2^r - 1$.

PROOF. We are given that matrix A corresponds to a degree r primitive polynomial. Since we are given a primitive polynomial, for every nonzero vector C, there exists an integer w ($0 < w < 2^r$) [1], [28] such that the vector E can be represented by α^w, where α is a nonzero primitive element in the Galois field defined by the degree r primitive polynomial. Note that such a representation may not exist for a nonprimitive polynomial. The vector obtained by considering the matrix product AC is represented by α^{w+1}. This is because matrix multiplication by A is essentially a multiplication by the primitive field element α. Also, addition of vectors corresponds to addition of corresponding field elements in the Galois field. From this discussion, it follows that $A^zC = C$ if and only if $\alpha^{w+z} = \alpha^w$. By canceling the common factor α^w, this equality is satisfied if and only if $\alpha^z = 1$. Note that the cancellation is allowed because α is a nonzero field element. Since α is a primitive element in the Galois field defined by a degree r primitive polynomial and $z > 0$, the equality $\alpha^z = 1$ is possible *if and only if* z is a nonzero multiple of $2^r - 1$ [1]. □

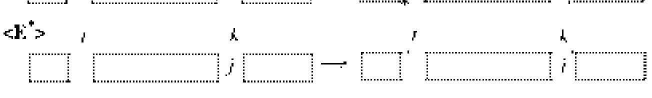

Fig. 13. Error sequences illustration.

Without any loss in generality assume $k > t$. This implies $k - t > 0$. With k and t less than L and $L < 2^r - 1$, it follows that $k - t < L < 2^r - 1$. Since we are given that the signature polynomial is primitive, from Lemma 1 the equality $A^{k-t}C = C$ is only possible *if and only if* $k - t$ is a nonzero multiple of $2^r - 1$. This contradicts the fact that $k - t < L < 2^r - 1$.

This replacement procedure generates

$$v_j N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$$

distinct permutations (Table 5 illustrates this) with the following error vector population

$$v_0, v_1, \ldots, v_i + 1, \ldots, v_j - 1, \ldots, v_{2^m-1}.$$

This cannot be greater than the multinomial coefficient

$$\binom{L}{v_0, v_1, \ldots v_i + 1, \ldots, v_j - 1, \ldots, v_{2^m-1}}$$

Therefore,

$$N\left(v_0, v_1, \ldots, v_{2^m-1}; S, U(X)\right) \le \frac{1}{v_j}\binom{L}{v_0, v_1, \ldots v_i + 1, \ldots v_j - 1, \ldots v_{2^m-1}}$$

Since any value of $i \ne j$ can be chosen, after some algebra, we get a family of upper bounds on $N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$. They are

$$N\left(v_0, v_1, \ldots, v_{2^m-1}; S, U(X)\right) \le \frac{1}{v_i + 1}\binom{L}{v_0, v_1, \ldots, v_{2^m-1}}\left(\exists j \forall (i \ne j) \in \left[0,\ 2^m - 1\right]\right)$$

Therefore, the tightest upper bound on

$$N(v_0, v_1, \ldots, v_{2^m-1}; S, U(X))$$

from this family of bounds is

$$min\left\{\frac{1}{v_i + 1}\binom{L}{v_0, v_1, \ldots, v_{2^m-1}}\right\}\left(\forall j\left(v_j > 0\right)\forall (i \ne j) \in \left[0,\ 2^{m-1}\right]\right)$$

This *min* value is obtained by picking the *max* value of v_i for some i and j in $[0, 2^m - 1]$ such that i is not equal to j and $v_j > 0$.

For the scan implementation of parallel signature analysis, the signature of the error sequence is given by $B^{L-1}E_1 + B^{L-2}E_2 + \ldots + E_L$, where $B = A^m$ and A is the matrix characterizing the primitive signature polynomial $U(X)$. If m is relatively prime to $2^r - 1$, the matrix B also will have a primitive characteristic polynomial. Now we can use the same proof we developed for parallel-load implementation.

ILLUSTRATIVE EXAMPLE. From Table 3, we have $N(2, 0, 0, 1, 0, 0, 1, 0; S = 000, X^3 + X + 1) = 3$. By picking the *max* value $v_0 = 2$ ($i = 0$ and $j = 3$ or 6) and using Theorem 1, we have

TABLE 6
SIGNATURE FOR ERROR SEQUENCE WITH EQUAL AND NONZERO VECTORS

Sequence	Time 1	Time 2	Time 3	Time 4	Signature
1	001	001	001	001	011
2	010	010	010	010	110
3	011	011	011	011	101
4	100	100	100	100	001
5	101	101	101	101	010
6	110	110	110	110	111
7	111	111	111	111	100

$$N\left(2,\ 0,\ 0,\ 1,\ 0,\ 0,\ 1,\ 0;\ 0, X^3 + X + 1\right) \le \frac{1}{2+1}\binom{4}{2,\ 1,\ 1} = 4$$

The exact value $N(2, 0, 0, 1, 0, 0, 1, 0; S = 000, X^3 + X + 1) = 3$ satisfies the above bound.

