A Cost-Effective Scan Architecture for Scan Testing with Non-Scan Test Power and Test Application Cost

Dong Xiang Shan Gu Jia-Guang Sun
School of Software
Tsinghua University
Beijing, 100084
China

Yu-liang Wu
Dept. of Comp. Sci. and Eng.
The Chinese Univ. of Hong Kong
Shatin N.T., Hong Kong
China

Abstract

A new scan architecture is proposed for full scan designed circuits. Scan flip-flops are grouped together if they do not have any common successors. This technique produces no new redundant faults. Scan flip-flops in the same group have the same values in all test vectors. All scan flip-flop groups form a scan forest, where each primary input drives the root of one scan tree. Test application time and test power based on the proposed scan forest architecture can be reduced drastically while pin overhead and delay overhead should be the same as that of conventional scan design. It is shown that test application cost and test power with the proposed scan forest architecture can be reduced to the level of non-scan design circuits.

Categories and Subject Descriptors: Reliability and Testing
General Terms: Algorithms and Reliability

1 Introduction

The complexity of sequential ATPG is prohibitive for large or highly sequential circuits. Scan design [9] make test generation of the circuit be that of a combinational one, which may make the test application cost prohibitively high.

Supported in part by the national science foundation of China under grant 69773030, the 985 fundamental foundation of national education ministry, Hong Kong RGC grant CUHK 4236/01, and NSFC/RGC CUHK 405/02.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
DAC 2003, June 2-6, 2003, Anaheim, Califoria, USA.
Copyright 2003 ACM 1-58113-9/03/0006 ... $5.00.

Narayanan, Gupta and Breuer [4] proposed an optimal $k-$scan chain configuration using dynamic programming, which can get an optimal solution in polynomial time and require a number of extra pins. Lee *et al.* [3] presented a novel test application scheme by supporting multiple circuits with a single scan-in signal, which can reduce test application time drastically without any degradation on test efficiency. The Illinois scan architecture [2] drives multiple scan segments by a single scan-in. Nicolici *et al.* [5] proposed a multiple scan architecture that can greatly reduce test power of scan testing. However, the proposed method cannot reduce test application time effectively. Chandra and Chakravarty [1] introduced a method to reduce test application cost and test data volume via test data compression based on Golomb encoding scheme. Sinanoglu and Orailoglu [6] introduced a new scan architecture to reduce test power and test application time greatly by using a single scan-in pin, demultiplexer, multiplexer and XOR trees. However, all of the methods cannot reduce test application and test power to non-scan DFT level, which needs only one test cycle for each test.

2 Scan Forest: A New Scan Architecture

A new scan architecture called scan forest is proposed to reduce test application cost and test power. The general scan forest architecture is presented in Fig. 1, which contains a number of scan trees. The root of each scan tree is driven by a primary input. In each scan tree, a group of scan flip-flops are controlled by the scan-in, where only one of the scan flip-flops has successors. All successors of the scan flip-flop forms another group of scan flip-flops as shown in Fig. 1, and so on. All leave nodes in the scan tree are connected with a number of exclusive-or trees. Outputs of all exclusive-or trees are connected with a multiple input signature analyzer, which has only one output. The $\times$-bounding problem should be handled for MISR-based methods. Two schemes are utilized to deal with the prob-

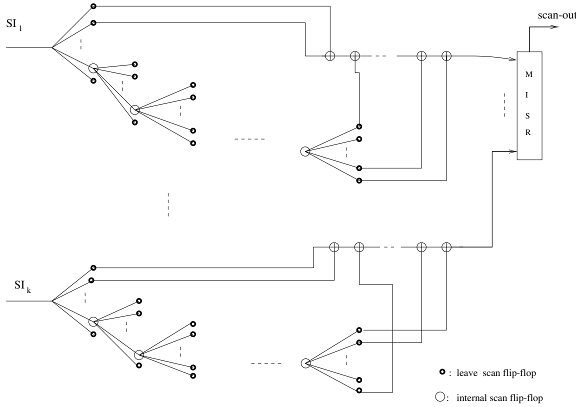

Figure 1: General structure of scan forest.

