Deterministic and Low Power BIST Based on Scan Slice Overlapping

Ji Li Yinhe Han Xiaowei Li
Institute of Computing Technology, CAS, Beijing, 100080, China
Graduate School of the CAS, Beijing, 100039, China
Leeji1212@ict.ac.cn

Abstract

This paper presents a new deterministic pattern generation structure that can be used in conjunction with any LFSR reseeding scheme. The proposed scheme utilizes Scan slices overlapping to reduce the number of specified bits and the number of transitions at the same time. Thus, it can significantly reduce test power and even further reduce test storage. A decoder is used to generate control signals. Experimental results indicate that the proposed method significantly reduces the switching activity by 80% and only needs relatively small test data storage.

1. Introduction

When the process of very large-scale integrated circuits (VLSI) scales down into deep sub-micron, the complexity of circuit designs has greatly increased. Testing of a chip has posed some new challenges, especially with the emergence of System-on-chip (SOC). Traditional test methods based on the automatic test equipment (ATEs) are becoming unacceptable. Built-in self-test(BIST) offers an attractive solution to these challenges.

In recent years, research on deterministic BIST has become a hot topic, which can provide very high test coverage and relatively short test time. The architecture of deterministic BIST mainly consists of three parts: test data storage, stimulus decoder, and response compactor. Many kinds of decoders have been proposed to reduce test data storage, such as LFSR[1-3], decoder of encoding compression[4-8], XOR network[9] and the other customized counters[10]. Among these decoder schemes, LFSR reseeding[1-3] generate deterministic test cubes by expanding seeds, providing significant reduction of test data storage and bandwidth. Therefore LFSR reseeding technique has been studied extensively, and supported by many commercial tools including TestKompress by Mentor Graphics[11] and SmartBIST by Cadence[12].

While LFSR reseeding is powerful for test data compression, it is not so good for power consumption. BIST implementation randomly fills don't care bits in the test cubes, which can result in excessive switching activities when the filled test patterns are shifted into the scan chains, which may give rise to severe hazards to the circuit reliability. One method has been proposed in [13] for reducing test power for LFSR reseeding, which considers together the problems of test data compression and low power test. In this scheme, two LFSRs are used. The main LFSR generates the test cube through conventional reseeding. An extra "masking" LFSR is used to generate a set of mask bits. Test power is reduced because the output of the two LFSRs are ANDed or ORed, thus reducing the transition probability. However, the test data storage for this scheme is greatly increased compared with conventional LFSR reseeding because it requires storing an extra set of seeds for the extra "masking" LFSR.

A novel parallel CWD (pCWD) approach is presented in [14] for lowering test power by shortening wrapper scan chains and adjusting test patterns. In [14], a two-phase process on test pattern: "partition" and "fill", is presented. Our method proposed in this paper expands Han's work, and applies relative idea to BIST. A key feature of the proposed approach is that it reduces the number of specified bits and the number of transitions at the same time. Thus, the total test storage and power dissipation can be reduced, since the amount of compression for LFSR reseeding depends on the number of specified bits and excessive transitions likely result in excessive power dissipation.

2. Scan Slices Overlapping

There are many don't care bits in test cubes. Conventionally, in BIST environment, the don't care bits are filled with random values, which result in excessive switching activity when they are shifted into a scan chain. Observing the X-bit distribution in test cubes can help us to find a method to handle this problem. Mintest patterns are examined. We get an observations from the result: The density of X-bit in circuit under test is high. To S13207, the X-bit occupies up to 93.2% of total test bits. "High density" of X-bit can make two consecutive scan slices in test patterns have a high probability to be overlapping. *Scan slice* is defined to be the set of inputs applied to the scan chain inputs at a scan cycle. *Scan slices overlapping* means that scan slices are same. See the example in Figure 1, the pattern contains 10 scan slices. Each slice consists of 4 bits. The first and second slice in the original pattern are:{{1 X X 1},{X X 1 1}}. If the second and third bits in the first slice and the first and second bits in the second slice are assigned to 1, they will be {{1 1 1 1}, {1 1 1 1}} and overlapping.

