Efficient BIST Design for Sequential Machines using FiF-FoF values in Machine States

S Roy[1] U Maulik[1] S Bandyopadhyay[2] S Basu[3] Biplab K Sikdar[3]

[1]Department of Computer Science & Technology, Kalyani Govt. Engineering College, Kalyani, India 741235 {samir, ujjwal_maulik}@kucse.wb.nic.in
[2]Machine Intelligence Unit, Indian Statistical Institute, 203 B.T. Road, Calcutta 700035, India sanghami@www.isical.ac.in
[3]Department of Computer Science & Technology, Bengal Engineering College (D U), Howrah, India 711103 biplab@becs.ac.in

Contact Author : S Roy, samir@kucse.wb.nic.in

Abstract— **This paper introduces a novel BIST-quality metric termed as the $FiF - FoF$ (*Fan-in-Factor & Fan-out-Factor*) defined on FSM-states. Based on the $FiF - FoF$ analysis, an efficient scheme is presented that ensures all state codes appear with uniform likelyhood at the *present state* (PS) lines during the test phase. This results in higher fault efficiency in a $BIST$ structure. Experimental results on MCNC benchmarks show that the scheme improves fault efficiency of sequential circuits significantly, with marginal area overhead.**

I. Introduction

Pseudo random pattern generators (PRPGs) report poor fault efficiencies in sequential circuits so that it is difficult to design efficient BIST for them[1], [2]. Attempts were made to improve the BIST quality of sequential circuits [3], [4]. In [5] a synthesis scheme based on degree-of-freedom (DOF) analysis in FSM-states targeting BIST quality and gate area is reported.

In this paper we propose a new metric, $FiF - FoF$ (*Fan-in-Factor & Fan-out-Factor*), to quantify the BIST quality of an FSM state. A state encoding scheme, based on the $FiF - FoF$ analysis, is presented that makes all state codes equalliy likely to appear on PS lines during the test phase that results in higher fault efficiency.

Rest of the paper is organized as follows. The causes of poor BIST quality of sequential machines are analysed in *Section II.* In *Section III* the concept of *Fan-in-Factor & Fan-out-Factor* has been introduced. The proposed BIST scheme of FSMs targeting improved testability is presented in *Section IV. Section V* reports the experimental results and is followed by *Section VI* that concludes the paper.

II. Causes of Low BIST Quality

The general structure of a synchronous sequential machine (FSM) is shown in *Fig.1.* It consists of a combinational circuit (CC) and the system register (SR). The outputs $y_1, y_2, \cdots, y_k$ from the k memory elements of SR define the present state (PS) of the machine.

This research work is supported by the AICTE sponsored project No. 8020/RID/R&D-142/01-02

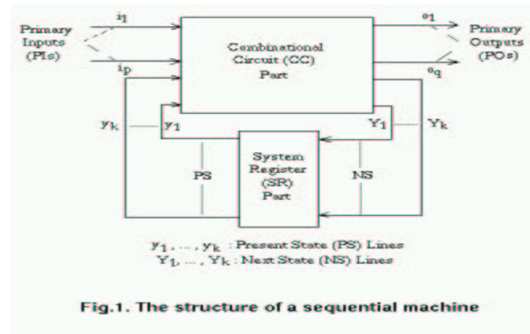


Fig.1. The structure of a sequential machine

FSMs have low BIST quality due to three kinds of states, viz., the *unreachable* states, the *hard-to-reach* states, and the *hard-to-exit* states [5]. We now explain the aforementioned factors.

