Testability Measures and Test Complexities for Testing with Internal Access*

M.G.KARPOVSKY, *Fellow, IEEE* and V.N.YARMOLIK
Research Laboratory of Design and Testing of Computer Hardware
Department of Electrical, Computer and Systems Engineering
Boston University, Boston, MA 02215

Abstract: This paper presents test sets, testability measures and attainable lower and upper bounds for numbers of test patterns for several important classes of networks for testing methods providing massive observability.

1 Introduction

New test techniques have recently been developed that provide visibility, either virtually or directly, to the logic node or even the electrical node level, which ensure massive observability in a circuit-under-test. Such techniques are referred to as *internal access test methods*. One of the most attractive techniques among the internal access test methods is monitoring of the quiescent power supply current (I_{ddq} testing) for static CMOS circuit [4]. In the following sections we will considere I_{ddq} testing only since the results may be easily modified for other internal access test methods [3]. I_{ddq} testing is slow. Typically, less than 1,000 test patterns for I_{ddq} testing can be applied. Thus, the problem of generation of short (minimal) tests is very important for I_{ddq} testing.

2 Fault Model

In this paper we will concentrate our investigation on bridging faults (BFs) as the most common fault model covering physical defects in CMOS circuits. [5, 6].

Detection of bridging faults between logic nodes in CMOS circuits by I_{ddq} testing have been investigated by several authors [3, 5, 6, 7, 8].

Stuck-at faults can be represented as BFs between the logic node and V_{dd} and V_{ss} nodes. Thus, BFs can be considered as the universal fault model, which cover stuck-at faults, as well as several previously intractable or types of faults [3, 8, 9].

The number P of BFs for a circuit with N logic nodes is given by

$$P = 2N + \binom{N}{2} = (N^2 + 3N)/2. \qquad (1)$$

This number includes AND and OR types BFs between any two logic nodes, V_{dd} and V_{ss} nodes.

The conditions for bridge detectability by I_{ddq} testing have been investigated by several authors [5, 1, 3, 10].

3 Test Matrices

For a test with T test patterns for a given network with m inputs and N nodes we denote by H_I the $(T \times m)$ *input matrix*, where rows of H_I are test patterns and by H the $(T \times N)$ *test matrix*, where ith row of H represents logical values at all nodes of the network for ith test pattern ($i = 1, \ldots, T$).

Theorem 3.1 *All BFs of the circuit-under-test are detectable iff all columns in a test matrix, H, are different and not equal to all ones or all zeros, where a row in, H, represents an input test pattern and the responses on internal and output nodes.* □

The number of rows, T, in a test matrix, H, determines the I_{ddq} test complexity.

Theorem 3.2 *The I_{ddq} test complexity, T, for detecting all BFs for a nonredundant circuit with m input, p internal and n output nodes, satisfies the following inequalities*

*This work was supported by the NSF under Grant MIP-9208487 and the NRC under Grant "Testing of Computer Hardware".

$$\lceil \log_2(m+p+n+2) \rceil \leq T \leq \lceil \log_2(m+2) \rceil + p + n, \quad (2)$$

where $N = n + m + p$.

We note that lower and upper bounds (2) given by Theorem 3.2 are attainable. The lower bound is attained for the case of one gate with $N-1$ inputs and the upper bound for the case of a two-level AND-OR networks with orthogonal product terms.

According to Theorem 3.2, test complexity for detection of all $P = (N^2 + 3N)/2$ BFs satisfies the inequality $T \leq N$.

