International Journal of VLSI System Design and Communication Systems

ISSN 2322-0929
Vol.04, Issue.09,
September-2016,
Pages:0825-0832

www.ijvdcs.org

Low-Power Programmable PRPG with Test Compression Capabilities

P. SUJATHA[1], M. MOSHE[2]

[1]PG Scholar, Dept of ECE, Princeton Institute of Engineering and Technology for Women, Ghatkesar, Hyderabad, TS, India,
E-mail: suji.padmam@gmail.com.

[2]Assoc Prof & HOD, Dept of ECE, Princeton Institute of Engineering and Technology for Women, Ghatkesar, Hyderabad, TS, India,
E-mail: moshe.nani@gmail.com.

Abstract: This project describes a low-power (LP) programmable generator capable of producing pseudorandom test patterns with desired toggling levels and enhanced fault coverage gradient compared with the best-to-date built-in self-test (BIST)- based pseudorandom test pattern generators. It is comprised of a linear finite state machine (a linear feedback shift register or a ring generator) driving an appropriate phase shifter, and it comes with a number of features allowing this device to produce binary sequences with preselected toggling (PRESTO) activity. We introduce a method to automatically select several controls of the generator offering easy and precise tuning. The same technique is subsequently employed to deterministically guide the generator toward test sequences with improved fault-coverage-to pattern-count ratios. Furthermore, this proposes an LP test compression method that allows shaping the test power envelope in a fully predictable, accurate, and flexible fashion by adapting the PRESTO-based logic BIST (LBIST) infrastructure. The proposed architecture is extended in such that the patterns generated from PRPG is gone through CUT and then to TRA to perform ATE.

Keywords: Built-In Self-Test (BIST), Low-Power (LP) Test, Pseudorandom Test Pattern Generators (PRPGS), Test Data Volume Compression.

I. INTRODUCTION

The test pattern generator produces test vectors that are applied to the tested circuit during pseudo-random testing of combinational circuits. The nature of the generator thus directly influences the fault coverage achieved the influence of the type of pseudo-random pattern generator on stuck-at fault coverage. Linear feedback shift registers (LFSRs) are mostly used as test pattern generators, and the generating polynomial is primitive to ensure the maximum period. We have shown that it is not necessary to use primitive polynomials, and moreover that their using is even undesirable in most cases. This fact is documented by statistical graphs. The necessity of the proper choice of a generating polynomial and an LFSR seed is shown here, by designing a mixed-mode BIST for the ISCAS benchmarks as the complexity of VLSI circuits constantly increases, there is a need of a built-in self-test (BIST) to be used. Built-in self-test enables the chip to test itself and to evaluate the circuit's response. Thus, the very complex and expensive external ATE (Automatic Test Equipment) may be completely omitted, or its complexity significantly reduced. Moreover, BIST enables an easy access to internal structures of the tested circuit, which are extremely hard to reach from outside. There have been proposed many BIST equipment design methods. In most of the state-of-the-art methods some kind of a pseudorandom pattern generator (PRPG) is used to produce vectors to test the circuit. These vectors are applied to the circuit either as they are, or the vectors are modified by some additional circuitry in order to obtain better fault coverage. Then the circuit's response to these vectors is evaluated in a response analyzer. Usually, linear feedback shift registers (LFSRs) or cellular automata (CA) are used as PRPGs, for their simplicity. Patterns generated by simple LFSRs or CA often do not provide a satisfactory fault coverage. Thus, these patterns have to be modified somehow. One of the most known approaches is the weighted random pattern testing. Here the LFSR code words are modified by a weighting logic to produce a test with given probabilities of occurrence of 0's and 1's at the particular circuit under test (CUT) inputs. Many papers dealing with the computation of the weights and the design of the weighting logic have been published.

II. EXISTING BASIC ARCHITECTURE

An n-bit PRPG connected with a phase shifter feeding scan chains forms a kernel of the generator producing the actual pseudorandom test patterns as shown in Fig.1. A linear feedback shift register or a ring generator can implement a PRPG. More importantly, however, n hold latches are placed between the PRPG and the phase shifter. Each hold latch is individually controlled via a corresponding stage of an n-bit toggle control register. As long as its enable input is asserted, the given latch is transparent for data going from the PRPG to the phase shifter, and it is said to be in the toggle mode. When the latch is disabled, it captures and saves, for a number of clock cycles, the corresponding bit of PRPG, thus feeding the phase shifter (and possibly some scan chains) with a constant value. It is now in the hold mode. It is worth noting that each phase shifter output is obtained by XOR-ing outputs of three different hold latches. Therefore, every scan chain remains in a low-power mode provided only disabled hold latches drive the corresponding phase shifter output the toggle control register supervises the hold latches. Its content comprises 0s and 1s, where 1s indicate latches in the toggle mode, thus transparent for data arriving from the PRPG. Their fraction determines a scan switching activity.