In certain cases, the bound derived in Theorem 1 gives the exact value. For example, the error sequence **<E>** $= E_1, E_2, E_3, E_4$ = 001, 001, 001, 010 (Fig. 12 signature register implementation) has signature $S = 0$. Here $v_1 = 3$, $v_2 = 1$, and $v_0 = v_3 = v_4 = v_5 = v_6 = v_7 = 0$. Any other permutation of this error sequence has a nonzero signature. Thus, in this case, the function $N(0, 3, 1, 0, 0, 0, 0, 0; S = 000, X^3 + X + 1) = 1$. Picking the *max* value $v_1 = 3$ (with $i = 1$, and $j = 2$) by Theorem 1, we have

$$N\left(0,\ 3,\ 1,\ 0,\ 0,\ 0,\ 0,\ 0;\ 0, X^3 + X + 1\right) \le \frac{1}{3+1}\binom{4}{3,\ 1} = 1,$$

which is the same as the exact value.

LEMMA 2. *Given a degree r primitive polynomial* ($r \ge m$, *m is the number of outputs*) *and a test of length* $L < 2^r - 1$, *the signature of an error sequence with nonzero error vectors* $E_1 = E_2 = \ldots = E_L = E$ *is nonzero.*

PROOF. The signature of this error sequence is given by the matrix polynomial $S = A^{L-1}E + A^{L-2}E + \ldots + E$, where A is the matrix representation of degree r primitive polynomial. Since we are given a primitive polynomial, for every nonzero vector E there exists an integer j ($0 < j < 2^r$) [1], [28] such that the vector E can be represented by α^j, where α is a nonzero primitive element in the Galois field defined by the degree r primitive polynomial. Note that such a representation may not exist for a nonprimitive polynomial. The vector obtained by considering the matrix product AE is represented by α^{j+1}. This is because matrix multiplication by A is essentially a multiplication by the primitive field element α. Also, addition of vectors corresponds to addition of corresponding field elements in the Galois field. From this discussion, it follows that the signature S of the error sequence is represented by the field element $\alpha^{L-1+j} + \alpha^{L-2+j} + \ldots + \alpha^j = \alpha^j(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1)$. The signature S is zero if and only if the corresponding field element $\alpha^j(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1)$ is zero. We will assume that the signature S is zero and show that such an assumption leads to a contradiction. If $S = 0$, then $\alpha^j(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1) = 0$. Since α is a nonzero element of the Galois field, $\alpha^j(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1) = 0$ only if $(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1) = 0$. Now let us multiply $(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1)$ by $(1 + \alpha)$. Because $(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1) = 0$, we have$(1 + \alpha)(\alpha^{L-1} + \alpha^{L-2} + \ldots + 1) = 1 + \alpha^L = 0$. This implies $\alpha^L = 1$. Since we know that α is a primitive element of the Galois field defined by a degree r primitive polynomial, the equality is possible *if and only if* L is a nonzero multiple of $2^r - 1$. This contradicts the assumption that L is less than $2^r - 1$. Therefore, the signature S cannot be zero. □

Table 6 illustrates the nonzero signature values of nonzero error sequences with the same error vectors for $L = 4$ and signature polynomial $X^3 + X + 1$ (Fig. 12 implementation).

THEOREM 2. *Given a degree r primitive polynomial; test length L* $< 2^r - 1$; *a combinational fault with detection probability distribution* $p_0, p_1, \ldots, p_{2^m-1}$; *and k in* $[0,\ 2^m - 1]$ *such that* p_k *is the max of* $p_0, p_1, \ldots, p_{2^m-1}$, *then the aliasing probability* $P_{al}(p_1, \ldots, p_{2^m-1}, L, U(X))$

- *in the parallel-load implementation for* $r \ge m$
- *in the scan implementations for* $r \ge m$ *and* $2^r - 1$ *relatively prime to m*

is bounded above by

$$P_{al}\left(p_1, \ldots, p_{2^m-1}, L, U(X)\right) \le \frac{1}{p_k(L+1)} - (1-p)^L.$$