4 Fault Coverages

Denote by $h_i = (h_{i1}, \ldots, h_{iN})$ the ith row, ($i \in \{1, 2, \ldots, T\}$) of test matrix, H, and by $w_i \in \{0, 1, 2, \ldots, N\}$ the row weight (number of 1's) in h_i. We note that h_i is generated by the ith test pattern. Then, the number, M_i, of BFs which are detected by the ith test pattern is

$$M_i = N + w_i(N - w_i). \quad (3)$$

The minimal number, M_{min}, of BFs detected by a test pattern is N and the maximal, M_{max}, is $N + \lceil N/2 \rceil (N - \lceil N/2 \rceil) \approx (N^2 + 4N)/4$. An average value of M for uniformly distributed h_i is determined by

$$\begin{aligned} M_{av} &= N + \lceil (N^2 - N)/4 \rceil \\ &\approx (N^2 + 3N)/4. \quad (4) \end{aligned}$$

The fraction of BFs that are detected by i randomly selected test patterns has the mean value $1 - 2^{-i}$.

5 Testability Measures

As a measure $C(z_i, z_j)$ of testability of a BF between nodes z_i and z_j we propose the probability of setting these nodes z_i and z_j to the opposite values, i.e. $C(z_i, z_j) = p(z_i \oplus z_j = 1)$ by one randomly chosen test pattern. For BFs between logic node z_i and V_{dd} and V_{ss} we have $C(1, z_i) = p(1 \oplus z_i = 1) = p(\overline{z_i} = 1)$ and $C(0, z_i) = p(0 \oplus z_i = 1) = p(z_i = 1)$. High values (close to 1) of testability $C(z_i, z_j)$ indicate a good testability of the BF between the nodes z_i and z_j.

For the testability measure $C(z_i, z_j)$ for any two nodes z_i and z_j we have

$$\begin{aligned} &0 \leq C(z_i, z_j) \leq 1, C_{min}(z_i, z_j) = |p(z_i) - p(z_j)| \\ &C_{max}(z_i, z_j) = 1 - |p(z_i) - p(\overline{z_j})|, \end{aligned} \quad (5)$$

where $p(z_i) = p(z_i = 1)$ is a signal probability of node z_i, which could be determined, for example, by the cutting algorithm.

It should be mentioned that for practical cases it is too difficult to determine the exact values of $C(z_i, z_j)$ for all (z_i, z_j). As a more realistic and practical approach to evaluate testability we propose to use an average value of $C(z_i, z_j)$ which is determined by signal probabilities $p(z_i)$ and $p(z_j)$ of two nodes z_i and z_j as

$$C^*(z_i, z_j) = p(z_i)p(\overline{z_j}) + p(\overline{z_i})p(z_j), \quad (6)$$

where $p(\overline{z_i}) = 1 - p(z_i)$.

It is easy to show that $C^*(z_i, z_j)$ defined by (6) equals to $C(z_i, z_j)$ for the case of independent random variables z_i and z_j and is a good approximation of $C(z_i, z_j)$ for the general case. Then by (5)

$$C_{min}(z_i, z_j) \leq C^*(z_i, z_j) \leq C_{max}(z_i, z_j). \quad (7)$$

If $p(z_i) = 1/2$, then for any $p(z_j)$ testability measure $C_{av}(z_i, z_j) = 1/2$, for $p(z_i) = 1$, $C(1, z_j) = 1 - p(z_j)$, and for $p(z_i) = 0$, $C(0, z_j) = p(z_j)$.

A mathematical expectation for the number of test patterns can be determined as a minimal T such that:

$$\sum_{i>j} (1 - C(z_i, z_j))^T \geq \binom{N}{2} (1 - FC), \quad (8)$$

where FC is the fault caverage for BFs.

The complexity of computing test lengths by is of the order $O(N^2)$. To simplify the computations we will estimate the average value C for $C(z_i, z_j)$. By (6) and (7) we have

$$\begin{aligned} C &= \binom{N}{2}^{-1} \sum_{i>j} C^*(z_i, z_j) \\ &= \binom{N}{2}^{-1} (N_0 N_1 - \sum_i p_i(1 - p_i)), \quad (9) \end{aligned}$$

where $N_0 = \sum_{i=1}^{N} p_i$ and $N_1 = N - N_0$.

