PASTA: Partial Scan to Enhance Test Compaction +

Irith Pomeranz and Sudhakar M. Reddy
Electrical and Computer Engineering Department
University of Iowa
Iowa City, IA 52242

Abstract

We propose a procedure to select flip-flops for partial scan targeting the reduction of test length. We show that significant reductions in test length can be achieved by this procedure. In addition, experimental results show that using heuristics that target the test length does not have to increase the numbers of flip-flops that need to be scanned in order to achieve a given level of fault coverage. Consequently, it may be possible to perform partial scan selection targeting the two parameters, test length and fault coverage, without requiring more flip-flops than required for one of the parameters.

1. Introduction

We investigate a procedure for partial scan selection based on test sequence length. Test compaction techniques were proposed before to reduce the length of test sequences generated by test generation procedures, without reducing the fault coverage [1]-[7]. Test length reduction is important in reducing tester memory requirements and test application time. Here, we show that it is possible to drive a procedure for partial scan selection such that it aids a test compaction procedure in reducing the test length for the partial scan circuit. At the same time, the target fault coverage, which is the primary goal of partial scan selection, is also achieved. In particular, we design a procedure for partial scan selection to be applied in conjunction with the static test compaction procedure of [7].

The selection of partial scan flip-flops in the proposed procedure is based on a given compacted test sequence. Partial scan selection is done so as to change the time units where faults are detected by the compacted test sequence. For example, if a fault f is the only one detected by the last test vector of the test sequence, and if f is propagated to flip-flop i by an earlier vector, then inserting flip-flop i into the scan chain allows f to be detected by the earlier vector, and the last vector of the test sequence can be omitted. As a result, the test sequence length can be reduced. We show that by inserting flip-flops into a scan chain so as to reduce the detection times of selected faults, the static test compaction procedure of [7] can continue and reduce the test sequence length, achieving two-fold reductions in test length and even higher compared to a *compacted* test sequence for the original circuit.

We investigate a procedure for partial scan selection based on the test length alone. However, our experimental results indicate that for a given number of scanned flip-flops, the fault coverage achieved by the proposed procedure (targeting the test length) is comparable to that achievable by partial scan selection procedures that target the fault coverage directly [8], [9]. This indicates that, without increasing the number of selected flip-flops, it is possible to consider both the fault coverage and the test length simultaneously. In this way, a more effective partial scan selection can be made than if the two parameters are targeted one following the other.

2. An example

We demonstrate the ability to enhance the effectiveness of static test compaction by partial scan through the following example.

We consider ISCAS-89 benchmark circuit $s27$. The initial test sequence is shown in Table 1. This sequence was obtained by compacting a random sequence of length 100. Compaction was done using the static compaction procedure of [7]. In Table 1, u is a time unit, $T[u]$ is the input vector applied at time unit u to the circuit inputs (1234), and $S[u]$ is the present state at time unit u, defined over present state variables (567). Next, we insert the flip-flop corresponding to state variable 6 into the scan chain. This changes present state variable 6 into a primary input, and its corresponding next state variable into a primary output. We refer to the new primary input as primary input 6. To ensure that the fault coverage of the sequence T shown in Table 1 does not go down, we assign to primary input 6 the values assigned to state variable 6 under

+ Research supported in part by NSF Grant No. MIP-9357581, and by NSF Grant No. MIP-9725053.

0-7695-0104-4/99 $10.00 © 1999 IEEE

T. The resulting sequence is shown in Table 2. Whenever state variable 6 is x in Table 1, we assign to primary input 6 the value 0 in Table 2. Applying the procedure of [7] to compact the sequence of Table 2, we obtain the sequence shown in Table 3. In this case, scanning a single flip-flop allows us to reduce the test length by three vectors, from 14 to 11.

Table 1: Initial

u	$T[u]$ 1234	$S[u]$ 567
0	0011	xxx
1	1101	$0x0$
2	0011	101
3	1011	000
4	0001	010
5	0011	010
6	0110	010
7	1000	010
8	0110	100
9	0101	000
10	1011	001
11	1010	100
12	0001	100
13	0001	000

Table 2: Scan 6

u	$T[u]$ 12346	$S[u]$ 57
0	00110	22
1	11010	00
2	00110	11
3	10110	00
4	00011	00
5	00111	00
6	01101	00
7	10001	00
8	01100	10
9	01010	00
10	10110	01
11	10100	10
12	00010	10
13	00010	00

Table 3: Compaction

u	$T[u]$ 12346	$S[u]$ 57
0	00110	22
1	10110	00
2	00011	00
3	01101	00
4	10001	00
5	01100	10
6	01010	00
7	10110	01
8	10100	10
9	00010	10
10	00010	00

3. The selection procedure

In this section, we describe a procedure for selecting flip-flops to be inserted into a scan chain so as to allow the static test compaction procedure of [7] to reduce the test length of an already compacted sequence. We first describe the overall procedure, and then describe the heuristic used to select the flip-flops. We denote the present state variable corresponding to flip-flop i by y_i, and the next state variable corresponding to flip-flop i by Y_i. We point out that the procedure of [7] produces highly compacted sequences compared to other static compaction procedures.

