NOVAK, O., HLAWICZKA, A. GARBOLINO, T. GUCZWA, K. PLIVA, Z. NOSEK, J,: Low Hardware Overhead Deterministic Logic BIST with Zero-Aliasing Compactor. Proc. IEEE DDECS conf. Györ (Hungary), Apr. 2001

Low Hardware Overhead Deterministic Logic BIST with Zero-Aliasing Compactor

Ondřej Novák**, Andrzej Hlawiczka*, Tomasz Garbolino*, Krzysztof Gucwa*, Jiří Nosek**, Zdenek Pliva**

* Silesian University of Technology, ul. Akademicka 16, 44-100 Gliwice, Poland
e-mail: {hlawicz, garbol, gucwa }@boss.iele.polsl.gliwice.pl
** Technical University Liberec, Hálkova 6, 461 17 Liberec I, Czech Republic
e-mail: ondrej.novak@vslib.cz

Abstract *In the paper we present a test-per-clock BIST scheme that spares memory for storing test patterns and reduces the number of clock cycles, which are necessary to be performed during testing. It has zero error aliasing space compactor and MISR. The test pattern generator (TPG) scheme is based on a T type flip-flop feedback shift register. The generator can be synchronously set similarly to a D type flip- flop shift register and it generates test patterns in a test per clock mode. The numbers of bits stored in a memory, the numbers of clock cycles, the hardware overhead and the parameters of resulting zero aliasing space compactor and MISR of the ISCAS benchmark circuits are given in the paper. The experiments demonstrate that the BIST scheme provides shorter test sequences than other methods while the hardware overhead and memory requirements are kept low.*

1. INTRODUCTION

The complexity and the restricted module accessibility of complex ICs require sophisticated test methods. Built-in-self-test (BIST) combined with Boundary Scan or other similar serial scan approach can help to tackle the problem at low cost. BIST structures for sequential circuits (SQC) can be divided in two categories [17]: test-per-scan (TPS) and test-per-clock (TPC). In TPS scheme, a large number of test sessions and scan operations are required, what causes either high power consumption during testing [11] or leads to additional hardware overhead and complicated control of test generation. So in this paper we focus our attention on the test-per-clock BIST structure, which needs only one test session and few scan operations. A classical TPC type BIST, given in [18] for example, is composed of a test pattern generator (TPG) in form of LFSR that feeds primary inputs (PIs) of SQC, a response compactor in form of MISR which is connected to SQC primary outputs (POs) and a multifunctional self-test register (e.g. BILBO) which replaces the memory register (MR) of SQC. Direct feedback lines imply that the signatures of test responses in BILBO have to be used as test patterns for SQC state variables, i.e. pseudo-primary inputs (PPIs). The most important drawback of the discussed BIST scheme is the fact that the state diagrams of SQC in normal mode and in test mode are different. The above problem was solved in [19] where the authors used TPC BIST scheme in form of BIST-PST (BIST - Parallel Self-Test), where the BILBO register was replaced by a MISR. In this structure, the MISR operates as MR in normal mode, what requires modifying the excitation function of SQC flip-flops (FFs). The BIST-PST hardware minimisation method using MISR, which contains D and T flip-flops, was proposed in [20]. Essential disadvantage of BIST-PST schemes is a fact that different signatures, which are simultaneously considered to be test patterns, appear at feedback loops of SQC with different probabilities. The mentioned drawback can be eliminated using the BIST-DFF scheme proposed in [19,21] (this scheme was designated as DFF, since in the system mode, the state registers are used as D flip-flops). In this structure, doubling the number of flip-flops breaks the direct feedback path from MR flip-flops to MR flip-flops via the combinational logic. In the testing mode, the MR is reconfigured into a TPG (e.g. LFSR) and the remaining flip-flops serve as compactor (e.g. MISR) of responses appearing at SQC pseudo-primary outputs (PPOs). Therefore, the SQC is treated as a combinational circuit (CC) in testing mode. Unfortunately, doubling the number of FFs leads to high hardware overhead. In [19] this scheme was minimised by using BIST-PAT (BIST with integrated PATtern generator). The TPG (LFSR) in autonomous mode passes through a fixed sequence of states and stimulates SQC PPIs. The LFSR structure is chosen in such a way that the generated sequence covers the maximum number of state transitions of SQC.