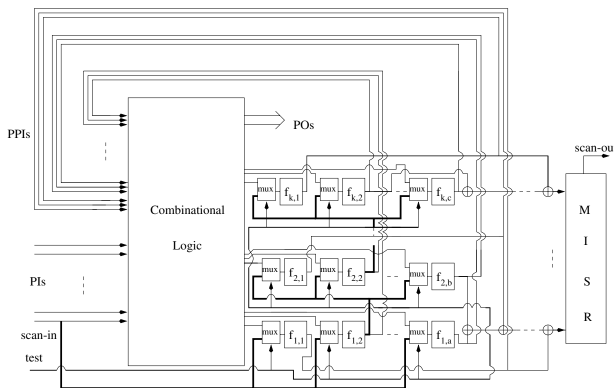

Figure 2: Scan designed circuit with the scan tree architecture.

lem. The first is to insert an AND gate into each stage of the MISR, where all AND gates are controlled by a common extra pin. The second scheme is not to use MISR, which uses multiple extra outputs. It is found that only several outputs can obtain the same fault coverage for almost all circuits.

Fig. 2 presents a circuit with a scan tree. The scan-in is connected with a primary input, which drives a group of scan flip-flops ($f_{1,1}, f_{1,2}, \ldots, f_{1,a}$). Scan flip-flops ($f_{2,1}, f_{2,2}, \ldots, f_{2,b}$) are driven by the output of flip-flop $f_{1,2}$. All scan flip-flops ($f_{k,1}, f_{k,2}, \ldots, f_{k,c}$) are driven by a predecessor scan flip-flop. It is unnecessary for outputs of all scan flip-flops to be connected with the same exclusive-or tree. All flip-flops are inserted into a multiplexer. All multiplexers are controlled by the same test pin as shown in Fig. 2. All outputs of the scan flip-flops feed back to the combinational logic as shown in Fig. 2. It is unnecessary to connect all internal scan flip-flops with the exclusive-or trees.

3 Scan Forest Construction for Full Scan Design

We would like to introduce the techniques in this paper to construct the scan forest. Here we propose a greedy algorithm is proposed to deal with this. The reason why we group scan flip-flops and combine them is that combination of scan flip-flops together can reduce test application cost and test power greatly. As for a fully scanned circuit, the circuit can have many primary outputs. It is quite possible to group many scan flip-flops together, which causes no or trivial degradation on fault coverage.

Initially, the remaining scan flip-flop list includes all scan flip-flops. First, the algorithm selects a scan flip-flop s from the list, and picks up any remaining scan flip-flop v to check whether it converges with s (both flip-flops v and s have at least one common successor) or not. If it does not converge with s, just put v into the same group with s. Continue the above process until all scan flip-flops in the remaining scan flip-flop list have been checked, which forms a group of scan flip-flops. Select another scan flip-flop in the remaining scan flip-flop list and do the same as stated above until the remaining scan flip-flop list is empty.

A scan flip-flop group is selected to connect with the first primary input. That is, all scan flip-flops in the same group can be driven by the primary input. Another group of scan flip-flops is connected with the second primary input. Continue the above process until all primary inputs are connected with a group of scan flip-flops or the scan flip-flop group list is empty. Select a group of scan flip-flops to connect all scan-ins of the scan flip-flops with one of the scan flip-flops in the same group which were connected to the first primary input. That is, all scan flip-flops in the current group are driven by the output of that scan flip-flop. Continue the above process until the scan flip-flop group list becomes empty. It should be noted that the level difference between any two scan trees is only one. Fig. 1 presents the general structure of a scan forest, where $SI_1, SI_2, \ldots, SI_k$ are primary inputs.