A. The *Unreachable* States

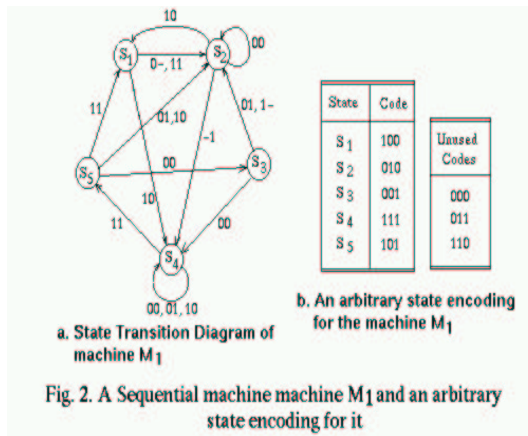
In the STG of an FSM, if there are n states, then at least k flip-flops are required to encode the n states, where $2^{k-1} < n \leq 2^k$. Out of 2^k number of state codes, only n of them will be assigned to the states of the FSM. The remaining $(2^k - n)$ unused codes are referred to as the *unreachable* states of FSM.

Example 1: Fig.2.b shows a random state code assignment for the FSM in *Fig.2.a.* The codes 000, 011 and 110 represent the unreachable states. □

B. The *Hard-to-reach* States

An FSM may contain a large number of states with very few transitions falling on them. These are called the *hard-to-reach* states.

Example 2: In *Fig.2.a* there is only one incoming transition on the state S_3, whereas, there are four outgoing transitions from this state. Therefore, S_3 is recognized to be a *hard-to-reach* state. □



State	Code
S_1	100
S_2	010
S_3	001
S_4	111
S_5	101

Unused Codes
000
011
110

a. State Transition Diagram of machine M_1

b. An arbitrary state encoding for the machine M_1

Fig. 2. A Sequential machine machine M_1 and an arbitrary state encoding for it

C. The *Hard-to-Exit* States

Again, some states have too many self loops. They act like a sink in the sense that once the FSM reaches such a state, it tends to remain there indefinitely. These are the *hard-to-exit* states.

Example 3: In *Fig.2.a* the state S_4 has one outgoing transition and four incoming transitions. Thus, S_4 is a *hard-to-exit* state. □

III. The $FiF - FoF$ Analysis

The criterion of designing sequential machines with improved BIST quality is to make all state codes appear at the PS lines with equal probability during testing. To achieve this should ensure that during testing a) The *unreachable* state codes appear at the PS lines, and b) The *hard-to-reach* (*hard-to-exit*) states are easy to reach (exit).

A. $FiF - FoF$: A New BIST metric

The FiF (FoF) value of an FSM state denotes the ease with which it can be reached (left).

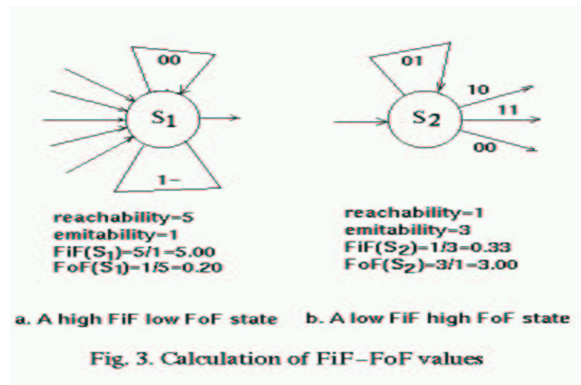
Definition 1: For each state S, *reachability(S)* is the number of edges incident on S from outside.

Definition 2: For each state S of an FSM, *emitability(S)* is the number of edges exit from S.

Definition 3: For each state S of an FSM, *Fan-in-Factor* of S, denoted as $FiF(S)$, is the ratio of *reachability(S)* and *emitability(S)*. Therefore, $FiF(S) = reachability(S)/emitability(S)$.

Definition 4: For each state S of an FSM, *Fan-out-Factor* of S, denoted as $FoF(S)$, is the ratio of *emitability(S)* and *reachability(S)*. Therefore, $FoF(S) = emitability(S)/reachability(S)$.