The main aim of this paper is to achieve very high fault coverage and zero-aliasing response compaction using TPC BIST structure with low hardware overhead. The TPG has both random and deterministic features, which means that the random test sequence is deflected so that it passes even patterns which exercise the random resistant faults. We decided to use the above-described BIST-DFF scheme instead of cheaper BIST-PAT structure because the latter cannot operate in deterministic mode, as it was previously mentioned. In order to significantly reduce the number of FFs, test responses appearing at SQC POs and PPOs are compacted in a zero-aliasing space compactor with short zero-aliasing MISR.

In [15,16] we studied different TPG structures. We found, that a CA, which is either formed by T flip-flops [15] or contains T flip-flops in addition to D flip-flops [16] and has one global feedback only is advantageous from several points of view. It generates test patterns that have better random properties than, for example, those from the LFSR; the fault coverage of these patterns is substantially higher than for other TPG structures. Particularly, it was shown in [15] that a CA formed by T flip-flops with only one global feedback can generate weighted random patterns with the help of seeding with

a previously determined vector without any structure modification. It can generate pseudoexhaustive test sets for large CUT cones without any reseeding. The hardware overhead of the automaton is very low.

From these reasons we used this CA also in [15] in order to generate test patterns which cover all the stuck at faults. We have found that the sequence of the CA could be easily modified by adding mod 2 a sequence of one-bit binary constants to one of the internal automaton flip-flops so that the automaton generates all the necessary test patterns. We developed an optimisation algorithm, which finds a CA seed and a minimal so called modification sequence, which deflects the CA output sequence in order to form a complete test set. In this paper, we improved the algorithm so that it can handle also large circuits at a reasonable CPU time.

In most cases, the number of SQC POs and PPOs is big, what leads to a high cost of the MISR. The issue of MISR size reduction was discussed in [8], where the use of XOR trees like space compactor was proposed. However, such structure can increase error aliasing, so the zero-aliasing compactor design methodology is required. In [9] the method for designing zero aliasing linear Space Compactor (SC) with *sc* outputs was introduced. This SC was designed for a set of modelled faults. In [10], a method for designing zero aliasing MISR with Internal-Exclusive-OR-Feedback-Path for a set of modelled faults was proposed. This method is based on searching such a characteristic MISR polynomial, which does not divide any of equivalent-error polynomials. The expected bounds of such a polynomial degree was given in [10]. Knowing the expected bounds one can search only a set of polynomials of expected degree, what significantly speeds up the computation. The issue of designing the combined zero-aliasing SC and zero-aliasing MISR was addressed in [14].

As it has been previously mentioned in this paper, we focused our attention on the TPC BIST-DFF method, which is modified in such a way allowing achieving sufficient fault coverage during testing of sequential circuits. During the test session, the sequential circuit is converted to the combinational one. Its PIs and PPIs are fed from a deterministic TPG. The CUT responses appearing at its POs and PPOs are compacted by zero-aliasing SC with short zero-aliasing MISR.

In the paper we for the first time combine our previous results [12,13,14] to design such BIST structure. Our BIST scheme uses scan technique with a simple modification so it can generate effective test patterns every clock cycle. As the number of clock cycles is smaller as compared to other methods the presented methodology can be considered to be low power.

The rest of the paper is organised as follows. Section 2 discusses the proposed BIST scheme and testing scenario. In Section 3, the method of finding a TPG seed and modification sequence is described. Section 4 is focused on experimental results using ISCAS circuits. We conclude in Section 5.

2. PROPOSED BIST SCHEME

We propose a BIST scheme consisting of modified scan chain, which is divided into three parts. The input part forms a CA that generates test patterns. The modification consists in replacing the D-type flip-flops (D-FFs) from the scan chain by T-type flip-flops (T-FFs) [12,13], adding one global feedback that is necessary for test pattern sequence generation [12] and adding other feedback loops which guarantee pseudo scanning [13]. The second part contains the internal D-FFs of the CUT. This part forms a shift register that is fed from the serial output of the CA in the testing mode. This register and the CA generate a new parallel pattern for PIs and PPIs every clock cycle [12]. The CUT responses to the test patterns do not influence the test pattern sequence during the testing phase. The third part of the scan chain forms a zero-aliasing compactor [14].