The testing scheme of the fully scanned circuit with scan forest is presented as follows: First of all, set the circuit into test mode and scan in values of the tests with respect to leave scan flip-flops into the scan forest, where the number of different values is at most the number of scan trees. The values of the scan flip-flops corresponding to the scan flip-flop groups in the last but one level are scanned in all scan trees. Simultaneously, the values of the PPIs of the leave scan flip-flops are scanned to the next level. Continue the above process until all values of the test vectors are scanned in. Set the circuit into formal mode, all flip-flops capture responses from the combinational logics when all values of the PPIs and values of PIs in the test are applied. After the test responses have been received at primary outputs and scan flip-flops, the next test vector is scanned in as stated above. Continue the above process until all test vectors have been applied.

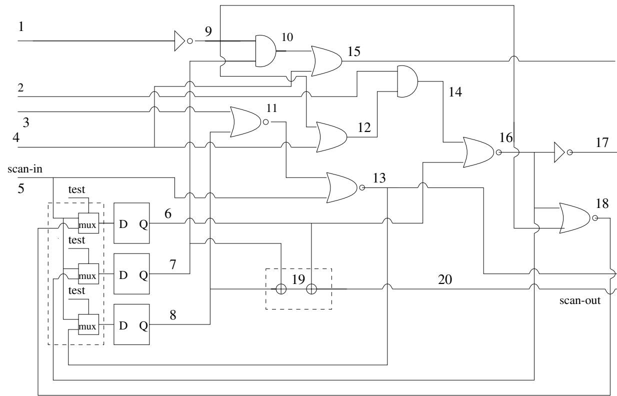

Figure 3: Example of the scan forest architecture.

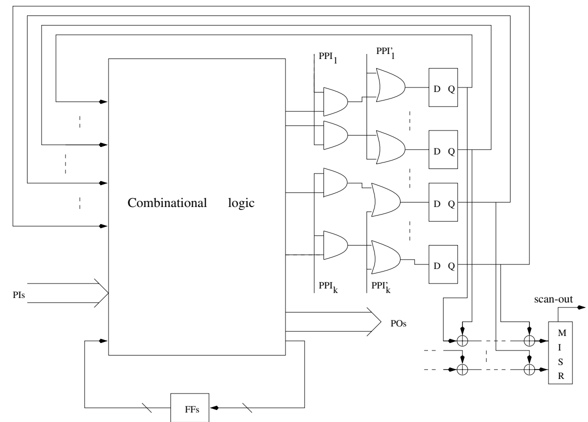

Figure 4: Test circuit of the scanned circuit with the scan forest architecture.

test-application-for-complete-scan()

1. Switch the scan forest to parallel shift mode by shifting the generated tests into the forests and apply the test with respect to the primary inputs.
2. Turn to the normal mode for all flip-flops to capture fault effects and propagated into the combinational logic simultaneously.
3. Switch the scan forest to parallel shift state and shift into initial state of the test pattern while shifting out the captured fault effect to the leave scan flip-flops.
4. Apply the values of the primary inputs with respect to the test and the circuit turns to the normal mode for all flip-flops to capture fault effects and propagated into the combinational logic.
5. If all tests have been applied, shift out all fault effects from the scan forest, end the procedure. Otherwise, go to step 3.

Let *#vectors* and *levels* be the number of test vectors and the number of levels of the scan forest. The test application cost TA for a fully scanned circuit with a scan forest is,

$$TA = \#\text{vectors} \cdot (\text{levels} + 1) + \text{levels}. \quad (1)$$

When the depth of the scan forest is limited to be small enough, test application cost based on the proposed scan forest architecture can be reduced to the level of non-scan DFT, which can generate the same fault coverage as full scan and combinational test generation.