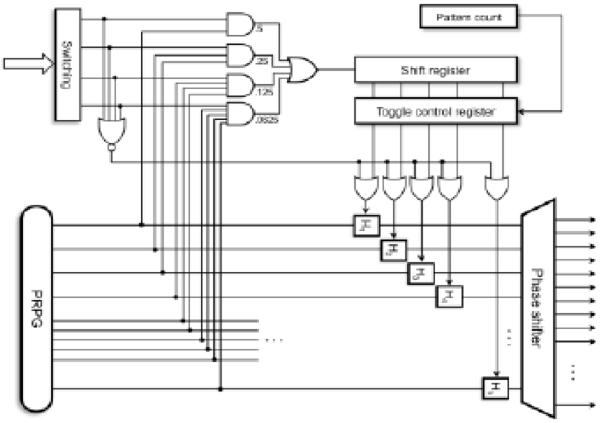


Fig.1. Basic architecture of a PRESTO generator.

The control register is reloaded once per pattern with the content of an additional shift register. The enable signals injected into the shift register are produced in a probabilistic fashion by using the original PRPG with a programmable set of weights. The weights are determined by four AND gates producing 1s with the probability of 0.5, 0.25, 0.125, and 0.0625, respectively. The OR gate allows choosing probabilities beyond simple powers of 2. A 4-bit register Switching is employed to activate AND gates, and allows selecting a user-defined level of switching activity. For example, the switching code 0100 will set to 1, on the average, 25% of the control register stages, and thus 25% of hold. Latches will be enabled. Given the phase shifter structure, one can assess then the amount of scan chains receiving constant values, and thus the expected toggling ratio. An additional 4-input NOR gate detects the switching code 0000, which is used to switch the LP functionality off. It is worth noting that when working in the weighted random mode, the switching level selector ensures statistically stable content of the control register in terms of the amount of 1s it carries. As a result, roughly the same fraction of scan chains will stay in the LP mode, though a set of actual low toggling chains will keep changing from one test pattern to another. It will correspond to a certain level of toggling in the scan chains. With only 15 different switching codes, however, the available toggling granularity may render this solution too coarse to be always acceptable. Section III presents additional features that make the PRESTO generator fully operational in a wide range of desired switching rates.

While preserving the operational principles of the basic solution, this approach splits up a shifting period of every test pattern into a sequence of alternating hold and toggle intervals. To move the generator back and forth between these two states, we use a T-type flip-flop that switches whenever there is a 1 on its data input. If it is set to 0, the generator enters the hold period with all latches temporarily disabled regardless of the control register content. This is accomplished by placing AND gates on the control register outputs to allow freezing of all phase shifter inputs. This property can be crucial in SoC designs where only a single scan chain crosses a given core, and its abnormal toggling may cause locally unacceptable heat dissipation that can only be reduced due to temporary hold periods if the T flip-flop is set to 1 (the toggle period), then the latches enabled through the control register can pass test. Data moving from the PRPG to the scan chains two additional parameters kept in 4-bit Hold and Toggle registers determine how long the entire generator remains either in the hold mode or in the toggle mode, respectively. To terminate either mode, a 1 must occur on the T flip-flop input. This weighted pseudorandom signal is produced in a manner similar to that of weighted logic used to feed the shift register. The T flip-flop controls also four 2-input multiplexers routing data from the Toggle and Hold registers. It allows selecting a source of control data that will be used in the next cycle to possibly change the operational mode of the generator .For example, when in the toggle mode, the input multiplexers observe the Toggle register as shown in Fig.2. Once the weighted logic outputs 1, the flip-flop toggles, and as a result all hold latches freeze in the last recorded state. They will remain in this state until another 1 occurs on the weighted logic output. The random occurrence of this event is now related to the content of the Hold register, which determines when to terminate the hold mode.

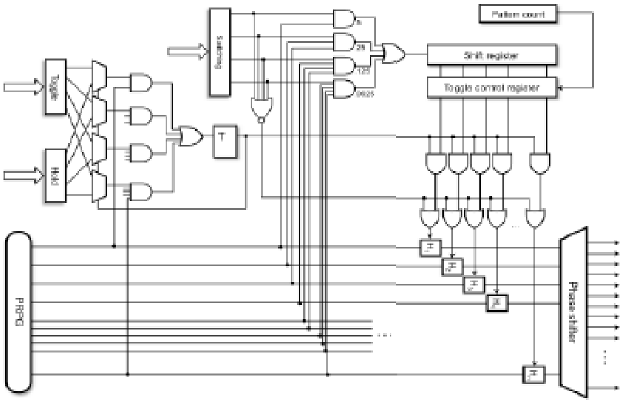


Fig.2. Fully operational version of PRESTO.