In order to achieve high operating frequency of the CA, it has to be constructed using special carefully designed T-FFs, the area and propagation delay of which are similar to those of D-FFs coming from the same standard cell library. In our paper an example of dedicated T-FF is presented which has been designed in AMS 0.6 um CMOS technology using JK flip flop. Such dedicated T-FF is faster and occupies less area than the classical two-cell solution consisting of D-FF with Exclusive OR gate [13]. Moreover, its maximal operating frequency is only several % lower and its area is only about 25% larger than those of D-FF [13]. Unfortunately it is difficult to use T-FFs for stimulating the PIs and PPIs because the binary stream that is serially shifted into the m-bit CA does not appear in the same form at its serial output with m clock cycles delay.

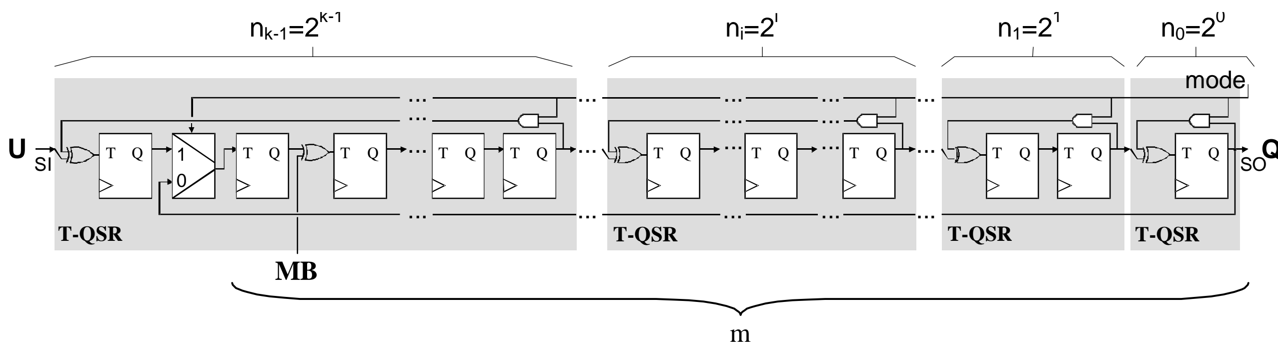

Fig. 1 The m-bit CA (T-FSR) equipped with quasi-scan mode and modification bit input MB.

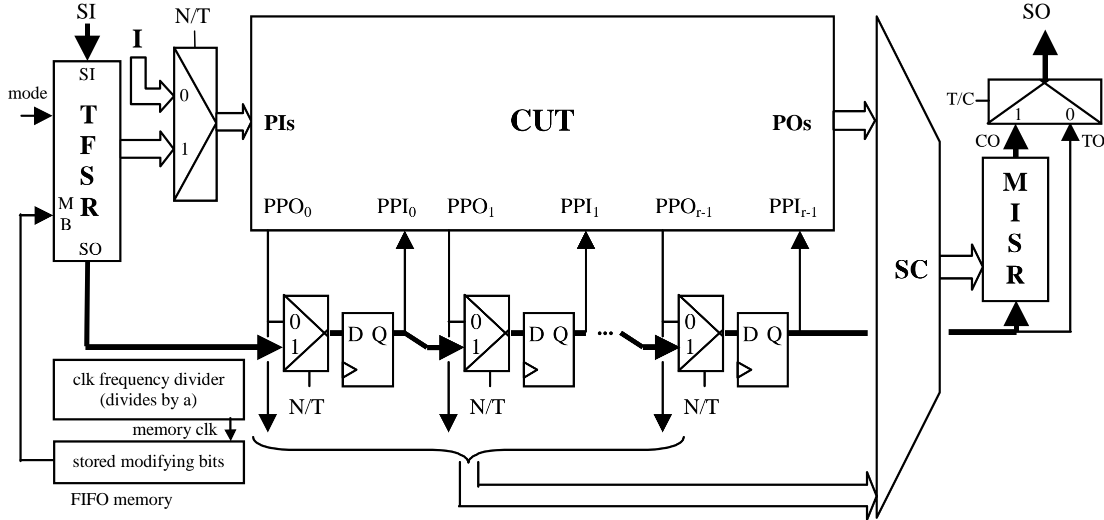

Fig. 2. Test-per-clock BIST scheme. (N/T - normal/test mode, T/C - TPG test/compactor test)