Fig. 3 presents an example circuit of the scan forest architecture. The circuit contains 3 scan flip-flops. Every two of them do not have any common successor in the combinational part of the circuit. Therefore, all three scan flip-flops are driven by the same scan-in signal. Outputs of the scan flip-flops are connected with an exclusive-or tree as shown in Fig. 3. Any test vector of the fully scanned circuit can be applied with two test cycles. Four test cycles are required for each test if they are connected as a scan chain.

The HITEC test generator is utilized to evaluate the scan design method. We have to adopt the two-gate test point structure to replace the scan flip-flop, where both gates have a 1-control and a 0-control test points in the test circuit, respectively as shown in Fig. 4. It is clear that outputs of all scan flip-flops in Fig. 4 can be controlled to values 1 and 0 easily. Outputs of the scan flip-flops are connected with multiple exclusive-or trees, where outputs of the exclusive-or trees are connected with a multiple input signature analyzer. All scan flip-flops in the same group have the common predecessor. That is, all the flip-flops are assigned the same values in all test vectors. It should be noted that the following two factors may produce some new untestable faults. First, faults on PPI_1, ..., PPI_k as shown in Fig. 4 may be hard-to-detect faults because the original inputs of the scan flip-flops may be hard-to-control. Second, the exclusive-or trees may also make some new aliased faults. The negative effects of the exclusive-or trees can be trivial when the number of exclusive-or trees is set properly.

4 Experimental Results

The proposed method called *scanforest* has been implemented and compared with conventional full scan design on test power and test application cost. Node transition count (NTC) [5] is reported as quantitative measure for power dissipation in the paper.

The output of each flip-flop corresponds to two inputs as shown in Fig. 4. All test vectors are modified before applying. Each pair of inputs are combined together to form a new test vector. HITEC is adopted to generate tests

circuit	full scan				no xor-tree		scanforest							
	#FF	FC(%)	#HF	vec	FC(%)	#HF	FC(%)	#HF	vec	TA (%)	TP(%)	po	ao(%)	aliased
s1423	77	99.08	14	178	99.31	14	99.31	15	348	12.99	12.90	6	16.86	1
s5378	179	99.13	40	590	99.32	40	99.36	40	959	2.71	1.88	2	13.60	0
s9234	211	93.47	452	1087	94.67	452	94.89	452	2075	3.60	2.11	2	8.83	0
s13207	638	98.46	151	1146	98.94	152	99.02	153	1659	1.32	0.72	2	15.90	1
s15850	534	96.67	390	1090	97.43	393	97.58	395	2099	1.08	0.93	2	12.22	2
s38417	1636	99.44	176	2712	99.59	180	98.75	581	2717	0.31	0.22	2	15.20	401
s38584	1426	95.85	1506	1718	96.66	1508	96.86	1509	3745	0.76	0.70	2	13.62	1

Figure 5: Performance of the scan forest.

circuits	scanforest		no spare register[6]		with spare register[6]		multiple chain [5]		Golomb code [1]	
	TA(%)	TP(%)	TA(%)	TP(%)	TA(%)	TP(%)	TA(%)	TP(%)	TA(%)	TP(%)
s1423	12.99	12.90	93.06	29.59	30.57	22.99	100	29.5	-	-
s5378	2.71	1.88	82.29	34.02	22.07	12.49	100	29.75	-	21.98
s9234	3.60	2.11	86.73	23.78	17.67	11.08	100	24.67	-	23.70
s13207	1.32	0.72	47.18	19.41	7.75	7.59	100	24.33	-	6.32
s15850	1.08	0.93	57.55	15.17	9.18	5.62	100	17.37	-	14.73
s38417	0.31	0.22	63.26	24.91	6.51	3.29	100	45.03	-	18.65
s38584	0.76	0.70	58.21	22.75	5.24	2.97	100	20.40	-	16.48

Figure 6: Comparison with previous methods.