A. Improving Fault Coverage Gradient

A quest to achieve higher BIST fault coverage with shorter test application time generated an immense amount of research in the past. Typically, LFSR-based pseudorandom test sequences were modified either by placing a mapping logic between the PRPG outputs and inputs of a circuit under test , or by adjusting the probabilities of outputting 0s and 1s so that the resultant vectors capture characteristics of test patterns for hard-to-detect faults, as done in various for weighted-random testing. Test patterns leaving a PRPG can also be transformed in a more deterministic fashion. Along the same lines, we will demonstrate that PRESTO-produced LP test patterns are also capable of visibly improving a fault coverage- to-pattern-count ratio. Assuming that the toggle control registers can also be driven by deterministic test data (see location of an additional multiplexer in the front of a shift register. test patterns can be produced with better-than-average fault coverage. The proposed method begins by computing the PRESTO parameters Given the PRESTO switching code, our goal is now to find the corresponding distribution of 1s in the control register that maximizes the fault detection probability. The procedure starts by reducing each ATPG-produced test cube to a set of scan chains containing more than one specified bit. This set will be further referred to as a base. For example, let a test cube feature the following specified scan cells whereas is a scan chain, and c is a cell location within the scan chain. The base is thus given by {4, 14}; note that chain 45 is not included as it features only one specified scan cell. A good chance (50%) of producing a given logic value in a purely pseudorandom fashion is a rationale behind excluding from any base scan chains hosting a single specified bit. As a result, more bases can be subsequently combined together to produce a single control setting. Given the phase shifter architecture, one can determine, for each base, the minimal number of phase shifter inputs—or equivalently the number of 1s in the toggle control register— required to activate the specified scan chains.

These inputs are obtained by solving the minimum hitting set problem, where we find, in a greedy fashion, the minimal set of phase shifter inputs that intersects all subsets of phase shifter inputs capable of activating specified scan chains of a given base. Recall that the number of such inputs (and thus the number of 1s in the control register) is further constrained by the preselected switching code. Let C be an initially empty set of bases. Once all weights are determined, we add to C a minimum-weight base. Next, every remaining base B is assigned a cost value, which is equal to the smallest number of 1s in the control register that would be required to activate all scan chains in {C ∪ B}.A minimum-cost base (or a minimum-weight base if there are two or more bases with the same minimal cost) is then added to C, and costs associated with the remaining bases are recomputed accordingly. The procedure continues until either the limit of 1s in the control register is reached or all bases are already in C. The control register content that activates all scan chains from C is then provided to PRESTO. For each control register setting, PRESTO is run to produce a certain number of pseudorandom test patterns. These patterns are subsequently fault-simulated, and detected faults are dropped from the list. Experimental results demonstrating feasibility of this method.

III. PROPOSED ARCHITECTURE

The main challenging areas in VLSI are performance, cost, power dissipation is due to switching i.e. the power consumed testing, due to short circuit current flow and charging of load area, reliability and power. The demand for portable computing devices and communications system are increasing rapidly. These applications require low power dissipation VLSI circuits. The power dissipation during test mode is 200% P more than in normal mode. Hence it is important aspect to optimize power during testing. Power optimization is one of the main challenges.

A. BIST Architecture

A typical BIST architecture consists of

- TPG - Test Pattern Generator
- TRA – Test Response Analyzer
- Control Unit

As shown in Fig.3 below.

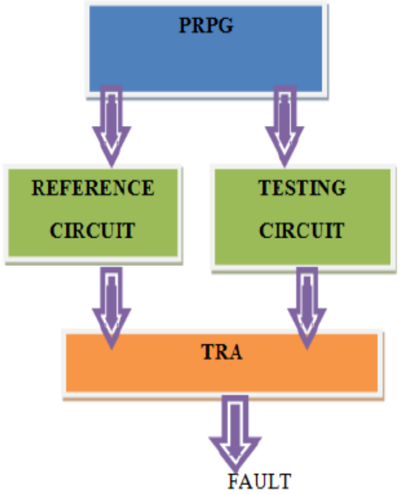


Fig.3.Test Pattern Generator.

It generates test pattern for CUT. It will be dedicated circuit or a micro processor. Pattern generated may be pseudo random numbers or deterministic sequence. Here we are using a Linear Feedback Shift Register for generating random number. The Architecture for LFSR is as shown below Fig.4.

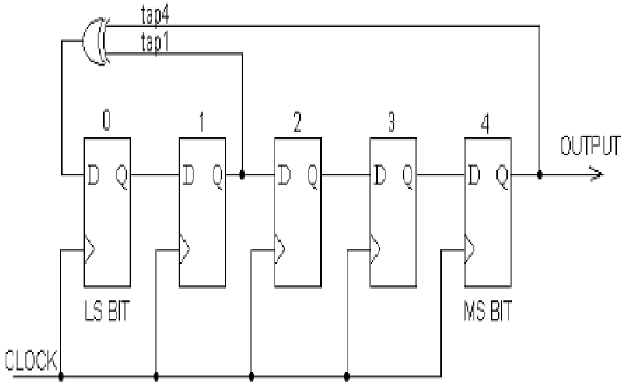


Fig.4. The Architecture for LFSR.

Tapping can be taken as we wish but as per taping change the LFSR output generate will change & as we change in no of flip-flop the probability of repetition of random number will reduce. The initial value loading to the LFSR is known as seed value.