The solution to this disadvantage is given in [13] where the idea of T-type Quasi Shift Register (T-QSR) composed of T-FFs was introduced. T-QSR has the same polynomial $p(x) = x^n$ as an n-bit shift register consisting of $n = 2^i$ D-FFs. The behaviour of T-QSR is similar to that of the shift register. The state diagram of T-QSR has a characteristic of binary tree form similar to tree-like state graph of shift register. One of the advantages of T-QSRs is the possibility of their resetting and setting to an arbitrary state after applying the specific input sequence. Any T-QSR of the length $n = 2^i$ can be reset or set in n clock cycles. Another observation is that any input sequence is repeated at the output of an $n = 2^i$ bit T-QSR with n clock cycles delay, though it is processed in the register. Such operation of T-QSR, which can be considered as a quasi scan mode facilitates integration of CA composed of T-QSRs into a scan-chain.

An overall structure of any CA (T-FSR) composed of described T-QSRs connected in series is given in Fig. 1. The number of m+1 T-FFs in the CA can be represented by the following formula:

$$m+1 = \sum_{i=0}^{k-1} n_i b_i, \quad b_i \in \{0,1\}, \quad n_i = 2^i \tag{1}$$

For the mode = 0 the register works as m-bit CA with the main feedback loop connecting output of the last T-FF with the input of the second T-FF. Quasi scan operation, resetting and setting to an arbitrary state of the T-FSR are possible for the mode = 1. In the quasi-scan mode the register has the length m+1. One T-FF is added to avoid connecting XOR gate and multiplexor in series (see Fig. 1), which would limit maximal operational frequency of T-FSR. Because of the same reasons the modification bit input MB is placed between the second and third T-FF.

In the most common testing strategies the scan path is tested before it is used for applying testing stimuli to the CUT. This task is usually accomplished by a certain test patterns, which are serially shifted through the scan path. We assume that every test pattern generator based on a CA is a part of the scan path. Thus, the serial test pattern provided for the scan path should also test all CAs incorporated in it while they are working in the quasi-scan mode. A universal test that guarantees exhaustive functional test of T-FF and D-FF was found in [13]. So, in our BIST we have used an idea of CA (T-FSR) design equipped with the quasi-scan mode with the following properties:

- the timing properties similar to existing solutions,
- lower area overhead,
- can be easily integrated to the scan path,
- can be easily tested by a single serial pattern.

We propose zero aliasing compactor of CUT responses consisting of zero aliasing linear SC and zero aliasing MISR. Such compactor is henceforth denoted as SC-MISR. It is often possible to design SC having only a few outputs ($4 < sc < 6$). Because of it, it is important to use zero aliasing MISR with the number of flip-flops as small as possible. To design such MISR the method proposed in [10] was used. However the expected bounds in [10] were calculated with a pessimistic assumption that the probability of error on any bit of the output sequence P_{err} is equal to 0.5. In the scan mode we disconnect the feedback loop of the MISR. We have a constant value at the parallel inputs of the MISR during feeding inputs PI and PPI of the CUT by constant zero patterns from TPG. The MISR behaves like a shift register that contains inverters at inputs of some of the D-type flip-flops.

The proposed BIST scheme is given in Fig. 2. It is assumed that all memory elements in this structure are clocked from the same source. It is also assumed that the MISR is the Internal Exclusive-OR type.

Scenario of testing contains the four following phases:

T1. *Testing of TPG* (CA and D-FF chain of length r). At the first phase we shift the patterns of length 3(m+r+1)+5, which functionally test the TPG into the TPG serial input SI.
We capture the responses at the output, which is denoted in Fig. 2 as TO. At the end of this phase the TPG is set to all zero state by the means of the quasi scan mode.