Thus $R = N_0 N_1 - \sum_{i=1}^{N} p_i(1 - p_i)$ can be used as a global testability measure for the whole device-under-test. The complexity of computing this measure is $O(N)$ only. Substituting (9) in (8) we have

$$T = \lceil (\log_2(1 - FC))(\log_2(1 - \left(\begin{array}{c} N \\ 2 \end{array} \right)^{-1} R))^{-1} \rceil . \tag{10}$$

For example, for the 8AND gate implemented as the tree of 2AND gates and as the 1 dim array of 2AND gates we have for testability measure $R_{tree} = 47.76$, $R_{1dim} = 44.81$ and $T_{tree} = 7.7$ and $T_{1dim} = 8.4$. (Minimal numbers of test patterns for these networks are 6 and 9 correspondingly.)

6 Test Pattern Generation and Test Complexities

6.1 Input/Output Test Sets

The fact that I_{ddq} testing provides for a good observability implies that for every BF the number of test vectors detecting this fault will be rather high. Some evidence for the above conjecture was presented in [2, 10, 11, 13, 14]. We will prove this result for several important classes of networks.

Let us consider test sets for input-output (IO) BFs. In this case for the first $q = \lceil \log_2(m+2) \rceil$ rows of test matrix H the ith column corresponding to input variable x_i is the binary representation of i, $(i = 1, \ldots, m)$. If column $F_j, (j = 1, \ldots, n)$ is equal to the column, corresponding x_i or to the column of all zeros or all ones, then H includes an additional row generated by a test pattern such that $F_j(x_1, \ldots, x_m) \oplus x_i = 1$.

For primitive m-input gates (AND, OR, NAND and NOR)

$$T = \begin{cases} \log_2 m + 2; & m = 2^k, \\ \lceil \log_2 m \rceil + 1; & m \neq 2^k, \quad k = 1, 2, \ldots . \end{cases}$$

For m-bit adder the IO test set consists of

$$T = \lceil \log_2 m \rceil + 3 \tag{11}$$

test patterns, presented in Table 1 for $m = 8$, where $Z = X + Y$.

The optimal test set for $m = 2^k$-bit multiplier has complexity $T = k + 3$. The test set for the 4-bit multiplier is also shown in Table 1.

6.2 Test Sets for Arrays of Disconnected Combinational Networks

For the case of an array of disconnected identical combinational networks (for example an array of q m-input NAND gates), it is easy to prove the following theorem.

Table 1: Test sets for 8-bit adders and 4-bit multipliers

$Z = X + Y$		
X	Y	Z
00000000	00000000	00000000
11111111	00000000	11111111
00000000	11111111	11111111
11110000	11110000	01111000
11001100	11001100	01100110
10101010	10101010	01010101

$Z = X \cdot Y$		
X	Y	Z
0001	0111	00001110
1110	1000	11100000
1100	0011	00100100
1010	0110	01111000
0011	1101	00100001

Theorem 6.1 *An array of q identical disconnected networks, where each network is described by a set of boolean functions $F_j = F_j(x_1, \ldots, x_m)$, $j = (1, \ldots, r)$ and there exist an input pattern $a = (a_1, \ldots, a_m)$, such that $\overline{F_j}(\overline{a}_1, \ldots, \overline{a}_m) = F_j(a_1, \ldots, a_m)$, for $j = 1, \ldots, r$, is I_{ddq} testable by*

$$T = T_F + \lceil \log_2 q \rceil \tag{12}$$

test patterns, where T_F is a number of test patterns for the network implementing $\{F_j, j = (1, \ldots, r)\}$.

6.3 Tests for Linear Circuits

For the case of linear circuits consisting of XOR gates only the following theorem is true.

Theorem 6.2 *Input matrix, H_I, for any m-input nonredundant linear circuit implementing*

$$F_j(x_1, x_2, \ldots, x_m) = \bigoplus_{i=1}^{m} c_i^j x_i; \quad j = 1, 2, \ldots n \tag{13}$$

Remark. The same test can be used for OR networks and AND networks and for multiple bridgings.

where $n \leq 2^m - 1$, $c_i^j \in \{0,1\}$ and for any j at least one $c_i^j \neq 0$, has the following standard form

$$H_I = \begin{vmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{vmatrix}. \qquad (14)$$

□

The complexity of this test is $T = m$. For the case when there exist j such that all $c_i^j = 1$, the test matrix contains an extra row of all zeros. For $n = 2^m - 1$ this test is optimal.