The proposed procedure accepts a test sequence T that has already been compacted by the procedure of [7] as much as possible. The scan selection procedure selects one by one the flip-flops that may allow the length of T to be reduced. When flip-flop i is selected, the flip-flop is inserted into the scan chain, and the test sequence T is modified to reflect this change. The change is made as illustrated in Section 2, by copying the value of state variable y_i into the new primary input corresponding to flip-flop i, and replacing x values on y_i by 0s. The static test compaction procedure is then called to compact T. If the length of T is reduced by static compaction, flip-flop i remains in the scan chain. Otherwise, flip-flop i is taken out of the scan chain, and the test sequence before the modification is restored. This is repeated as long as we can select additional flip-flops to be inserted into the scan chain.

Next, we describe the selection of the flip-flop to be inserted into the scan chain at every iteration. Our goal is to identify a flip-flop that can potentially allow the length of T to be reduced after its insertion into the scan chain. The following heuristic is used for this purpose. Let fault f be detected by T at time unit $u_{det}(f)$. Suppose that f is propagated to a next state variable Y_i at time unit $u_{sv}(f)$, where $u_{sv}(f) < u_{det}(f)$. By inserting flip-flop i corresponding to Y_i into the scan chain, it is possible to reduce the detection time of f from $u_{det}(f)$ to $u_{sv}(f)$. This may allow us to omit test vectors between time units $u_{sv}(f)$ and $u_{det}(f)$ of T, if these test vectors are only necessary in order to detect f. For example, we consider $s27$ under the test sequence shown in Table 1. The fault f_{10} is detected by this sequence at time unit 13. The results of simulating the fault free circuit and the faulty circuit in the presence of f_{10} are shown in Table 4. Here, $f.free$ stands for the fault free circuit, and $z[u]$ is the primary output sequence. Comparing the fault free and faulty states, it can be seen that by scanning state variable 7, f_{10} can be detected at time unit 5. Thus, we have $u_{det}(f_{10}) = 13$ and $u_{sv}(f_{10}) = 5$. By inserting the flip-flop corresponding to state variable 7 into the scan chain, test vectors previously required to detect f_{10} may now be omitted.

Table 4: Simulation of $s27$ in the presence of f_{10}

u	$T[u]$ 1234	$S[u]$ 567 f.free	$S[u]$ 567 faulty	$z[u]$ f.free	$z[u]$ faulty
0	0011	xxx	xxx	x	x
1	1101	$0x0$	$0x0$	1	1
2	0011	101	101	1	1
3	1011	000	000	0	0
4	0001	010	010	0	0
5	0011	010	011	0	0
6	0110	010	010	0	0
7	1000	010	010	1	1
8	0110	100	101	1	1
9	0101	000	000	1	1
10	1011	001	001	1	1
11	1010	100	100	1	1
12	0001	100	100	1	1
13	0001	000	001	0	1

The heuristic above is based on the selection of a fault for which the detection time $u_{det}(f)$ may be reduced by inserting a flip-flop into the scan chain. It is possible to consider the faults in increasing or decreasing order of $u_{det}(f)$. Our experiments indicate that increasing order of detection times yields the best overall results, and we use it in our experiments.

In the procedure described above, the same flip-flop may be selected more than once. This is needed if the following scenario occurs. In iteration j_1, flip-flop i_1 is selected. However, it is determined that scanning flip-flop i_1 does not help the static compaction procedure reduce the test length, and flip-flop i_1 is removed from the scan chain. At this point, we set $tried[i_1] = 1$ and we do not allow flip-flop i_1 to be tried again. In iteration j_2 where $j_2 > j_1$, flip-flop i_2 is selected and left in the scan chain because it helps reduce the test length. The next time a flip-flop needs to be selected, it may be possible to use i_1 to reduce the test length. Therefore, we set $tried[i_1] = 0$ after flip-flop i_2 is selected, and we allow flip-flop i_1 to be selected again. The procedure terminates when no additional flip-flop can improve the test length, i.e., when $tried[i] = 1$ for every flip-flop i which is not in the scan chain.

4. Experimental results

The results of the proposed procedure are reported next. As initial test sequences, we use the test sequences produced by the test generation procedure of [10], and compacted by the procedure of [7]. We use the test sequences of [10] for static compaction since they achieve high fault coverages. However, we expect the results to be relatively independent of the test generation procedure, since the test sequence is compacted before it is considered by the proposed procedure. Note that since the initial test sequence is compacted, further reductions in test length are due to partial scan selection.