T2.1. *Seeding MISR.* We disconnect the MISR loop and feed all XOR gates in the feedback path with constant value 0. As a result of the previous phase we also have zero value set at all PIs and PPIs of the CUT and at the serial input of the MISR. During the next *sc* clock cycles we feed the serial input of the TPG with zero until the MISR is set to the defined state *V*.

T2.2. *Shifting out the seed V of the MISR to the serial scan output SO through CO*. We merely repeat again the phase T2.1.

T3. *Test of the CUT*. Firstly we connect loops of the MISR, seed the TPG with the first prepared seed (see Section 2) and set the TPG mode to the functional mode. Then we perform the same number of clock cycles, as is the planned number of test patterns. After every *a* clock cycles we feed the MB input with a new modification bit from the serial FIFO memory. Simultaneously we compact the CUT responses in each clock cycle.

T4. *Shifting out the final signature*. Now we apply a zero value at the serial TPG input SI and then we perform *sc* clock cycles. During this period additional patterns are applied to the CUT inputs (PIs and PPIs). The CUT's responses to these additional patterns are also considered in the process of zero aliasing SC-MISRs design. The serial bit stream S' that is received at the output SO represents the final signature S ($S' \neq S$) of the SC-MISR.

3. FINDING A TPG SEED AND THE MODIFICATION SEQUENCE

In order to minimise the number of TPG clock cycles during testing the CUT we have to find an effective TPG seed and a modification sequence. We start finding with generation of deterministic test vectors with the help of some ATPG tool. We use a non-optimised test set where one three state vector exists for each considered fault of the CUT with the bit values 0, 1 and 2 where 2 means *don't care*. We choose the dimension of the CA equal to the number of primary CUT inputs. We set the parameter *a* which tells the system how many test patterns are generated before using another modification bit.

At the beginning, the automaton has to be seeded with some pre-calculated vector. As a measure of the seed quality we introduce a fault coverage gain G. G characterises a seed quality from the point of view of the number of detected faults and the difficulty of covering the faults. In our experiments we choose the seed in such a way that it causes maximum G in the following *a* generated vectors. A given number of test vectors with the biggest number of care bits is chosen from the test vector list and a seed which can cause that the CA passes some of the test vectors within the *a* following clock cycles is found by the means of solving a set of linear equations which describes the CA behaviour. After finding a seed which can contain the don't care bits we prove whether the CA passes the states which correspond to another test vectors. If there are no more vectors which can be obtained without increasing the number of care bits in the seed the algorithm tries to complete this seed with other care bits which cause that some other test vectors are generated within the *a* following clock cycles. If there are more possibilities how to continue with the seed care bits completing, the solution that causes greater G is chosen. The measure of coverage gain increment caused by each of the test vectors is equal to the following formula:

$$\Delta G = -\log_2 (1-(1-1/2^b)^a) \quad (2)$$

where *b* equals to the number of care bits in the reached test vector. After calculating G for all the prospective seeds we choose the seed with the highest G to be a CA seed. The fault coverage of the test vectors which are generated after seeding the CA within the *a* test steps is simulated, all the covered faults are deleted from the fault list and the corresponding test

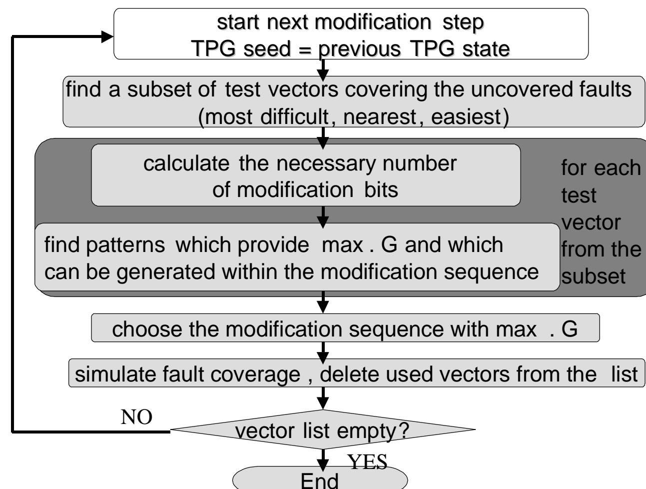

Fig. 3. Finding the modification sequence

vectors are deleted from the test vector list as well.

