A New BIST Scheme with Encoding Logic to Achieve Complete Fault Coverage

Ling Zhang, Junjin Mei, Guan-zhong Wang, Tonghan Li

School of Computer,
Hubei Polytechnic University,
Huangshi, 435003, China

Abstract—**Built-in Self Test(BIST)has been proved as one of the effective design for testability techniques, where on-chip test architectures are designed to test the digital circuits themselves. To reduce test application time and improve fault coverage, A deterministic Built-in Self Test(BIST) technique that can get complete fault coverage without using any storage device is proposed in this paper. The test architecture contains a novel on chip encoding logic that generates all required test vectors in real time. Experimental results show that the proposed technique requires much less test application time to achieve complete fault coverage for all testable stuck-at faults with reasonable hardware cost.**

Keywords-Built-in Self Test(BIST); test application time; fault coverage

I. INTRODUCTION

Logic BIST architecture commonly uses linear feedback shift registers(LFSRs) as test generator for Pseudo-random testing. The architecture is simple and effective. However, because complex circuits often contains some hard-to-detect faults that random-pattern resistant, the conventional BIST often requires long test application time, and hard to achieve high fault coverage[1].To solve these problems, many of techniques have been proposed. Test point insertion[2-4], weighted pattern testing[5-8], and reseeding[9-11] are three most popular logic BIST schemes.

The test point insertion scheme inserts test points into circuits under test(CUT) intrusively to improve random testability. Timing sequence of the CUT is affected at the same time, which is not desired for high performance or critical designs. For biggest circuits, there are lots of unspecified bits in the corresponding test cubes. Reseeding techniques take advantage of this fact, which use a Linear Feedback State Regiter(LFSR) to encode for the specified bits for a test cubes by solving the linear equation, and gain the appropriate seed. To generate more effective random patterns, the weighted random pattern testing uses a weighted bit stream to bias the weight of the bit stream to inputs of circuit under testing. For achieving the target fault coverage, many weights must be stored in an ROM then resulting in large are overhead.

There are other BIST schemes also to reduce test data volume, test power or test application time. Paper[12] Proposes a input vector monitoring scheme for concurrent BIST for on-line testing. To reduce hardware cost of paper[12], a new Multilevel Decoding Logic is proposed in paper[13]. Paper [14] presents a novel BIST architecture for comute the number of switch of the circuits to reduce the test power. Paper[15] illustrates a BIST architecture for fault diagnosis that can be used to identify permanent failures in embedded ROM. A Complete Logic BIST Technology with No Storage Requirement was presented in paper[16]. Though these BIST schemes reduce the size of test set efficiently, and some need not storage memory, the long test application time and the high hardware cost are also the difficulties of the BIST.

In this paper, a novel deterministic BIST is proposed to reduce test application time and improve fault coverage. A pre-generated test set is encoded as a combinational circuit called as Encoding Logic Unit(ELU). It can generates all required test vectors without use storage device, and corresponding test application time is reduce more compared with the previous BIST schemes with reasonable hardware cost.

The rest of the paper is organized as follows. The proposed BIST scheme is described in Section.2. In Section 3, The simplify of module of ELU is presented. Experimental results are given in Section.4. Finally, we conclude our paper in section 5.
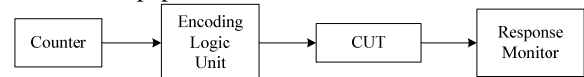

Figure 1. the architecture of the deterministic BSIT

II. PROPOSED BIST SCHEME

The schematic of the proposed deterministic BIST is given in Figure 1. It contains a Encoding Logic Unit(ELU), a counter and the response monitor. All required test vectors are generated by the ELU along with the counter. Test responses are captured by the response monitor. The deterministic test sets are encoded as the unit of the ELU, then just a simple counter can generate all vectors for the CUT. The CUT's primary inputs and scan cells are connected to the Encoding Logic Unit directly, and the ELU is designed as a simple combinational circuit that contains m outputs and $\log_2 n$ inputs, where m equals to the number of inputs of the CUT and n equals to the number of vectors. During test process, each value of the counter corresponds to a test vector. Thus, all required test vectors can be generated through the operation of the counter until achieve 100% fault coverage.