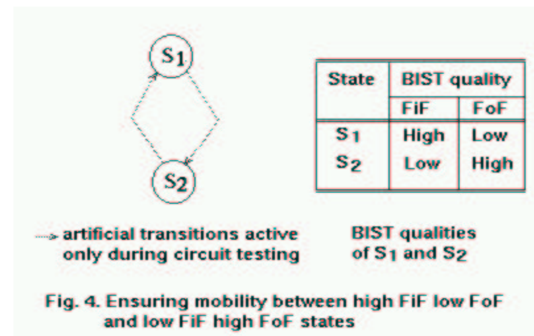
Example 4: In *Fig.3*, the number of incoming edges on S_1 (*Fig.3.a*) and S_2 (*Fig.3.b*) are 5 and 1 respectively. hence, *reachability*(S_1)=5, and



a. A high FiF low FoF state

b. A low FiF high FoF state

Fig. 3. Calculation of FiF−FoF values



State	BIST quality	
	FiF	FoF
S_1	High	Low
S_2	Low	High

Fig. 4. Ensuring mobility between high FiF low FoF and low FiF high FoF states

reachability(S_2)=1. Moreover, emitability(S_1)=1. Thus, $FiF(S_1) = 5/1 = 5$. Similarly, $FoF(S_1) = 1/5 = 0.20$. For S_2 the FiF and FoF values are 0.33 and 3 respectively (*Fig.3.b*). □

IV. The $FiF - FoF$ Based BIST Scheme

The proposed $FiF - FoF$ based BIST scheme is presented below.

A. Handling the *unreachable* states

To ensure mobility between unreachable and reachable, the reachable state codes are assigned from 0 to $n-1$. This way, all unreachable states lie between n to 2^k, so that, for every unreachable state there is a reachable state with MSB as 0 [5]. To input an unreachable state to CC, we propose to add a control primary input (CPI) at the MSB of PS lines. The CPI is implemented with an XOR as shown in *Fig.5*. One input to XOR is the MSB of PS lines and the other one is the CPI. Since the reachable states do appear on the PS lines, unreachable state codes can be fed to CC by keeping CPI as 1. In test mode, the CPI is fed from the PRPG, whereas, for normal functioning the CPI is kept 0.

B. $FiF - FoF$ Based State Encoding

The key idea is to establish a pair of transitions between two states, one with high FiF & low FoF values and the other with low FiF & high FoF values (*Fig.4*).

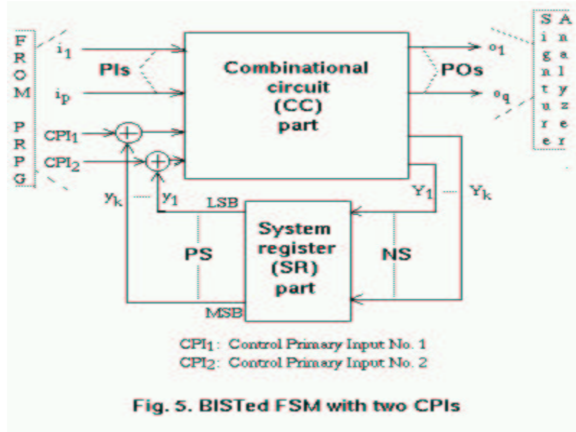


Fig. 5. BISTed FSM with two CPIs

There is no direct transition between S_1 and S_2 in the original FSM. The artificial transitions (shown in dotted lines) are active only during testing. These transitions are achieved by directly manipulating the outputs of SR, without any primary input pattern.

Consider two states (S_{hi}, S_{ho}), $S_{hi} \in H_{FiF}$ and $S_{ho} \in H_{FoF}$. To ensure an easy mobility between S_{hi} & S_{ho} pair, the codes C_{hi} and C_{ho} are assigned to S_{hi} and S_{ho} respectively, such that C_{hi} and C_{ho} differ only in a single bit (say, the LSB). If another control point with XOR is added at the LSB of the PS line, then the FSM may switch between S_{hi} and S_{ho}. This makes the pair, as a whole, quite easy to reach and exit.