Let us consider now a special class of linear networks with m inputs where each output depends on at most l input variables.

Theorem 6.3 *Universal input matrix $H_I(m, l)$ for all m-input nonredundant linear circuit where each output depends on at most l input variables is a parity-check matrix of a code of length m and a distance $d = 2l + 1$.* □

Table 2: Minimal numbers of test patterns T_{32}, T_{64}, T_{128} for XOR networks with 32, 64, and 128 inputs where every output depends on at most l inputs

l	2	3	4	5	6	7	8	9	10
T_{32}	10	15	19	21	25	26	29	30	30
T_{64}	12	17	24	28	34	36	42	45	46
T_{128}	14	21	28	35	42	49	56	57	64

Using the Varshamov-Gilbert bound [15] we have

$$T \leq \lceil \log_2 \sum_{i=0}^{2l-1} \binom{m-1}{i} \rceil. \qquad (15)$$

Let us consider now I_{ddq} testing of trees of XOR gates.

Theorem 6.4 *For any binary XOR tree with m inputs a number of test patterns is upperbounded by*

$$T \leq 2\lceil \log_2 m \rceil + 1. \qquad (16)$$

6.4 Tests for Trees of Primitive Gates

In this section we will construct tests for trees consisting of g-input ($g \geq 2$) AND, NAND, OR, NOR gates.

For the tree primitive g-input NAND gates we have

$$T = (g+1)\lceil \log_g m \rceil. \qquad (17)$$

For the binary case we have $T = 3\lceil \log_2 m \rceil$.

As it was shown in [16] lower bounds for T, as $m \to \infty$ for g-ary OR or AND trees are:

$$T \geq \begin{cases} 1.29\lceil \log_2 m \rceil + 1, & \text{for } g = 2; \\ 1.64\lceil \log_3 m \rceil + 1, & \text{for } g = 3; \\ 2\lceil \log_4 m \rceil + 2, & \text{for } g = 4; \\ \log_2 g \lceil \log_g m \rceil + 1, & \text{for } g \geq 5. \end{cases} \qquad (18)$$

Using results from [16] we have for $g = 2$

$$T \leq \begin{cases} \frac{3}{2}\lceil \log_2 m \rceil + 1, & \text{for } m = 2^{4k}; \\ \lceil \frac{3}{2} \log_2 m \rceil + 2, & \text{otherwise}; \end{cases} \qquad (19)$$

which is close to lower bound (18).

Table 3: Minimal values of test patterns, T, for m-input OR (AND) trees

m	2	4	8	16	32	64	128	256
T	3	5	6	7	9	10-11	11-12	13

6.5 Tests for Two-Level Unate Circuits

In this section we consider two-level AND-OR (NAND-NAND, NOR-NOR) circuits implementing unate functions $F = F(x_1^*, \dots, x_m^*)$, where x_i^* denotes either x_i or $\overline{x_i}$ and the expression for F contains only x_i or $\overline{x_i}$ for $i = 1, \dots, m$.

Theorem 6.5 *An input matrix for a m-input two-level AND-OR nonredundunt circuit implementing unate function $F = F(x_1^*, \dots, x_m^*)$ consists of m columns $|h_{1i} \dots h_{ii} \dots h_{mi}\ h_{m+1\ i}|^{TR}$ equal to $|\overbrace{1\,1 \dots \underbrace{0}_{i} \dots 1\,1}^{m+1}|^{TR}$ if $x_i^* = x_i$ or to $|0\,0 \dots \underbrace{1}_{i} \dots 0\,0|^{TR}$ if $x_i^* = \overline{x_i}$ $(i = 1, \dots, m)$.*