The results are shown in Table 5. After circuit name, we show the number of flip-flops in the circuit. Next, we show the length and the number of detected faults for the initial compacted sequence, before applying the proposed procedure to it. Under column *modified seq* we show the results of the proposed procedure at an intermediate stage (subcolumn *interm*), and after it terminates (subcolumn *final*). For each stage, we show the number of scanned flip-flops, the compacted test length, and the number of detected faults. The intermediate stage was selected such that approximately half the number of flip-flops included in the scan chain at the end of the procedure are included in it at the intermediate stage. Additional intermediate results are reported below. It can be seen that two-fold reductions and more in test length can be achieved by selecting the partial scan flip-flops appropriately.

Next, we report on the results of applying test generation to several circuits obtained during the application of the proposed scan selection procedure. The purpose of this experiment is to check the effectiveness of the scan selection heuristic based on static compaction in achieving increased fault coverage. The test generation procedure we use is *MIX*, described in [11]. The results are compared to the results of the partial scan selection procedures from [8] and [9].

Table 5: Experimental results of test length reduction

				modified seq					
		init seq		interm			final		
circuit	ff	len	det	ff	len	det	ff	len	det
s298	14	117	265	3	87	268	7	36	274
s344	15	57	329	-	-	-	1	46	331
s382	21	516	364	7	88	383	15	39	394
s400	21	611	380	6	138	387	12	44	393
s444	21	608	424	5	267	435	11	101	440
s526	21	1006	454	7	539	481	15	59	495
s641	19	101	404	5	73	407	10	56	408
s820	5	491	814	2	241	830	4	165	830
s1196	18	238	1239	7	201	1239	15	145	1242
s1423	74	1024	1414	27	567	1448	54	274	1477
s1488	6	455	1444	2	290	1464	5	138	1475
s5378	179	646	3639	20	359	3674	40	202	3730

The results of this experiment are reported in Table 6. For every circuit considered, each row in Table 6 corresponds to a different number of scanned flip-flops. The number of flip-flops inserted into the scan chain is shown under column *ff*. For each circuit and each number of scanned flip-flops considered, we show the total number of faults, the number of detected faults, the number of faults proved undetectable, the number of faults aborted, and the fault coverage of *MIX*. In addition, we report the test length, and the run time of *MIX* in seconds on an HP C180. When the number of flip-flops inserted into the scan chain equals that reported in [8] or [9], we show the fault coverage from [8] or [9] for comparison.

From Table 6, it can be seen that scan selection based on the effectiveness of static test compaction helps increase the effectiveness of the test generation procedure *MIX* as well. Comparing to the results from [8] and [9], one must note that different test generation procedures were used in [8] and [9]. Nevertheless, we observe that the fault coverage reported for the circuits with the scan flip-flops selected by the procedure proposed here is higher in several cases. We believe that this indicates that the proposed selection procedure is at least competitive with the ones from [8] and [9], and the proposed heuristics based on test length can be incorporated into a partial scan selection procedure based on fault coverage in order to improve both the fault coverage and the test length simultaneously.

In the experiment above, we first obtained a complete, compacted test sequence for the circuit, and then used the proposed procedure for partial scan insertion. However, it is not necessary to apply the procedure to a complete test sequence. Instead, it is possible to apply it

Table 6: Experimental results of test generation

circuit	ff	tot	det	und	abr	f.c	len	time	f.c [8]	f.c [9]
s298	1	308	308	0	0	100.0	139	0.6	94.8	
s298	10	-	-	-	-	-	-	-		99.4
s344	1	342	335	5	2	98.0	138	4.5	96.2	
s344	5	-	-	-	-	-	-	-		98.8
s382	2	399	388	5	6	97.2	801	35.0	97.0	
s382	3	399	388	9	2	97.2	378	8.8	97.2	
s382	6	399	395	4	0	99.0	158	1.9		99.3
s382	15	399	399	0	0	100.0	123	0.4		
s400	2	424	395	20	9	93.2	508	96.0	95.5	
s400	3	424	407	15	2	96.0	375	6.7	95.8	
s400	5	424	410	14	0	96.7	184	8.6		97.6
s400	12	424	417	7	0	98.3	124	1.8		
s444	2	474	449	17	8	94.7	785	70.0	94.5	
s444	6	474	454	19	1	95.8	160	3.3		96.0
s444	11	474	460	14	0	97.0	100	4.7		
s526	2	555	513	14	28	92.4	1145	231.0	91.4	
s526	5	555	533	21	1	96.0	533	9.2	98.7	
s526	15	555	554	1	0	99.8	341	1.1		98.7
s641	1	467	408	59	0	87.3	188	2.0	95.7	
s641	7	467	408	59	0	87.3	79	1.1		99.4
s820	0	850	815	35	0	95.9	823	20.0	94.2	
s820	1	850	841	9	0	98.9	653	22.0	98.9	
s820	2	850	850	0	0	100.0	764	17.1		100.0
s1423	7	1515	1463	17	35	96.6	2203	5579	50.3	
s1423	34	1515	1492	14	9	98.5	681	367	97.6	
s1423	41	1515	1499	14	2	98.9	563	126		97.9
s1423	54	1515	1500	14	1	99.0	219	9.3		
s1488	0	1486	1446	40	0	97.3	676	19	94.4	
s1488	1	1486	1466	20	0	98.7	402	11	99.1	
s1488	2	1486	1486	0	0	100.0	320	8.7		99.9
s5378	17	4603	4003	585	15	87.0	1664	617	88.0	
s5378	27	4603	4291	312	0	93.2	1799	170	93.8	
s5378	40	4603	4321	282	0	93.9	1257	54		
s5378	48	-	-	-	-	-	-	-		97.1