Test Response Analyzer (TRA): TRA will check the output of MISR & verify with the input of LFSR & give the result as error or not.

Circuit under Test (CUT): CUT is the circuit or chip in which we are going to apply BIST for testing stuck at zero or stuck at one error as shown in Fig.5.

C14 Bench Mark Circuit:

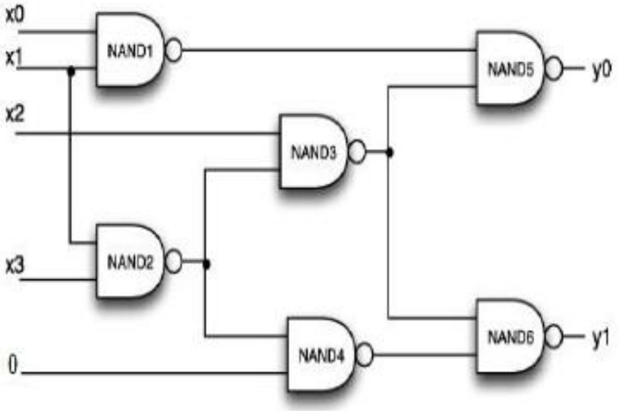


Fig.5. C14 Bench mark.

Need for using BIST Technique: Today's highly integrated multi-layer boards with fine-pitch ICs are virtually impossible to be accessed physically for testing. Traditional board test methods which include functional test, only accesses the board's primary I/Os, providing limited coverage and poor diagnostics for board-network fault in circuit testing, another traditional test method works by physically accessing each wire on the board via costly "bed of nails" probes and testers. To identify reliable testing methods which will reduce the cost of test equipment, a research to verify each VLSI testing problems has been conducted. The major problems detected so far are as follows:

- Test generation problems
- Gate to I/O pin ratio

Test Generation Problems: The large number of gates in VLSI circuits has pushed computer automatic-test-generation times to weeks or months of computation. The numbers of test patterns are becoming too large to be handled by an external tester and this has resulted in high computation costs and has outstripped reasonable available time for production testing.

The Gate to I/O Pin Ratio Problem: As ICs grow in gate counts, it is no longer true that most gate nodes are directly accessible by one of the pins on the package. This makes testing of internal nodes more difficult as they could neither no longer be easily controlled by signal from an input pin (controllability) nor easily observed at an output pin (observe ability). Pin counts go at a much slower rate than gate counts, which worsens the controllability and observe ability of internal gate nodes.

Cyclic Analysis Test System (CATS): Cyclic analysis test system (CATS) is a typical example of circular BIST. The architecture of CATS is shown in Fig.6. In test mode, the outputs are fed back to the inputs directly. The responses are used as the test vector without modification. If there are more inputs than outputs, one output may drive multiple inputs. If there are more outputs than inputs, we can use XOR gates to do space compression, as the one shown in Fig.6. The hardware overhead is very low. However, the fault coverage is circuit dependent. The recycling of test responses might create the fault masking effects. Note that, fault masking here is different from the aliasing discussed earlier. Here, the faulty and fault-free circuits have different test patterns.

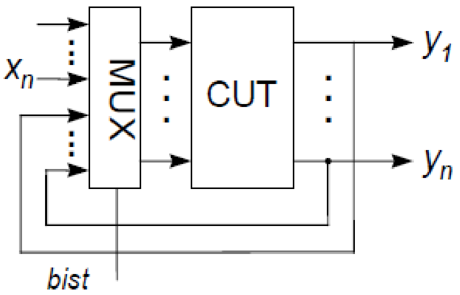


Fig.6. Cyclic analysis test system architecture.

Random Test Data (RTD): Random test data (RTD) transforms internal flip-flops into MISR. The circuit structure is shown in Fig.7. In normal mode, the MISR is operated as latches. In test mode, it operates as MISR. Both internal responses are compressed into and the internal test vectors are generated from the MISR. RTD is able to do one test per clock cycle. As compare to CATS, the hardware overhead is much higher. However, due to the extensive use of MISR, the test responses are scrambled before being used as the test patterns. Hence, the self masking probability can be lowered.

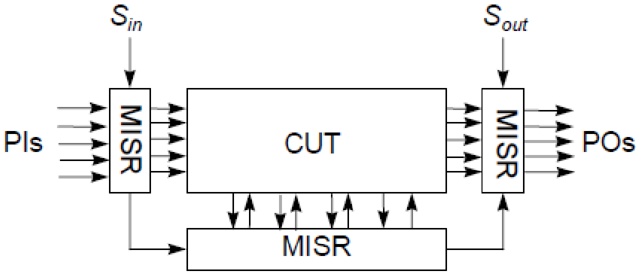


Fig.7. Random test data architecture.