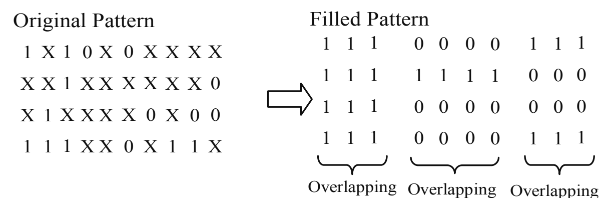

Figure 1. Slices overlapping in test patterns

3. Test Pattern Generation

Rather than using LFSR reseeding to directly encode the specified bits as in conventional LFSR reseeding, we can divide the test cube into blocks and slices of a block overlap each other. We call this kind of block as *overlapping block*. Then, for each test cube, when calculating the seed of this cube by solving equations, we can only take the first slice in every block into account.

* This paper is supported by the National Natural Science Foundation of China(NSFC) under grant No.90207002 and 60242001, key technique project of Beijing (No. H020120120130) and Basic Research Foundation of Institute of Computing Technology, Chinese Academy of Sciences(No. 20036160).

0-7803-8834-8/05/$20.00 ©2005 IEEE.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on February 10, 2009 at 03:13 from IEEE Xplore. Restrictions apply.

That is, when the LFSR expands a seed into a test pattern to randomly fill the scan chains, only the value of the first slice in every block is generated from the LFSR, and the value of the rest slices can be got by directly shifting the value of the first slices of blocks. After the scan chains have been filled, the test pattern is applied to the circuit under test and the response is shifted out to MISR. The process is then repeated to generate the next test pattern.

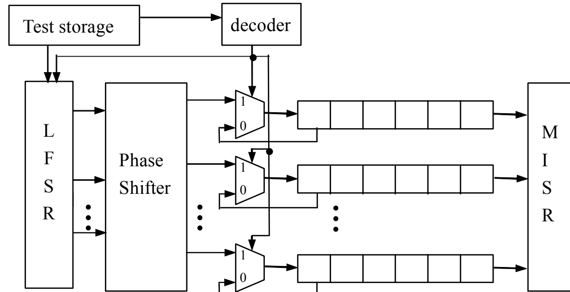

Figure 2. Hardware implementation

The hardware implementation for the proposed scheme is shown in Figure 2. Compared to the STUMPS structure, additional hardware overhead consists of one 2-to-1 MUX per scan chain and one decoder. The decoder is used to generate the 0 or 1 control signals to control the behavior of MUX. When the first slice of overlapping block is shifted into scan chains, the value of control signal is 1, so that the scan chains are loaded from the LFSR. And when the rest slices of overlapping block are shifted into the scan chains, the value of control signal is 0, so that the last value shifted into the scan chains is repeatedly shifted into the scan chains and the data from the LFSR is ignored. We use the control signals to gate the clock of the LFSR so that the power of the LFSR and the phase shifter is reduced at the same time, which leads to the reduction of the total power during the test.

4. Pattern partition

According to the proposed architecture, the pattern is needed to be partition into some overlapping slice sets. Obviously, in order to utilize the overlapping of consecutive scan slices to reduce the specified bits and transitions furthest, we should minimize the number of blocks. The partitions which can lead to the minimal number of blocks are not unique, and different partitions schemes may result in different results. A simple example is shown. See Figure 3, the original pattern contains 7 scan slices. Because of incompatibility between the third scan slice and the fifth scan slice, the original pattern should be divided into two overlapping blocks and the incompatible two scan slices should be contained in these two blocks respectively. Note that the forth scan slice is compatible with the two incompatible scan slice at the same time, which results in two different partition schemes as shown in Figure 3. We can see, after combination of compatible scan slices, total number of specified bits in the first partition scheme is 7, more than that in the second partition scheme which is reduced to 6.