TABLE I. TEST VECTORS AND RESPONSE OF C17 BENCHMARK

	Input Test matrix	Output response matrix
T1	11111	10
T2	10010	00
T3	00101	01
T4	01010	11

The critical point of the proposed architecture is the ELU design. The number of inputs of the CUT is assumed as m, and the number of pre-generated deterministic vectors is n. The design of ELU can be described as follows: to encode these test vectors, the number of inputs for the ELU is designed as $\log_2 n$, and the number of outputs is designed as m that for applied these vectors to the CUT. All minterms of the inputs make up the inputs value of the ELU, and all outputs could be calculated by the logic operation of these minterms.

To clearly present the ELU design, an example of well-known ISCAS'85 C17 benchmark' test set[10] is given in Table II. The test set consists of four test vectors that denoted as *Input Test matrix* in the table, and the corresponding responses that presented as *Output Response matrix* are also given in the table. The number of vectors n=4, and the number of inputs m=5. The number of the ELU's inputs is computed as $\log_2 n = \log_2 4 = 2$, the number of outputs is equal to the number of inputs of CUT that is 4. To design the ELU for this test set of C17 benchmark circuit, the truth table of the ELU is listed in Table II. As shown the table, all minterms of the inputs formed all values of the ELU's inputs, so each output of the ELU can be computed easily from the minterms of these inputs. The outputs computation results for Table I are shown as expression(1) expression(2), expression(3) and expression(4).

$$y_1 = \sum(M1, M2) = \overline{x1}\,\overline{x2} + x1\overline{x2} \qquad (1)$$

$$y_2 = \sum(M1, M4) = \overline{x1}\,\overline{x2} + x1x2 = \qquad (2)$$

$$y_3 = y_5 = \sum(M1, M3) = \overline{x1}\,\overline{x2} + \overline{x1}x2 \qquad (3)$$

$$y_4 = \sum(M1, M2, M4) = \overline{x1}\,\overline{x2} + \overline{x1}x2 + x1x2 = \overline{x1} + \overline{x1}x2 = \overline{x1} + x2 \qquad (4)$$

TABLE II. THE TRUTH TABLE OF ELU FOR EXAMPLE C17

inputs			outputs				
minterms	x1	x2	y1	y2	y3	y4	y5
M1	0	0	1	1	1	1	1
M2	0	1	1	0	0	1	0
M3	1	0	0	0	1	0	1
M4	1	1	0	1	0	1	0

The ELU architecture of the proposed scheme for C17 is given in Figure 2. As shown in the figure, corresponding to truth table of Table II, the unit contains two inputs *x1* and *x2* that are actuated by a simple counter which can share with the on- chip counters, five outputs *y1* to *y5* connect directly to the inputs of CUT. All

minterms are also given in the figure, which are denoted as M1, M2, M3 and M4. The operation process can be described as follows: the counter generates each minterm of the inputs, which will generates a determined test vector through the ELU architecture. Until all minters generated, then the determined test sets are all applied to the CUT with 100% fault coverage. To reduce the hardware cost, the architecture could be optimized by simplify schemes, and several simplify schemes are given in next section.
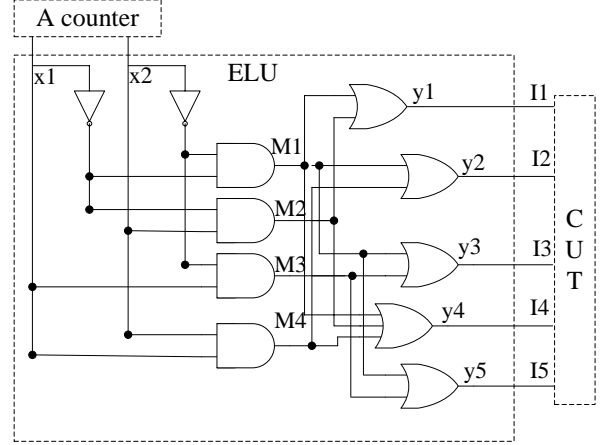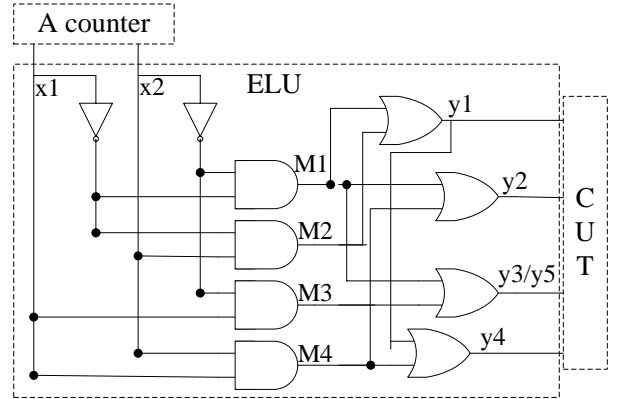