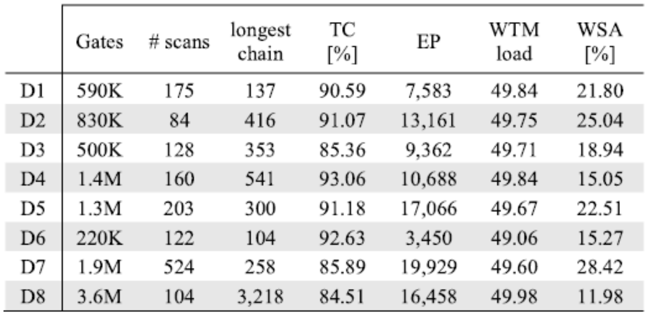
IV. EXPERIMENTAL RESULTS

This section presents experimental results obtained for the PRESTO generator and several industrial designs whose characteristics are given in Table I. For each test case, the table provides the number of gates, the number of scan chains, and the size of the longest scan chain. Furthermore, the column TC reports the resultant test coverage after applying 128K pseudorandom test patterns produced by the PRESTO generator with its LP features disabled. The next column (EP) lists the corresponding number of test patterns that effectively contributed to that level of fault coverage. Finally, the last two columns provide the WTM load for scan shift-in operations and the weighted switching activity (WSA) during the capture operation. As can be seen, WTM remains close to 50%, as typically observed in scan vectors produced in a pseudorandom fashion. The primary objective of the experiments was to measure test coverage as a function of several parameters, including:

- The number of test patterns;
- The switching activity code;
- The duration of Toggle (T) period;
- The duration of Hold (H) period.

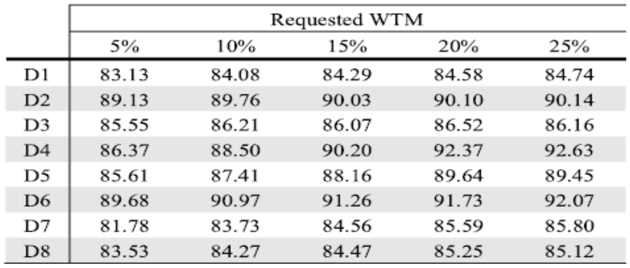
The actual results are presented in Tables II and III for the industrial designs of Table I. In all experiments reported here, we have used the PRESTO generator with a 32-bit ring generator producing 128K pseudorandom test patterns in a LP mode. Table II is vertically partitioned into columns corresponding to five different (target) toggling rates. Switching activity codes as well as parameters H and T were selected automatically. The columns of Table II list the fault coverage for successive test cases. As can be seen, the resultant fault coverage remains close to the reference coverage reported in Table I, while the switching activity is reduced to the desired levels of toggling. Note that some results indicate higher fault coverage if the scan chains receive the low toggling patterns rather than conventional pseudorandom vectors. Even if this is a circuit-specific feature, it nevertheless appears to be the case across several designs.

TABLE I: Circuit Characteristics—128K Random Patterns



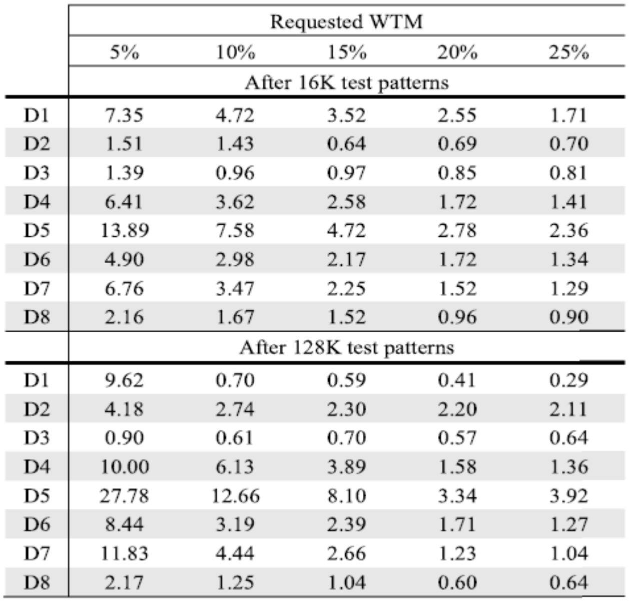
	Gates	# scans	longest chain	TC [%]	EP	WTM load	WSA [%]
D1	590K	175	137	90.59	7,583	49.84	21.80
D2	830K	84	416	91.07	13,161	49.75	25.04
D3	500K	128	353	85.36	9,362	49.71	18.94
D4	1.4M	160	541	93.06	10,688	49.84	15.05
D5	1.3M	203	300	91.18	17,066	49.67	22.51
D6	220K	122	104	92.63	3,450	49.06	15.27
D7	1.9M	524	258	85.89	19,929	49.60	28.42
D8	3.6M	104	3,218	84.51	16,458	49.98	11.98

TABLE II: Fault Coverage—128k Low Toggling Test Patterns



	Requested WTM				
	5%	10%	15%	20%	25%
D1	83.13	84.08	84.29	84.58	84.74
D2	89.13	89.76	90.03	90.10	90.14
D3	85.55	86.21	86.07	86.52	86.16
D4	86.37	88.50	90.20	92.37	92.63
D5	85.61	87.41	88.16	89.64	89.45
D6	89.68	90.97	91.26	91.73	92.07
D7	81.78	83.73	84.56	85.59	85.80
D8	83.53	84.27	84.47	85.25	85.12