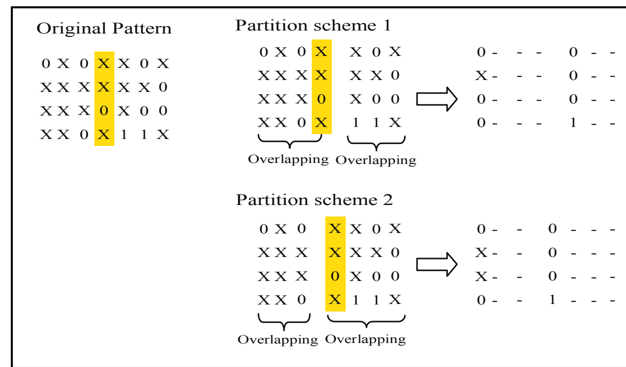

Figure 3. An illustration of different partition schemes

It's possible to find the best partition schemes by simulating all possible partition situations. However, this is time consuming work. In this paper, we use a simple heuristic method to get partition of overlapping blocks. The algorithm is described as Algorithm 1.

```
while( partition of scan slices is not finished )
    Judge if the current scan slice is compatible with frontal
    compatible sets.
    If (compatible)
        Jump to 1
    Else
        An overlapping block has been found, and all don't
        care bits in this block are specified.
  1: continue the next scan slice
endwhile
```

Algorithm 1. pattern partition algorithm

An example of the pattern partition procedure is shown in Figure 4. Figure 4-(1) shows the original pattern. In Figure 4-(2), the original pattern is divided into overlapping blocks by the algorithm 1. In Figure 4-(3), in every overlapping block, the specified bits are combined into the first slice and the assignments of all slices are equal. Thus, the LFSR only need to generate the specified bits in the first slice of every block after partition and combination, and don't care (what the symbol '-'stand for) the rest slices.

This approach reduces the number of transitions in the scan chains and also reduces the total number of specified bits that must be generated by the LFSR as compared with conventional LFSR reseeding. It can be seen that the total number of specified bits in the original pattern is 16, and the total number is reduced to 10 after partition in this example.

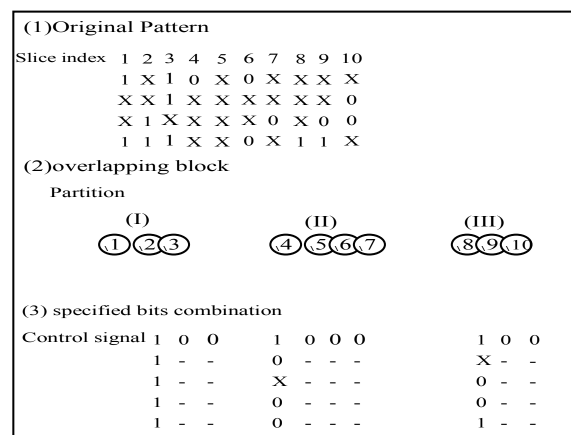

Figure 4. Pattern partition.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on February 10, 2009 at 03:13 from IEEE Xplore. Restrictions apply.

Besides the reduction of number of specified bits, the total number of transitions is also greatly reduced in this way. Now we make a simple analysis about reduction of transitions. Because the don't care bits in each test cube get filled with random values by the LFSR, the transitions may happen to every neighbor scan cell in original pattern. When the pattern is divided into overlapping blocks, the transitions only happen between blocks. Obviously, the total number of transitions can be reduced after pattern partition.

It must be noticed that the control signals become the one part of total test storage. Therefore in order to reduce test storage, the control signals should be compressed. This can be done using fixed run-length encoding.

5. Experimental results

To validate the efficiency of the proposed method, experiments were performed on the largest ISCAS 89 benchmark circuits. Test sets used in this experiment were obtained using Mintest dynamic compaction [15].

Table 1 presents the results. Each test cube is divided into overlapping blocks, and the number of blocks is shown in each case. The number of specified bits and the number of transitions required for the proposed scheme is compared with the original test cubes. The number of transitions is calculated as described in [16]. Column NP, NC and NB respectively show number of patterns, the number of scan chains and the corresponding number of overlapping blocks. Column NCB shows the total number of control signals before compression. Column T_D shows the size of the original test sets. The total number of specified bits after our method is applied to the test cube is shown in column partNSB. The last column of storage gives the percentage of compression. Finally, Column testing Power lists the original number of transitions(origNT), the number of transitions after dividing and filling(partNT), and the reduction of transitions(Pred%). As can be seen, our scheme yields great reduction in the total number of transitions, which will result in a significant reduction of power dissipation. And preferable test data compression is also obtained. It should be noted there is a tradeoff between the storage compression and the power reduction. When the number of scan chains is added, the test data compression increases, but the power reduction decreases. Figure 5 shows the change of test data compression and the power reduction with the number of scan chains.