PROOF. From Theorem 1, it follows (we can pick v_k that corresponds to p_k) that the aliasing probability is bounded above by

$$P_{al}\left(p_1, \ldots, p_{2^m-1}, L, U(X)\right) \le$$
$$\sum \frac{1}{v_k+1}\binom{L}{v_0, v_1, \ldots, v_{2^m-1}} p_0^{v_0} \cdots p_{2^m-1}^{v_{2^m-1}} - (1-p)^L$$

The above summation includes the cases of error sequences with nonzero error vectors $E_1 = E_2 = \ldots = E_L = E$ for all nonzero E. We know from Lemma 2 that these nonzero error sequences with $E_1 = E_2 = \ldots = E_L = E$ for all nonzero E do not produce a zero signature and therefore do not cause aliasing. Keeping these error sequences in the summation simplifies the algebra required to derive a closed-form aliasing probability formula. The expression $(1 - p)^L$ is subtracted in the above aliasing probability bound to exclude the probability of an all-zero error sequence. The above summation is over all v_i such that $0 \le v_0, v_1, \dots v_{2^m-1} \le L$ and $v_0 + v_1 + \dots + v_{2^m-1} = L$. This summation has exactly the same number of terms as the expansion of the polynomial $(p_0 + p_1 + \dots + p_{2^m-1})^L$ in multinomial coefficients [29].

We will use the multinomial coefficient expansion

$$\left(p_0 + p_1 + \dots + p_{2^m-1}\right)^L = \sum \binom{L}{v_0, v_1, \dots, v_{2^m-1}} p_0^{v_0} \dots p_{2^m-1}^{v_{2^m-1}}$$

to derive a closed-form bound on the aliasing probability. If we integrate the multinomial expansion, treating p_k as an auxiliary variable, we have

$$\int_0^{p_k} \left(p_0 + p_1 + \dots + p_{2^m-1}\right)^L dp_k =$$

$$p_k \sum \frac{1}{v_k + 1} \binom{L}{v_0, v_1, \dots, v_{2^m-1}} p_0^{v_0} \dots p_{2^m-1}^{v_{2^m-1}}$$

After some algebra we have

$$\sum \frac{1}{v_k + 1} \binom{L}{v_0, v_1, \dots, v_{2^m-1}} p_0^{v_0} \dots p_{2^m-1}^{v_{2^m-1}} =$$

$$\frac{1}{p_k(L+1)} - \frac{(1-p_k)^L}{p_k(L+1)}$$

Ignoring the second term in the right hand side of the above expression and excluding the all-zero error sequence probability, $(1 - p)^L$, we have the following bound on the aliasing probability

$$P_{al}\left(p_1, \dots, p_{2^m-1}, L, U(X)\right) \le \frac{1}{p_k(L+1)} - (1-p)^L$$

The scan implementation has the same bound on the aliasing probability as that of the parallel-load implementation but puts an additional constraint on the primitive signature register size r such that $2^r - 1$ and m (the number of outputs) are relatively prime. Given any value of m, a smallest value of $r \ge m$ can be found such that $2^r - 1$ and m are relatively prime. Table 7 illustrates some values. □

TABLE 7
SMALLEST r ($r \ge m$) SUCH THAT $\gcd(m, 2^r - 1) = 1$

m	*r*
2	2
3	3
7	7
21	23
31	31
32	32
42	43
63	65

The results proved thus far do not yet provide a simple bound on the aliasing probability since they depend on the detection probability. Theorem 3 provides a family of simple bounds on the aliasing probability building upon results derived in Theorems 1 and 2.

THEOREM 3 *If y uniformly spaced signature samples $S_1, S_2, \dots, S_y$ are taken from the response of an m-output combinational circuit by a signature register defined by a primitive polynomial of degree r, then the simple bound, $f(L, U(X))$, on the aliasing probability for*

- *the parallel-load implementation with $r \ge m$, and*
- *the scan implementation with $r \ge m$ and m relatively prime to $2^r - 1$ is*

$$f\left(L, U(X)\right) \le \frac{y^{y/2}}{L^{y/2}}$$

PROOF. For a given test length L, consider the signature at test length $L - 1$. The aliasing probability is always bounded above by the maximum conditional probability that the signature of the error vector sequence at test length L is zero given a particular signature at test length $L - 1$. Independent of the signature value at test length $L - 1$, this conditional aliasing probability is always bounded above by $p_k = max\ p_i$. Using the results in Theorem 2, the worst case bound on the aliasing probability is obtained by

$$f\left(L, U(X)\right) \le min\left\{\frac{1}{p_k(L+1)} - (1-p)^L, p_k\right\}$$