TABLE III: Low Toggling Test Pattern Count Versus Random Vectors



	Requested WTM				
	5%	10%	15%	20%	25%
	After 16K test patterns				
D1	7.35	4.72	3.52	2.55	1.71
D2	1.51	1.43	0.64	0.69	0.70
D3	1.39	0.96	0.97	0.85	0.81
D4	6.41	3.62	2.58	1.72	1.41
D5	13.89	7.58	4.72	2.78	2.36
D6	4.90	2.98	2.17	1.72	1.34
D7	6.76	3.47	2.25	1.52	1.29
D8	2.16	1.67	1.52	0.96	0.90
	After 128K test patterns				
D1	9.62	0.70	0.59	0.41	0.29
D2	4.18	2.74	2.30	2.20	2.11
D3	0.90	0.61	0.70	0.57	0.64
D4	10.00	6.13	3.89	1.58	1.36
D5	27.78	12.66	8.10	3.34	3.92
D6	8.44	3.19	2.39	1.71	1.27
D7	11.83	4.44	2.66	1.23	1.04
D8	2.17	1.25	1.04	0.60	0.64

The objective of the analysis summarized in Table III was to determine the impact of our LP test generator performance on a pattern count. Alternatively, we would like to assess how long it takes to match fault coverage of purely pseudorandom test patterns (shown in the middle column of Table I) with vectors produced by the PRESTO generator. Let L(p) and R(p) denote fault coverage obtained by applying p low toggling and purely random test patterns, respectively. Clearly, there are two possible scenarios: either L(p) < R(p) or L(p) > R(p). In the first case, we can assess a pseudorandom test length q to get fault coverage L(p), where q < p. The other case is symmetrical; we need to find the number of LP test patterns r that suffice to match fault coverage R(p), where r < p. The entries of Table III, corresponding directly to those of Table II, are ratios vs that (depending on one of the above scenarios) are either equal to p/q or r/p. Clearly, v < 1 indicates cases where an LP test is shorter than its random counterpart. If v > 1, then the presented values are indicative of how many additional LP test patterns must be applied to obtain R(p). In Table III, two horizontal segments present results for two values of p: 16K and 128K. As an example, the entry 2.78 for design D5, 16K vectors, and WTM = 20% indicates that the resultant fault coverage due to 16K low toggling test patterns can be reached almost three times faster by using pseudorandom tests. On the other hand, the entry 0.57 for design D3, 128K vectors, and WTM = 20% indicates that LP tests can offer the same fault coverage as that of 128K random patterns in approximately half shorter test time. One may also observe that for some test cases the ratio v is quite large. It occurs either for aggressively low toggling rates or in some designs where certain groups of faults are much more difficult to detect by means of test patterns with relatively low diversity of binary sequences.

The objective of the second group of experiments is to assess effectiveness of the scheme, i.e., to measure a degree of test time reduction that one can achieve when using a pre-computed deterministic content of the control register as compared with application of pseudorandom patterns with otherwise similar power constraints. We present experimental results for industrial designs D1–D6 whose characteristics are given in Table I. All experiments are conducted using 32-bit PRESTO generator producing 1K test patterns for each of 128 predetermined control register settings. Hence, the total amount of control data is limited to 32×128 = 4096 b for 128K patterns. The number of test cubes generated in each iteration was set to 1000 resulting in typically three different control register settings per iteration (Section VI). In addition, in order to minimize the average number of specified bits occurring in test cubes, ATPG used a SCOAP-based decision order. The experimental results for 10% toggle rate represented by the WTM are shown in Fig.8. The presented curves correspond to the designs of Table I as follows. For BIST-ready designs D1 and D2, we depict their individual curves, while (in addition to their individual curves) a bold red line is averaging results over test cases D3, D4, D5, and D6. Given a number t of LP pseudorandom PRESTO generated test patterns (and hence the corresponding fault coverage C not shown in the figure), a single entry in these plots demonstrates a difference (or equivalently a gain) t–g, where g is the number of test patterns applied by a deterministically controlled PRESTO to arrive at the fault coverage C. For example, consider circuit D2 and its gain curve. As can be seen, we need roughly 70K fewer vectors to reach the same fault coverage as that of 100K PRESTO-produced pseudorandom test patterns with the same switching activity.

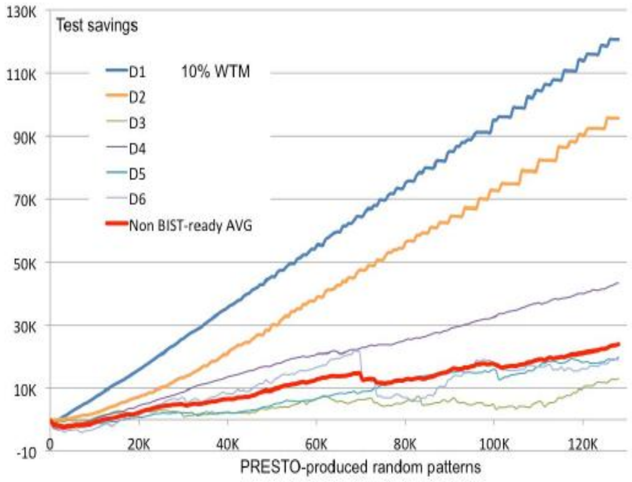


Fig.8. Pattern count savings for 10% WTM.