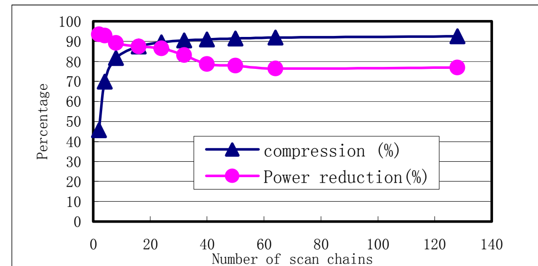

Figure5. Circuit s13207: Comp% vs. Pred%

Table 2 provides a test data compression comparison between the proposed scheme and the best previously published results[5-8], whose compression ratios are all relative to the size of the dynamic compacted test set from MinTest. Column T_D gives the size of precomputed test set. With the exception of one case when proposed scheme has lower compression rates than VIHC (s35932), the table clearly shows that the proposed method leads to better compression ratios than previous approachs.

Finally a comparison of the previous approaches [4,13] and the proposed scheme is given in Table3. To the each scheme, the first column shows the required storage and the second column shows the power reduction. Note that the power consumption resulting from the test pattern generation is not included in the power computing list in the table(the same to [4] and [13]), since this part power consumption is relatively very small to the power

Table1.Experimental Result for Proposed Scheme

Circuit				Data Storage				Test Power		
Name	NP	NC	NB	NCB	T_D	partNSB	Comp%	origNT	partNT	Pred%
s9234	160	4	2785	9920	39273	7970	54.45	587162	188445	67.91
		8	1881	4960		9102	64.20	292826	117485	59.88
		16	1250	2560		9713	68.75	150211	68102	54.66
		32	773	1280		10097	71.04	70027	40687	41.90
s13207	236	4	3053	41300	165200	8549	69.83	7174481	513681	92.84
		8	2266	20768		9518	81.67	3614877	393556	89.11
		16	1557	10384		10289	87.49	1786081	224177	87.45
		32	1064	5192		10645	90.42	869403	147262	83.06
s15850	126	4	3272	19278	76986	9199	63.02	2896184	505570	82.54
		8	2409	9702		10606	73.63	1447180	336952	76.72
		16	1597	4914		11603	78.55	736383	212328	71.17
		32	1007	2520		12092	81.02	378865	124083	67.25
s38417	100	4	260	41600	164736	25256	59.42	79814	20621	74.16
		8	9145	20800		31847	68.05	14347971	3580890	75.04
		16	7009	10400		38127	70.55	7249499	2270429	68.68
		32	5032	5200		44014	70.13	3678937	1406398	61.77
s38584	137	4	3319	50142	199104	26474	61.52	1804967	856030	52.57
		8	8798	25071		29882	72.39	17907510	3453106	80.72
		16	6046	12604		32292	77.45	8959395	2105740	76.50
		32	4013	6302		33688	79.92	4499316	1314616	70.78

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on February 10, 2009 at 03:13 from IEEE Xplore. Restrictions apply.

Table 2. Test Data Compression Comparison for Same Test Set

Circuit	T_D	Test Data Compression (%)				
		Golomb [5]	FDR [6]	VIHC [7]	Selective Huffman[8]	Proposed
S5378	23754	40.70	48.19	51.52	55.1	**70.38**
S9234	39273	43.34	44.88	54.84	54.2	**71.04**
S13207	165200	74.78	78.67	83.21	77.0	**90.42**
S15850	76986	47.11	52.87	60.68	66.0	**81.02**
S35932	28208	N/A	10.19	**66.47**	N/A	53.23
S38417	164736	44.12	54.53	54.51	59.0	**70.13**
S38584	199104	47.71	52.85	56.97	64.1	**79.92**

Table 3. Result comparing proposed scheme with previous schemes

Circuit	Dual-LFSR reseeding[13]		Alternating run-length code[4]		Proposed	
	Storage	Pred%	Storage	Pred%	Storage	Pred%
S9234	19440	24.35	21612	76.30	12273	54.66
S13207	11803	25.26	32648	93.68	15837	83.06
S15850	14518	25.14	26306	85.27	16517	71.17
S38417	66234	24.90	64976	81.35	48527	61.77
S38584	23835	24.70	77372	83.52	44896	70.78

resulting from the transitions. When compared to dual-LFSR reseeding proposed in [13], the proposed scheme is much more effective and obtains up to 60% better power reduction (s13207). Because dual-LFSR reseeding scheme uses different test sets, the optimality of the ATPG and compaction procedures used to obtain the test sets strongly affects the results, and 1000 pseudo-random patterns are applied first. Even though all of this, our method needs less storage in s9234 and s38417. When compared to alternating run-length code in [4], the reduction of test power for the proposed scheme is not as much as the method proposed in [4] which is based on run-length encoding and there is still 20% gap, but the storage is much less than the latter.