If the first *a* test steps after seeding the CA do not detect all the faults from the possible fault list, the algorithm continues with finding the modification sequence. The modification sequence causes that with the help of *k* modification bits and *k*a* clock cycles we obtain a CA state sequence which contains some of the remaining test vectors with the maximum G.

The final state, which is reached after applying the best sequence, is considered to be a seed in the next step. The fault simulation is performed; the covered faults and vectors are deleted from the lists. If the fault list is not empty the algorithm repeats finding the modification sequence bits until the fault list is empty.

4. EXPERIMENTAL RESULTS

Some of the experiments we have performed are summarised in Tables 1 and 2. In Table 1 we have compared the TPG properties with [1]. We implemented the TPGs in the 0.8 µm technology. We compared the memory requirements and the area, which is necessary for implementing this memory and a scan chain for deterministic testing of the benchmark circuits with the help of a minimum compact test sequence [1] and the memory requirements and a TPG hardware overhead for the proposed method. In the table we used the following symbols: VECT - minimum number of test vectors [1], MEM – number of stored bits, AREA – hardware overhead of the method given in μm^2. We can see that the extra area is smaller and the number of stored bits is lower for the proposed method than for the method [1]. In the proposed method, we have chosen parameter *a* equal to 1, which means that the number of clock cycles is equal to the number of stored bits. We can say that the number of necessary clock cycles is the lowest within all the known testing approaches.

In Table 2 we compare the proposed BIST overhead with the circuits without any scan and the circuits with a scan chain only. For the experiments demonstrated in Table 2 we have used the technology AMS 0,6 µm in the Cadence environment. In the table we use the following symbols: M – number of CUT PIs, OUT - number of CUT POs, FF – number of CUT flip-flops (PPIs=PPOs), MEM – the memory used for storing the seeds and a modification sequence, CLK - number of clock cycles introduced within testing the circuit, OA - original area (given in μm^2) of the circuit without any diagnostic equipment, SDA - area of the circuit with the boundary scan and internal flip-flop scan loop, BIST - area of the circuit with complete BIST equipment, SC - number of space compactor outputs, MISR number of MISR flip-flops, MIShR – the

TABLE 1.
Comparison between the proposed method and the one presented in [1].

	[1]			proposed method	
circuit	VECT	MEM	AREA	MEM	AREA
C1355	84	3444	466126	1116	396637
c1908	106	3498	439192	1250	369678
c3540	84	4200	525422	821	420532
c7552	73	15038	1472777	8274	1254448
s820	93	2139	341141	719	300066
s832	94	2162	341957	717	299995
s1196	113	3616	390303	1004	306913
s5378	97	20758	1212728	6765	727221
s9234	105	25935	1380853	10992	860667
s13207	233	163100	6919248	20532	1882697
s15850	95	58045	2819351	51587	2599428

number of Multi Input Shift Register flip-flops, CPU - approx. time of a PC with Intel Pentium III, 850 MHz, which was necessary for finding the modification sequence given in seconds.

In columns MEM and CLK we give only the numbers of memory bits and clock cycles, respectively, used for the testing phase T3 (CUT testing) in order to enable the reader to compare the results with other possible methods of the CUT testing.

In some cases the SC has more outputs than the number of flip-flops required for zero aliasing MISR. This leads to a solution which consists of Multi-Input Shift Register (MIShR) with no feedback, which output is connected to MISR input (see Fig. 4).

From the table we can notice that the hardware overhead of the complete BIST is similar to the overhead of the circuit featuring the scanning only. We have chosen the parameter *a* so that the number of clock cycles was less than several tens of thousands. This number is dramatically lower than that of [3,6] while the hardware overhead is similar.