Clearly, test application time is reduced in this case by more than half. In the large majority of test cases, the deterministic control data allowed us to reduce the number of test patterns, and thus test application time, in a similar fashion. In particular, BIST-ready designs with a moderate number of scan chains witness considerably steep gain curves. We have also noticed little improvement in test time reduction for a few non-BIST ready circuits. It appears that these designs have featured a large number of scan chains driven by a relatively small phase shifter. Increasing the number of phase shifter inputs typically alleviates the situation. Fig.9 plots fault coverage results obtained for two BIST-ready designs D1 and D2 while choosing different toggling rates and sweeping the number of applied test patterns. As can be seen, in all examined cases fault coverage of test patterns generated by a deterministically controlled PRESTO (solid lines) is visibly improved over the baseline results (dashed lines) obtained for PRESTO-produced pseudorandom patterns with a similar switching activity. The improvement in fault coverage occurs systematically across all toggling rates, and the deterministically controlled PRESTO outperforms its conventional counterpart for virtually all examined test durations.

Eventually, we experimentally assess performance of the compression scheme of Sections VII and VIII. Experiments are run on industrial designs whose characteristics are given in Table IV. Table V presents results of experiments conducted with 64-bit de-compressors and the desired scan shift-in switching level set to 5%, 10%, and 15%. Again, the average WTM estimates the resultant switching activity for scan shift operations, while the average WSA measures toggling in the capture mode by observing the switching activity at each gate in the circuit. All experiments are conducted in such a way that the original EDT-based test coverage is always preserved. As can be seen, in all examined test cases the resultant scan shift-in switching activity (WTM load) remains very close to the requested one. We have also observed a similar trend for other switching rates, for which results are not reported in Table V. It is worth noting that reducing the load switching has a positive impact on the switching activity during capture and unloading of scan chains. Hence, the corresponding two figures of merit are included in the table as Capture WSA and WTM unload. It is also worth observing that the proposed solution is the first LP compression scheme that offers a mechanism to shape the power envelope in such a flexible and accurate fashion.