for the fully scanned circuit with a scan forest, where the XOR trees are not included. PROOFs is then utilized to do fault simulation on the circuit as shown in Fig. 4 based on the generated tests because PROOFs can handle multiplexers and exclusive-or gates. Fig. 5 shows comparison of the *scanforest* with the conventional full scan design on all iscas89 and iscas93 benchmark circuits. In Figs. 5 and 6, parameters FC, #HF, vec, TA, TP, po, ao and *aliased* represent fault coverage, the number of hard faults including aborted faults and redundant faults, the number of test vectors, test application cost reduction ratio, test power reduction ratio of the scan forest architecture, pin overhead, area overhead, and aliased faults, respectively. Columns with respect to *no xor-tree* present results of the full scan design results after the control logic as shown in Fig. 4 have been inserted. All the above results are obtained by HITEC. Test vectors are modified by combining each pair of inputs as shown in Fig. 4. The columns with respect to *scanforest* present results of the scan forest architecture after the exclusive-or trees have been inserted (based on PROOFs). It is found that most of the new hard faults based on the scan forest architecture are faults of the control logic and XOR trees. Depth of the scan forest for each benchmark circuit used is no more than 4, that is, at most 4 test cycles are required to apply a test for all fully scanned circuits.

Fig. 6 presents comparison of the method with three recent methods on test application cost and average test power reduction ratios. The multiple scan chain [5] can effectively reduce test power, but not test application cost. However, both [1] and [6] consider only scan chain transitions, but [5] and the proposed method also consider test power in the combinational logic. The test power and test application cost reduction ratios should still be convincing although different test generators are used by the methods.

5 Conclusions

A new scan architecture called scan forest was proposed to reduce test application cost and test power completely. Scan flip-flops were grouped if they do not generate new reconvergent fanouts. A new reconvergent fanout may cause new redundant faults. Test application cost and test power based on the proposed scan forest architecture can be reduced to the level of non-scan DFT while fault coverage is the same as full scan for almost all benchmarck circuits. The advantage of the proposed method becomes clearer when the size of the circuit or the number of flip-flops in the circuit increases.

References

[1] A. Chandra and K. Chakrabarty, "Combining low-power scan testing and test data compression for systems-on-a-chip," Proc. of *ACM/IEEE Design Automation Conf.*, pp. 166-169, 2001.

[2] I. Hamzaoglu and J. H. Patel, "Reducing test application time for full scan embedded cores," Proc. of *IEEE Int. Symp. on Fault-Tolerant Computing*, pp. 260-267, 1999.

[3] K. J. Lee, J. J. Chen, and C. H. Huang, "Using a single input to support multiple scan chains," Proc. of *IEEE/ACM Int. Conf. on Computer-Aided Design*, pp. 74-78, 1998.

[4] S. Narayanan, R. Gupta, and M. A. Breuer, "Optimal configuring of multiple scan chains," *IEEE Trans. Computers*, vol. 42, no. 9, pp. 1211-1131, 1993.

[5] N. Nicolici and B. M. AI-Hashimi, "Multiple scan chains for power minimization during testing application in sequential circuits," *IEEE Trans. on Computers*, vol. 51, no. 6, pp. 721-734, 2002.

[6] O. Sinanoglu and A. Orailoglu, "A novel scan architecture for power-efficient, rapid test," Proc. of *IEEE/ACM Int. Conf. on Computer-Aided Design*, Nov., 2002.

[7] H. C. Tsai, K. T. Cheng, and S. Bhawmik, "On improving test quality of scan-based BIST," *IEEE Trans. Computer-Aided Design of ICAS*, pp. 928-938, no. 8, 2000.

[8] S. Wang and S. Gupta, "ATPG for heat dissipation minimization during scan testing," Proc. of *ACM/IEEE Design Automation Conference*, pp. 614-619, 1997.

[9] M. J. Y. Williams and J. B. Angell, "Enhancing testability of large-scale integrated circuits via test points and additional logic," *IEEE Trans. Computers*, vol. 22, pp. 46-60, no. 1, 1973.