The above technique can not properly handle the states with $FiF(.) = FoF(.) = 1$. For such states, $reachability(.) = emitability(.)$, so that it is not possible to distinguish among such states with respect to their BIST qualities. For such states, instead of their $FiF - FoF$ values, we take $reachability(.) + emitability(.)$ as the BIST mesure. In the proposed scheme, states with $|FiF(.) - FoF(.)| \leq \epsilon$, where ϵ is a predefined small positive value, say 0.2, are considered for such treatment. *Fig.5* shows the resulting FSM.

Thus, by adding two control points, at MSB and the LSB, we can ensure mobility between 2^{k-2} pairs of states. Here is the complete algorithm.

Algorithm 1: Encode_with_FiF-FoF

Input: Description of sequential machine as a STT or STG.
Output: A state encoding with enhanced BIST quality

Step 1: Find the number of states n and k, $2^{k-1} < n \leq 2^k$

Step 2: Compute $FiF - FoF$ values for all the states.

Step 3: Consider the states with $|FiF(.) - FoF(.)| \leq \epsilon$. Compute their BIST qualities as $reachability(.) + emitability(.)$. Make a pair (S_i, S_j) such that S_i has the highest BIST quality and S_j has the lowest BIST quality. Repeat the process for all states in the same category.

Step 4: Create two lists of the states $S_1, \cdots, S_x$. One in ascending order of FiF and another in ascending order of FoF values with the states left after Step 3. Let the sorted lists be $S_1^{hi}, S_2^{hi}, ..., S_x^{hi}$ and $S_1^{ho}, S_2^{ho}, ..., S_x^{ho}$.

Step 5: Make the BIST-pairs (S_i^{hi}, S_i^{ho}) until all states are exhausted.

Step 5: Let CS is the code set containing codes $\{0, 1, \cdots, (n-1)\}$. Decide the bit position p (say, LSB) where second control point is to be inserted

Step 6: Take a pair of code C_1 & C_2 from {CS}, where C_1 and C_2 differ only in the pth bit position.
Assign codes C_1 & C_2 to a BIST-pair of states.
Repeat the process for all BIST-pairs.

Example 5: Consider the FSM in *Fig.2.a.* The list of states with descending order of FiF values is $\{S_4, S_2, S_1, \{S_3, S_5\}\}$. That with descending order of FoF values is $\{\{S_3, S_5\}, S_1, S_2, S_4\}$. States with same FiF or FoF values are enclosed in braces. The set P_{BIST} of BIST-pairs is given by $P_{BIST} = \{(S_4, S_3), (S_2, S_5)\}$. The pool of codes C={000, ..., 111}. The unreachable states are 101, 110 and 111. If the second control point is inserted at p=LSB of the PS lines, then, a state assignment according to the present scheme is given by: (000) $\rightarrow S_4$, (001) $\rightarrow S_3$, (010) $\rightarrow S_2$, (011) $\rightarrow S_5$, (100) $\rightarrow S_1$. □

V. Experimental Results

The proposed scheme has been carried out in the framework of SIS [7]. Extensive experimentation was done on a large number of MCNC benchmark circuits samples of which are reported here. *Table I* notes the parameters of the MCNC benchmark circuits.

The results of the proposed scheme are reported in *Table II*. *Column 2* shows the number of test vectors applied. An FSM, synthesized with the proposed scheme, is tested with a fixed number of test vectors generated by the cellular automata (CA) [6] based PRPG that used to test the original circuit. The single s-a fault coverage of the original circuit (synthesized from SIS [7]) is shown in *column 3 and 4*. The results of *column 5* denote the fault coverage of the circuits synthesized with the proposed scheme with two/one control point(s). *Column 6 and 7* of *Table II* depicts the area of the circuit synthesized through JEDI and from proposed scheme with two/one control points (XORs).

The results shown in the *Table I & II* are obtained through a set of similar logic optimization steps (available in SIS) for all the test cases. For fault simulation the synthesized FSM is first converted to 'blif' format and then to bench using public domain tool *blif2bench*. All circuits are subjected to test after prior state minimization with *stamina* of SIS.