TABLE 2.
Properties of the proposed BIST structure.

circuit	M	OUT	FF	MEM	CLK	OA	SDA	BIST	SC	MISHR	MISR	CPU
c1355	41	32	0	0	1471	245453	390915	390381	4	-	8	300
c1908	33	25	0	18	9810	373304	488877	497099	6	-	10	480
c2670	157	140	0	2496	9988	571618	1163432	1094931	4	-	10	72000
c3540	50	22	0	15	9285	806882	950352	969801	5	-	11	300
c5315	178	123	0	0	1931	1479598	2079371	2040608	4	-	11	720
c7552	206	108	0	4552	9104	1811735	2222219	2440284	14	2	12	72000
s820	18	19	5	5	9995	180088	253814	261476	4	-	8	60
s832	18	19	5	11	9240	181904	255630	264577	4	-	9	60
s953	16	23	29	31	9345	275755	353467	377989	6	-	8	120
s1196	14	14	18	56	8848	316276	372070	401741	5	-	9	120
s1423	17	5	74	53	9222	398485	442322	533131	11	1	10	240
s5378	35	49	179	589	9424	1634613	1801992	1957734	18	7	11	5400
s9234	19	22	228	10992	10992	3158174	3307619	3500428	9	-	11	302400
s13207	31	121	669	20532	20532	6329141	6755566	7149147	26	14	12	691200
s15850	77	150	534	51587	51587	6718756	7170941	7546048	23	11	12	950400

The experiments prove that using zero aliasing SC significantly reduces the number D-FFs in zero aliasing MISR. For example for s15850 the number of D-FFs is reduced from 684 (534+150) to 23. Simultaneously, the maximum number of the zero-aliasing MISR stages is 12 only.

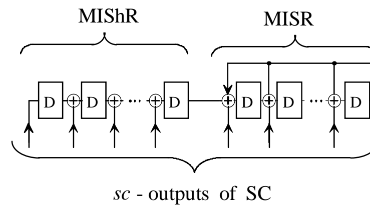

Fig.4. Structure of compactor in the case when *sc* is bigger then the required number of flip-flops in zero aliasing MISR

5. CONCLUSION

In the paper a new test-per-clock BIST scheme is introduced. During the test session the sequential CUT is converted to the combinational one. Its primary inputs (PIs) and pseudo-primary inputs (PPIs) are fed from a deterministic TPG. The CUT responses appearing at the CUT primary- and pseudo-primary outputs (POs and PPOs) are compacted by a zero-aliasing compactor. A CA that is equipped with a quasi-scan mode forms the main part of the TPG. Such a CA has the following properties: it can be easily integrated into the scan path and it can be tested by a single serial pattern. The area overhead of this solution is significantly lower than the cost of the solution with D flip-flops and local feedback taps with XORs on each of them. Simultaneously, the both solutions have similar timing properties.

The CA sequence can be easily modified with the help of XORing constants stored in the memory with some of internal CA flip-flops. This modification sequence forces the automaton to pass the states, which correspond to previously generated test vectors. The algorithm for finding the seed and optimising the modification sequence was developed. It is effective enough that it can handle even large circuits.

The resulting test sequences are very effective from the point of view of the number of stored bits and the number of clock cycles that are necessary for introducing the test sequence to the CUT inputs. A trade off must exist between the memory requirements and the number of clock cycles. If we want to generate a test sequence, which spares the power consumed during testing, we can change the test parameter so that the transition from one test pattern to another is very fast and thus the number of clock cycles for testing is low.

Using the SC leads to a significant reduction of the number of MISR's D-FFs. In cases when at-speed testing is necessary one can decrease the number of XOR gate levels in zero-aliasing SC by increasing the SC output number and, in consequence, the number of D-FFs in zero-aliasing MISR. It is a question of the trade-off between the speed of the whole compactor and the area overhead of the MISR. It can be found by changing the value sc between its maximum (number of POs and PPOs) and its minimum (values indicated in the columns "MIShR" and "MISR" of Table 2). When boundary scan

path (BSP) is connected to POs of CUT then zero aliasing SC can be connected only to PPOs and BSP is combined directly with zero aliasing MISR.

The hardware overhead of the proposed scheme is low; it is comparable with that of the scheme with a simple scan design.

ACKNOWLEDGEMENT

The research was in part supported by the research grants from the Czech Grant Agency GACR 102/01/0566 and from the Polish Committee for Scientific Research (KBN).