6. Conclusions

LFSR reseeding is an attractive approach for compressing test data. We have shown that the proposed deterministic BIST scheme provides a way to reduce test power for LFSR reseeding. By utilizing *Scan slices overlapping* combined with LFSR reseeding, it can reduce transitions during test(up to 90%) and specified bits in test cubes. Experimental results indicate that the proposed method significantly reduces test power and even further reduces test storage. The proposed technique can be combined with other techniques, such as partial reseeding and seed encoding, and thus the better results can be obtained.

Reference

[1] B. Koenemann, "LFSR-coded test patterns for scan designs",Proc.of ETC, pp. 237-242, 1991

[2] S.Hellebrand, J.Rajski, S.Tarnick, S.Venkataraman, and B.Courtois, "Built-in test for circuits with scan based on reseeding of multiple-polynomial linear feedback shift registers", IEEE Trans. On Comput., vol.44, pp. 223-233, 1995.

[3] C.V.Krishna, Nur A. Touba, "Reducing Test Data Volume Using LFSR Reseeding with Seed Compression,"Proc. of ITC, pp. 321-330, 2002.

[4] Chandra, A., and K. Chakrabarty, "Reduction of SOC Test Data Volume, Scan Power and Testing Time Using Alternating Run-length Codes," Proc. of DAC, pp. 673-678, 2002.

[5] A. Chandra and K. Chakrabarty, "System-on-a-chip test data compression and decompression architectures based on Golomb codes," IEEE Trans. CAD, vol. 20, pp. 355–368, March 2001.

[6] A. Chandra and K. Chakrabarty, "Frequency-directed run-length (FDR) codes with application to system-on-a-chip test data compression," Proc. of VTS,pp.114-121, Apr.2001.

[7] P. T. Gonciari, et.al, "Improving compression ratio, area overhead, and test application time for system-on-a-chip test data compression /decompression," Proc.of DATE.,pp. 604–611, 2002.

[8] A. Jas., et.al, "An efficient test vector compression scheme using selective Huffman coding," IEEE Trans. CAD, Volume: 22,pp. 797- 806, June 2003.

[9] I. Bayraktaroglu and A. Orailoglu, "Test Volume and Application Time Reduction Through Scan Chain Concealment", Proc. of DAC, pp. 151-155, 2001.

[10] H.-G.Liang, et.al, "Two dimensional test data compression for scan-based deterministic BIST," Proc. ITC., pp. 894–902, 2001.

[11] J. Rajski., et.al, "Embedded deterministic test for low cost manufacturing test" , Proc. of ITC, pp. 301-310, 2002.

[12] B.Koenemann, et.al, "A SmartBIST variant with guaranteed encoding," Proc. Of VTS, pp. 325-330, 2001.

[13] P.M.Rosinger, et.al, "Low Power Mixed-Mode BIST Based on Mask Pattern Generation Using Dual LFSR Re-seeding," Proc. of ICCD, pp. 474-479, 2002.

[14] Y. Han, et.al, "Wrapper scan chains design for rapid and low power testing of embedded cores", Proceedings of the 13th Asian Test Symposium, Pingdong, pp.9-14,Nov.,2004

[15] I. Hamzaoglu and J.H. Patel,"Test set compaction algorithms for combinational circuits,"Proc. of CAD, pp.283-289, Nov. 1998.

[16] R. Sankaralingam, et.al, "Static compaction techniques to control scan vector power dissipation," Proc. of ATS., pp. 35-40, 2000.

Authorized licensed use limited to: INSTITUTE OF COMPUTING TECHNOLOGY CAS. Downloaded on February 10, 2009 at 03:13 from IEEE Xplore. Restrictions apply.