If F equals to 1 for all input patterns defined by these m columns, then one more test pattern is required such that for this pattern $F = 0$. □

For the test complexity, T, for the procedure described by Theorem 6.5 we have

$$m + 1 \leq T \leq m + 2. \tag{20}$$

Example 6.1 As an example let us construct a test matrix for the two-level AND-OR circuit implementing $F = \overline{x_1}x_2 + \overline{x_1}x_3 + x_2x_3$. Then by Theorem 6.5 we have

$$H = \left| \begin{array}{cccccccc} x_1 & x_2 & x_3 & \overline{x_1} & \overline{x_1}x_2 & \overline{x_1}x_3 & x_2x_3 & F \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right|,$$

As we can see all columns in H are different and not equal to all zeros and all ones, except the last column. We should add one extra row corresponding to $x_1x_2x_3 = 000$ since $F(0,0,0) = 0$. □

References

[1] M.W.Levi, "CMOS Is Most Testable", *Proceedings International Test Conference*, 1981, pp.217-220.

[2] R.R.Fritzemeier, J.M.Soden, R.K.Treece, C.F.Hawkins, "Increased CMOS IC Stuck-at Fault Coverage with Reduced I_{ddq} Test Sets", *Proceedings International Test Conference*, 1990, pp.427-434.

[3] W.H.Debany, "Measuring the Coverage of Node Shorts by Internal Access Methods", *Digest of Papers IEEE VLSI Test Symposium*, 1991, pp.215-220.

[4] W.Maly, P.Nigh, "Built-In Current Testing - Feasibility Study", *Proceedings ICCAD-88*, Santa Clara CA, November 1988, pp.340-343.

[5] E.Isern, J.Figueras, "Analysis of I_{ddq} Detectable Bridges in Combinational CMOS Circuits", *Proceedings International Test Conference*, 1994, pp.386-373.

[6] S.W.Bollinger, S.F.Midkoff, "On Test Generation for I_{ddq} Testing of Bridging Faults in CMOS Circuits", *Proceedings International Test Conference*, 1991, pp.598-607.

[7] R.Rajsuman, "An Analysis of Feedback Bridging Faults in MOS VLSI", *Digest of Papers IEEE VLSI Test Symposium*, 1991, pp.53-58.

[8] S.Chakravarty, P.J.Thadikaran, "Simulation and Generation of I_{ddq} Tests for Bridging Faults in Combinational Circuits", *Digest of Papers IEEE VLSI Test Symposium*, 1993, pp.25-32.

[9] P.Nigh, W.Maly, "Test Generation for Current Testing", *Proceedings European Test Conference*, 1989, pp.194-200.

[10] S.Chakravarty, "On Computing Tests for Bridging and Leakage Faults: Complexity Results and Universal Test Sets", *Proceedings of the Fifth International Conference on VLSI Design*, 1992, pp.49-54.

[11] R.Rajsuman, "Transistor-Level Fault Modeling and Testing", Artech House, Boston, 1992.

[12] M.Abramovici, M.A.Breuer, A.D.Friedman, "Digital Testing and Testable Design", Computer Science Press, New York, 1990.

[13] M.G.Karpovsky, S.Su, "Detection and Location of Input and Feedback Bridging Faults", *IEEE Transactions on Computers*, C-26, N6, June 1980, pp.523-528.

[14] M.G.Karpovsky, S.Su, "Detection of Bridging and Stuck-at Faults at Input and Output Pins of Standard Computer Components", *Proceedings of 17th Design Automation Conference*, 1980, pp.494-506.

[15] F.J.Macwilliams, N.J.A.Sloane, "The Theory of Error-Correcting Codes" Noth Holland Publishing Company, Amsterdam, 1977.

[16] M.G.Karpovsky, L.B.Levitin, F.S.Vainstein, "Diagnosis by Signature Analysis of Test Responses", *IEEE Transactions on Computers*, C-43, N2, February 1994, pp.141-152.