REFERENCES

[1] Hamazaoglu, I. - Patel, J.H.: Test Set Compaction Algorithms for Combinational Circuits. Proc. of International Conf. on Computer-Aided Design, November 1998

[2] Koenemann, B.: LFSR – coded test patterns for scan designs. Proc. Europ. Test Conf., Munich , Germany, 1991, pp. 237-242

[3] Hellebrand, S. - Rajski, J.- Tarnick, S.- Venkatarman, S. - Courtois, B.: Built-In Test for Circuits with Scan Based on Reseeding of Multiple-Polynomial Linear Feedback Shift Registers. IEEE Trans. on Comp., vol. 44, No. 2, February 1995, pp. 223-233

[4] Chakrabarty, K. – Murray, B.T. – Iyengar, V.: Built-in Test Pattern Generation for High-Performance Circuits Using Twisted-Ring Counters. Proc. of IEEE VLSI Test Symp. 1999

[5] Chatterjee, M., Pradhan, D. K.: A Novel Pattern Generator for Near Perfect Fault-Coverage, Proc. of IEEE VLSI Test Symp., 1995

[6] Touba, N.A, McCluskey, E.J.: Synthwesis of Mapping Logic for Generating Transformed Pseudo/random Patterns for BIST. Proc. of Inter. Test Conf., 1995, pp. 674-682

[7] H.-J. Wunderlich, G. Kiefer: Bit-Flipping BIST, Proc. of ACM/IEEE Inter. Conf. on CAD-96 (ICCAD96), San Jose,California, November 1996, pp.337-343

[8] Savir, J. Reducing the MISR Size. IEEE Trans. on Computers, vol 45, No. 8. August 1996, s. 930-938.

[9] Chakrabarty, K Zero-Aliasing Space Compaction Using Linear Compactors with Bounded Overhead. IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 17, No5, May,1998, pp 452-457.

[10] M. Lempel, M, Gupta, S. K, Zero-Aliasing for Modeled Faults. IEEE Trans. on Computers, vol. 44, no. 11, November 1995, pp 1283-1295

[11] Chou, R.M, Saluja, A, K. K., Agrawal, V.D.: Scheduling Tests for VLSI System Under Power Constraints; IEEE Trans. on Very Large Scale Integration Systems, Vol. 5 No. 2, June 1997, pp. 175-185

[12] Novak, O, Nosek, J.: On Using Deterministic Test Sets in BIST. Proc. of 6th IEEE Inter. On-Line Testing Workshop, Palma de Mallorca, Spain 2000, pp.127-132

[13] T. Garbolino, A. Hlawiczka: Easy Integration of Based on T-Type Flip-Flop Test Pattern Generators to the Scan Path. Proc. of European Test Workshop (ETW'00), Cascais, Portugal, May 2000.

[14] K. Gucwa: Zero aliasing linear compactors for BIST. A dissertation submitted to the Dep. of Automatic Control, Electronics and Computer Science of Silesian University of Technology, Gliwice, Poland, 2000

[15] Novak, O.: Pseudorandom,Weighted Random and Pseudoexhaustive Test Patterns Generated in Universal Cellular Automata, Springer: Lecture Notes in Computer Science 1667, Sept. 1999, pp. 303-320

[16] Garbolino T., Hlawiczka A.: A New LFSR with D and T Flip-Flops as an Effective Test Pattern Generator for VLSI Circuits, Springer: Lecture Notes in Computer Science 1667, Sept. 1999, pp. 321-338

[17] Agrawal V.D., Kime C.R., Saluja K.K: A Tutorial on Built-In Self-Test Part 2: Applications. IEEE Design and Test of Computers, June 1993, pp. 69-77.

[18] Abramovici M, Breuer M.A., Friedman A.D.: Digital Systems Testing and Testable Design. IEEE Press, 1990.

[19] Eschermann B., Wunderlich H.J.: Optimized Synthesis Techniques for Testable Sequential Circuits. IEEE Trans. on CAD. vol. 11, No 3, March, 1992, pp. 301-312.

[20] Hlawiczka A., Binda J.: The Optimized Synthesis of Self-Testable Finite State Machines Using BIST-PST Structures in ALTERA Structures. Proc. of 4th International Workshop on Field Programmable Logic and Applications, September 7-9, 1994, Praque, Lecture Notes in Computer Science No 849, Springer Verlag Press, pp. 120-122.

[21] Wang L.-T., McCluskey E.J.: Built-In Self-Test for Sequential Machines. Proc. 1987 International Test Conference, Washington, September 1987, pp. 334-341.