using a partial test sequence generated by imposing an upper bound on run time for test generation, or even using a random sequence. As flip-flops are inserted into the scan chain, a more complete test sequence can be obtained, until the target fault coverage is achieved.

6. Concluding remarks

Test compaction procedures alone may not be sufficient in reducing the test length for a given circuit in applications where the test sequence length is critical (e.g., if the tester memory is limited). We showed that after a static test compaction procedure achieves the maximum level of compaction it can achieve for a given circuit, inserting selected flip-flops into a scan chain can allow the procedure to continue and reduce the test length for the circuit. We proposed a procedure for selecting flip-flops to be scanned so as to increase the effectiveness of static test compaction, and demonstrated its performance. We also investigated the performance of the resulting flip-flop selection heuristic as a heuristic for increasing the fault coverage that can be achieved for the circuit. The results indicated that it is possible to perform partial scan selection targeting the two parameters, test length and fault coverage, without requiring more flip-flops than required for one of the parameters.

References

[1] R. K. Roy, T. M. Niermann, J. H. Patel, J. A. Abraham and R. A. Saleh, "Compaction of ATPG-Generated Test Sequences for Sequential Circuits", in Proc. Intl. Conf. on Computer-Aided Design, Nov. 1988, pp. 382-385.

[2] I. Pomeranz and S. M. Reddy, "On Generating Compact Test Sequences for Synchronous Sequential Circuits" in Proc. EURO-DAC '95, Sept. 1995, pp. 105-110.

[3] S. T. Chakradhar and A. Raghunathan, "Bottleneck Removal Algorithm for Dynamic Compaction and Test Cycles Reduction", in Proc. EURO-DAC '95, Sept. 1995, pp. 98-104.

[4] E. M. Rudnick and J. H. Patel, "Simulation-based Techniques for Dynamic Test Sequence Compaction", in Proc. Intl. Conf. on Computer-Aided Design, Nov. 1996.

[5] F. Corno, P. Prinetto, M. Rebaudengo and M. Sonza Reorda, "New Static Compaction Techniques of Test Sequences for Sequential Circuits", in Proc. 1997 Europ. Design & Test Conf., March 1997, pp. 37-43.

[6] M. S. Hsiao, E. M. Rudnick and J. H. Patel, "Fast Algorithms for Static Compaction of Sequential Circuit Test Vectors", in Proc. VLSI Test Symp., April 1997, pp. 188-195.

[7] I. Pomeranz and S. M. Reddy, "Vector Restoration Based Static Compaction of Test Sequences for Synchronous Sequential Circuits", in Proc. Intl. Conf. on Computer Design, Oct. 1997, pp. 360-365.

[8] I. Park, D. S. Ha and G. Sim, "A New Method for Partial Scan Design Based on Propagation and Justification Requirements of Faults", in Proc. 1995 Intl. Test Conf., Oct. 1995, pp. 413-422.

[9] G. S. Saund, M. H. Hsiao and J. H. Patel, "Partial Scan Beyond Cycle Cutting", in Proc. 27th Fault-Tolerant Computing Symp., June 1997, pp. 320-328.

[10] M. S. Hsiao, E. M. Rudnick, and J. H. Patel, "Sequential Circuit Test Generation Using Dynamic State Traversal", in Proc. 1996 Europ. Design & Test Conf., March 1996, pp. 22-28.

[11] X. Lin, I. Pomeranz and S. M. Reddy, "MIX : A Test Generation System for Synchronous Sequential Circuits", in Proc. 1998 VLSI Design Conf., Jan. 1998, pp. 456-463.