Figure 2. the ELU architecture example of C17

Figure 3. the simplified ELU for C17 using compatible columns merging

III. SIMPLIFY ALGORITHM FOR ELU UNIT

The intuitively scheme to reduce the number of logic gates is merging the compatible columns of the test set. As shown in table I, the logic value of y3 and y5 are always same, so they could be merged to as one output broadcast to two inputs input 3 and input 5 of CUT. As shown in Figure.3, the number of output reduced to 4, and the number of logic gate are decreased to 10. If the combinational ELU module with n inputs and m outputs is considered, it encodes for a pre-computed test set of CUT with 100% fault coverage. The test set contains x columns, which contains j compatible columns. Then the

number of outputs are reduced to x-j+1, at the same time, the logic gate will be reduced by j-1.

The second simplify scheme to reduce hardware cost of ELU is the Boolean algebra simplifying methods. By doing this, some redundancy variable will disappear from the logic expression of the outputs, and the corresponding architecture sure becomes simple. Take C17 for example too, the expression of $y1$ presented as $y_1 = \overline{x1x2} + x1\overline{x2}$.

Using the rule of $(\overline{A} + A) = 1$ (A is a Boolean variable) $y_1 = \overline{x1}\overline{x2} + x1\overline{x2} = (\overline{x1} + x1)\overline{x2} = \overline{x2}$, where

variable of $x1$ disappear from the expression by Boolean algebra simplifying scheme. The other outputs of the ELU are simplified in the same way, the results are given as expression(5), expression(6) and expression(7). The last architecture is given in Figure 4, the output is reduced as 4, and the number of logic gates is reduced to 6.

$$y_2 = \overline{x1\overline{x2}} + x1x2 \tag{5}$$

$$y_3 = y_5 = \overline{x1\overline{x2}} + \overline{x1}x2 = \overline{x1}(\overline{x2} + x2) = \overline{x1} \tag{6}$$

$$y_4 = \sum(M1, M2, M4) = \overline{x1}\overline{x2} + \overline{x1}x2 + x1\overline{x2} = \overline{x1} + \overline{x1}x2 = \overline{x1} + x2 \tag{7}$$
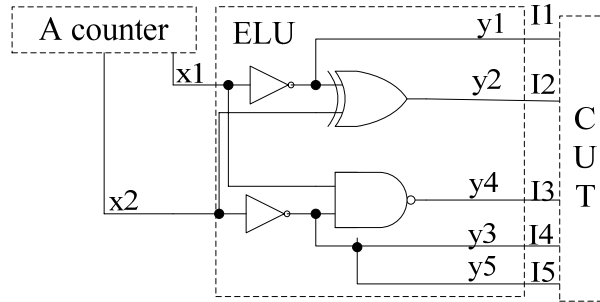

Figure 4. The simplified ELU for C17 using Boolean algebra simplifying

IV. EXPERIMENTAL RESULTS

In this section, the experimental results for the proposed BIST for ISCAS'85 and ISCAS'89 benchmarks are implemented. Test sets generated by ATALANTA[17] is used in the experiments. The simplified schemes for ELU module are achieved by using GUN C programming language. The experimental results are given in Table III.

Table III gives the experimental results of test application time, the test application time and the hardware costs are both given in the table. The first three columns give the circuit name(*CN*), the number of

ports(#*P*), the number of testable faults, the number of cycles required is denoted as #*Cycles*. The test application time is compared with related BIST solutions without storage memories, as shown in the table, they contains paper[2], paper[9], paper[11] and paper[16]. The last column is the test application time and the hardware cost of ours where the hardware cost is measured by the number of gate they contain. It is easy to see that the proposed scheme needs much less test application time with reasonable.

TABLE III. TEST APPLICATION TIME COMPARISON RESULTS

CN	#PI	#TF	Test application time #Cycles				proposal	
			[2]	[9]	[11]	[16]	#Cycles	cost
C432	36	523	1024	-	-	279	51	95
C499	41	750	1024	-	-	298	54	185
C880	60	942	2048	1K	980	278	52	215
C1355	41	1566	4096	2K	1046	595	85	238
C1908	33	1870	8192	4K	3327	900	110	217
C2670	233	2630	6144	5K	1002	343	110	766
C3540	50	3291	32768	4.5K	-	919	152	309
C5315	178	5293	4096	-	-	507	122	702
C6288	32	7710	1024	-	-	40	32	890
C7552	207	7419	32768	8K	3958	737	209	1053

As can be seen from Figure 1, Figure 2 and Table I, The optimal results are very close. But the Cpu time of Approach 1 is far less than that of Approach 2. It can be seen that the proposed algorithm.