It is seen from *Table II* that the proposed scheme

TABLE II

Experimental results for MCNC benchmarks

Circuit name	No. of test vectors	Fault cov(%) with PRPG			area	
		JEDI		Proposed scheme	JEDI (FSM+ 2/1 XOR)	Proposed scheme
		n cell TPG	(n+2/1) cell TPG			
ex1 (+)	1000	68.22	76.59	91.62	345	354
s208	1200	40.53	56.83	82.98	184	129
s420	1000	50.51	61.62	76.83	157	147
s832	8000	41.88	46.05	90.28	450	442
s1494	1200	42.89	51.54	95.09	892	901
styr (+)	2000	42.78	64.51	85.43	740	836
opus	700	57.78	63.89	85.86	153	156
kirkman (*)	7000	78.75	82.32	88.83	262	270.5
scf	10000	38.84	39.00	93.74	1072	1113

(*) one XOR used since no unreachable state exists for these circuits
(+) Flip-flops initialized to all 0s

TABLE I

MCNC Circuit Descriptions

Circuit name	# PI	# PO	# states	#FFs	Optimum area
ex1	9	19	20	5	345
s208	11	2	18	5	184
s420	19	2	18	5	157
s832	18	19	25	5	450
s1494	8	19	48	6	892
styr	9	10	30	5	740
opus	5	16	10	4	153
kirkman	12	6	16	4	262
scf	27	56	121	7	1072

improves fault coverage by a significant amount for all the circuits. The design targets only fault efficiency but gate area. However, experimental results show little area overhead.

For all cases simulation is done in the platform of *hope* [8], and the fault coverage is expressed as

$$faultcoverage = \frac{\text{Total no. of detected faults}}{\text{Total no. of faults in the CUT}}$$

VI. Conclusion

A technique, based on a novel BIST-characterising metric $FiF - FoF$, is proposed in this paper to design sequential machines with enhanced BIST quality. The design improves parameter values for a set of state pairs to improve fault coverage in a BIST environment. Experimental results show that the proposed method improves the BIST quality of the synthesized machines with occasional, marginal, area overhead.

References

[1] V. D. Agrawal, C. R. Kime and K. K. Saluja *'A tutorial on built-in-self-test part 1: principles'*, IEEE Design and Test of Computers, pp. 73-82, March 1993.

[2] V. D. Agrawal, C. R. Kime and K. K. Saluja *'A tutorial on built-in-self-test part 2: applications'*, IEEE Design and Test of Computers, pp. 69-77, June 1993.

[3] I. Pomeranz and S. M. Reddy *'Built-in-test generation for synchronous sequential circuits,*, in Proc. of ICCAD, pp.421-426, 1997.

[4] I. Pomeranz and S. M. Reddy *'Improved Built-in test pattern generators based on comparison units for synchronous sequential circuits'*, in Proc. of ICCAD, pp. 26-31, 1998.

[5] S. Roy, B. K Sikdar, M. Mukherjee and D. K Das, *'Degree-of-Freedom Analysis of Sequential Machines Targeting BIST quality and Gate Area'*, Proc. of (ASPDAC/VLSID-2002), pp.671-676, 2002.

[6] P. Pal Chaudhuri, D. R. Chowdhury, S. Nandi and S. Chatterjee *'Additive Cellular Automata - Theory and Application'*, Vol.1, IEEE Computer Society Press, California, USA, 1997.

[7] *SIS: Asystem for sequential circuit synthesis*, Univ. of California, Berkley, Rep. M92/41, 1992.

[8] H. K. Lee and D. S. Ha *'HOPE: An efficient Parallel fault Simulator for Synchronous Sequential Circuits'*, IEEE TCAD of Integrated Circuits and Systems, Vol. 15, No. 9, pp. 1048-1058, 1996.