The expression $min\left\{\frac{1}{p_k(L+1)} - (1-p)^L, p_k\right\}$ attains maximum value when

$$p_k = \frac{1}{p_k(L+1)} - (1-p)^L$$

Solving for p_k we have the following bound on the root, z, of the above equation.

$$z \le \frac{1}{\sqrt{L+1}} \approx \frac{1}{\sqrt{L}}$$

Therefore, a simple bound on the aliasing probability is

$$f\left(L, U(X)\right) \le \min\left\{\frac{1}{z(L+1)} - (1-p)^L, z\right\} \le \frac{1}{\sqrt{L+1}} \approx \frac{1}{\sqrt{L}}$$

If we take y equally spaced samples of signatures at test interval lengths of L/y, then, by the statistical independence of random test vectors, aliasing occurs if all the signature samples alias. The probability for each sample to alias is bounded above by $(y/L)^{1/2}$. Therefore, the probability that all y signature samples $S_1, S_2, \dots, S_y$ alias is bounded above by $(y/L)^{y/2}$. Also, the constraint on test length L for y samples is $L < y(2^r - 1)$. □

5 CONCLUSIONS

This paper addressed design issues related to parallel signature analysis. The results in this paper present a simple relationship between the number of outputs m in the circuit under test, the random pattern test length L, the primitive signature register size r, the number of signature register samples y, and the aliasing probability. It is shown that the aliasing probability is less than $(y/L)^{y/2}$ if $r \geq m$ and $2^r - 1 > L/y$. For example, if m = 32, then using two signatures samples (y = 2), a test length L = 2,000,000, a primitive signature register of size r = 32 will guarantee the aliasing probability to be less than one in a 1,000,000. If only one signature sample (y = 1) is used, then the aliasing probability is guaranteed to be less than one in 1,414. Readers can note that for y = 1, the results in this paper show that the aliasing probability is less than $1/\sqrt{L}$ for a random pattern test length L and a primitive signature register period greater than L. This is not as tight as the $1/L$ bound for serial-signatures. Researchers are encouraged to improve this bound. Clearly, serial-signature bounds are not applicable to parallel signature analysis. Fig. 2 demonstrates this by comparing the experimental estimates of aliasing probability for parallel signature analysis with the $1/L$ bound for serial-signature analysis. The key differences between the design guidelines for parallel signatures and serial signatures are:

- Multiple signature samples may be required for parallel signatures to accomplish acceptable quality levels. For serial-signature analysis a single sample may suffice if an appropriate test length and a signature register period are chosen.
- The signature register polynomials for parallel signatures are required to be primitive; whereas, for serial-signature analysis, this requirement is not necessary. The only requirement that should hold good for serial-signature registers is that their period be greater than the test length; nonprimitive polynomials can also satisfy this requirement.
- For parallel signature designs the signature register size must be greater than or equal to the number of circuit outputs. There is no such constraint for serial-signature designs.

For parallel signature analysis scheme, by using two signature samples we can guarantee that the aliasing probability to be less than $2/L$. If more than two signature samples are used, the aliasing probability can be significantly reduced. Experimental results in [13], [25] demonstrate this.

Acknowledgments

The authors would like to thank Siegrid Munda for the administrative support in preparing this manuscript. This work was supported in part by the Ballistic Missile Defense Organization, Innovative Science and Technology (BMDO/IST) Directorate, and administered through the U.S. Department of the Navy, Office of Naval Research under Grant No. N00014-92-J-1782, by the U.S. National Science Foundation under Grant No. MIP-9107760, and by the Advanced Research Projects Agency under prime contract No.*DABT63-94-C-0045.

References

[1] E.R. Berlekamp, *Algebraic Coding Theory*, revised edition. Aegean Park Press, 1984.

[2] S.Z. Hassan, D.J. Lu, and E.J. McCluskey, "Parallel Signature Analyzers—Detection Capability and Extensions," *Proc. 26th IEEE CS Int'l Conf., COMPCON, Spring 1983*, pp. 440-445, Feb. 1983.

[3] R. David, "Signature Analysis of Multi-Output Circuits," *Digest of Papers 14th Ann. Int'l Symp. Fault-Tolerant Computing*, pp. 366-371, June 1984.

[4] M. Abramovici, M.A. Breuer, and A.D. Friedman, *Digital Systems Testing and Testable Design*. Computer Science Press, 1990.

[5] M. Damiani et al., "An Analytical Model for the Aliasing Probability in Signature Analysis Testing," *IEEE Trans. Computer-Aided Design*, vol. 8, no. 11, pp. 1,133-1,144, Nov. 1989.

[6] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W.H. Freeman and Co., 1978.