V. CONCLUSIONS

The paper proposes a new novel deterministic BIST architecture without using any storage memory. It contains a counter which can shared with the counter on-chip, ELU module that used to generate all test vectors for

achieve 100% fault coverage. To reduce the area cost of the ELU, two simplify schemes are also given in the paper. The experimental results also proved the effectiveness of the proposed scheme.

VI. ACKNOWLEDGEMENTS

REFERENCES

[1] L.Lai, W.Cheng, T.Rinderknecht. programmable scan-based Logic Built-In Self Test. *IEEE Asian Test Symposium*, pp.371-777.2007.

[2] L. Lai, J. H. Patel, T. Rinderknecht and W.-T. Cheng.Logic BIST with Scan Chain Segmentation. *International Test Conference*, pp.57-66,2004.

[3] M.Chen. D.Xiang. Pseudorandom scan BIST using improved test point insertion techniques. *International Conference on Solid-State and integrated Circuits Technology Proceedings,ICSICT*, pp: 2043-2046,2004.

[4] S.Yang, D.Nadeau, Benoit and N.Touba. Reducing test point area for BIST through greater use of functional flip-flops to drive control pins. *Proc. IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 20-28, 2009.

[5] S.Wang, K.Balakrishnan, ST.Chakradhar. XWRC: Externally-loaded weight random pattern testing for input test data compression. *In Proc International Test Conference*, pp.571-580,2005.

[6] C.Yu, S.Reddy and I.Pomeranz. Circuit independent weighted pseudo-random BIST pattern generator, *Proc.in Asian Test Symposium*, pp.132-137, 2005.

[7] W.Wang, J.Kuang, P.Liu, et.al. Switing activity reduction for scan-based BSIT using weighted scan input data. *IEICE Electronics Express*,9(10), pp.874-880, 2012.

[8] pashchalis. Antonis, Voyiatzis.loannis, Gizopoulos.Dimitris, Accumulator based 3-weight pattern generation, *IEEE Transcations on Very Large Scale Integration Systems*,20(2), pp.357-361, 2012.

[9] C. Fagot, O. Gascuel, P. Girard and C. Landrault.A RingArchitecture Strategy for BIST Test Pattern Generation.*Journal of Electronic Testing-Theory & Applications*,19(3), pp.223-231, 2003

[10] T.Liu, J.Kuang, S.Cai, Z.You, An effective logic BIST scheme based on LFSR-reseeding and TVAC, *International Journal of Electronics*, 2013.

[11] E. Kalligeros, X. Kavousianos, D. Bakalis and D. Nikolos. On-the-Fly Reseeding: A New Reseeding Technique for Test-Per-Clock BIST. *Journal of Electronic Testing-Theory & Applications*, 18(3), pp. 315-332, 2002.

[12] I. Voyiatzis, C.Efstathiou, H.Antonopoulou. Anoval SRAM-cell based input vector monitoring concurrent BIST architecuture.*IEEE European Test Symposium*, pp.206. 2011.

[13] T.Wu, H.Liu, P.Liu, et.al. A Cost-efficient Input Vector Monitoring Concurrent On-line BIST Scheme Based on Multilevel Decoding Logic. *J.Electron Test*,29(1), pp.585-600. 2013.

[14] X.Chen. N.Wu, W.H,and W.Shan. BIST design of power switch. *IEICE Electronics Express*, 10(15).2013.

[15] M.Nilanjan,P.Artur, R.Janusz,et.al. BIST-based fault diagnosis for read-only memories. *IEEE transactions on Computer-Aided Design of Integrated Circuits and Systems*.30(7),pp.1072-1085, 2011.

[16] W.Lien, K.Lee. A Complete Logic BIST Technology with No Storage Requirement. *IEEE Asian Test Symposium*, pp.129-134.2010.

[17] Lee HK, Ha DS On the generation of test patterns for combinational circuits. Dept Elect Eng, *Virginia Polytechnic* Inst State Univ, Blacksburg, Virginia, Tech Rep, pp 12–93 (1993)