[7] P. Gelsinger et al., "Computer Aided Design and Built-In Self-Test on the i486TM CPU," *Proc. ICCD*, pp. 199-201, Oct. 1989.

[8] A. Ivanov. and V.K. Agarwal, "An Analysis of the Probabilistic Behavior of Linear Feedback Signature Registers," *IEEE Trans. Computer-Aided Design*, vol. 8, no. 10, pp. 1,074-1,088, Oct. 1989.

[9] J. Savir et al., "Random Pattern Testability," *IEEE Trans. Computers*, vol. 33, no. 1, pp. 79-90, Jan. 1984.

[10] K. Iwasaki and N. Yamaguchi, "Design of Signature Circuits Based on Weight Distribution of Error-Correcting Codes," *Proc. ITC*, pp. 779-785, Sept. 1990.

[11] E.J. McCluskey et al., "Probability Models for Pseudorandom Test Sequences," *IEEE Trans. Computer-Aided Design*, vol. 7, no. 1, pp. 68-74, Jan. 1988.

[12] I.M. Ratiu and H.B. Bokoglu, "Pseudorandom Built-In Self-Test Methodology and Implementation for the IBM RISC System/6000 Processor," *IBM J. Research and Development*, vol. 34, no. 1, pp. 78-84, Jan. 1990.

[13] N.R. Saxena, "Test Compression Methods," MS thesis, Univ. of Iowa, May 1984.

[14] N.R. Saxena, E.J. McCluskey, and P. Franco, "Bounds on Signature Analysis Aliasing for Random Testing," *Proc. FTCS*, pp. 104-111, 1991.

[15] N.R. Saxena, P. Franco, and E.J. McCluskey, "Refined Bounds on Signature Analysis Aliasing for Random Testing," *ITC '91 Proc.*, pp. 818-827, Oct. 1991.

[16] N.R. Saxena, P. Franco, and E.J. McCluskey, "Simple Bounds on Serial Signature Analysis Aliasing for Random Testing," Special Issue on Fault Tolerant Computing, *IEEE Trans. Computers*, vol. 41, no. 5, pp. 638-645, May 1992.

[17] M. Serra et al., "The Analysis of One-Dimensional Linear Cellular Automata and Their Aliasing Properties," *IEEE Trans. Computer-Aided Design*, vol. 9, no. 7, pp. 767-778, July 1990.

[18] J.J. Shedletsky, "Random Testing: Practicality vs. Verified Effectiveness," *Proc. 17th Ann. Int'l Conf. Fault-Tolerant Computing*, pp. 175-179, June 1977.

[19] T.W. Williams et al., "Aliasing Errors in Signature Analysis," *IEEE Design and Test of Computers*, pp. 39-45, Apr. 1987.

[20] T.W. Williams et al., "Aliasing Errors in Multiple Input Signature Analysis Registers," *Proc. European Test Conf.*, pp. 338-345, Apr. 1989.

[21] M. Damiani et al., "Aliasing in Signature Analysis Testing with Multiple-Input Shift Registers," *Proc. European Test Conf.*, pp. 346-353, Apr. 1989.

[22] N. Benowitz et al., "An Advanced Fault Isolation System for Digital Logic," *IEEE Trans. Computers*, vol. 24, no. 5, pp. 489-497, May 1975.

[23] R.A. Frohwerk, "Signature Analysis: A New Digital Field Service Method," *Hewlett-Packard J.*, pp. 2-8, May 1977.

[24] D.K. Pradhan and S.K Gupta, "A New Framework for Designing and Analyzing BIST Techniques and Zero Aliasing Compression," *IEEE Trans. Computers*, vol. 40, no. 6, pp. 743-763, June 1991.

[25] I. Pomeranz, S.M. Reddy, and R. Tangirala, "On Achieving Zero Aliasing for Modeled Faults," *Proc. EDAC*, 1992.

[26] J.L. Carter, "The Theory of Signature Testing for VLSI," *Proc. 14th ACM Symp. Theory of Computing*, pp. 66-76, San Francisco, May 1982.

[27] A. Gill, *Linear Sequential Circuits—Analysis, Synthesis, and Applications*. McGraw-Hill, 1966.

[28] W.W. Peterson and E.J. Weldon Jr., *Error-Correcting Codes*, second edition. The MIT Press, 1984.

[29] R.L. Graham, D.E. Knuth, and O. Patashnik, *Concrete Mathematics-A Foundation for Computer Science*. Addison-Wesley, 1989.

[30] M.G. Karpovsky, S.K. Gupta, and D.K. Pradhan, "Aliasing and Diagnosis in MISR and STUMPS Using General Error Model," *ITC '91 Proc.*, pp. 828-839, Oct. 1991.