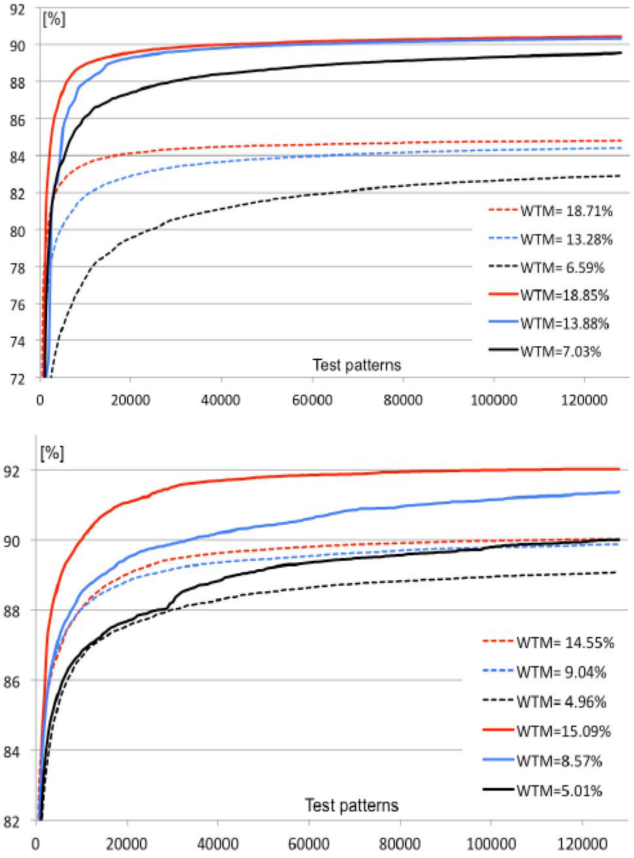


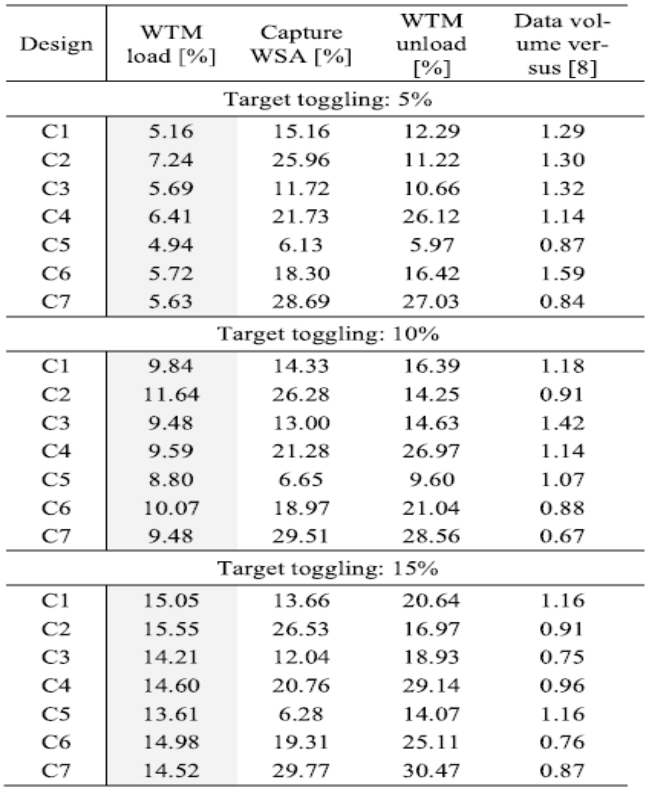
Fig.9. Fault coverage for two BIST-ready designs.

TABLE IV: Circuit Characteristics

Design	Gates	Scan cells	Scan chains	The longest chain	EDT inputs
C1	1.4M	86.4K	160	541	2
C2	2.0M	127K	523	256	2
C3	3.6M	297K	104	3,488	10
C4	1.3M	60.5K	203	300	2
C5	1.0M	75K	160	470	2
C6	226K	16.8K	122	138	2
C7	1.1M	110K	861	128	2

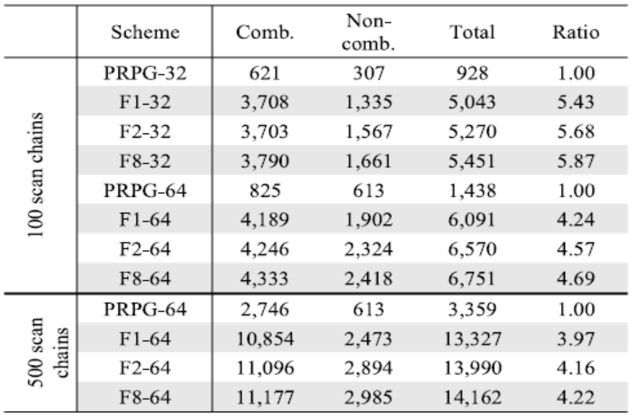
The last column reports the ratio V_P/V_F, where V_P is the volume of test data used to control the proposed scheme, and V_F is the corresponding amount of data used up by the LP EDT-based scheme presented in [9]. In addition to the actual seed variables, V_P comprises bits employed to feed the toggle control register, the Hold and Toggle registers, and the offset. Similarly, V_F includes seed variables and data necessary to control a broadcast scheme delivering gating signals to individual scan chains in a LP mode. Our solution gulps on the average only slightly more (1.05 times) test data than [9] for otherwise similar test coverage and switching activity. At the same time, the proposed technique delivers substantial functionality gains, as it is inherently capable of working as a programmable LP PRPG.

TABLE V: Experimental Results



Design	WTM load [%]	Capture WSA [%]	WTM unload [%]	Data volume versus [8]
Target toggling: 5%				
C1	5.16	15.16	12.29	1.29
C2	7.24	25.96	11.22	1.30
C3	5.69	11.72	10.66	1.32
C4	6.41	21.73	26.12	1.14
C5	4.94	6.13	5.97	0.87
C6	5.72	18.30	16.42	1.59
C7	5.63	28.69	27.03	0.84
Target toggling: 10%				
C1	9.84	14.33	16.39	1.18
C2	11.64	26.28	14.25	0.91
C3	9.48	13.00	14.63	1.42
C4	9.59	21.28	26.97	1.14
C5	8.80	6.65	9.60	1.07
C6	10.07	18.97	21.04	0.88
C7	9.48	29.51	28.56	0.67
Target toggling: 15%				
C1	15.05	13.66	20.64	1.16
C2	15.55	26.53	16.97	0.91
C3	14.21	12.04	18.93	0.75
C4	14.60	20.76	29.14	0.96
C5	13.61	6.28	14.07	1.16
C6	14.98	19.31	25.11	0.76
C7	14.52	29.77	30.47	0.87

TABLE VI: Area Overhead



	Scheme	Comb.	Non-comb.	Total	Ratio
100 scan chains	PRPG-32	621	307	928	1.00
	F1-32	3,708	1,335	5,043	5.43
	F2-32	3,703	1,567	5,270	5.68
	F8-32	3,790	1,661	5,451	5.87
	PRPG-64	825	613	1,438	1.00
	F1-64	4,189	1,902	6,091	4.24
	F2-64	4,246	2,324	6,570	4.57
	F8-64	4,333	2,418	6,751	4.69
500 scan chains	PRPG-64	2,746	613	3,359	1.00
	F1-64	10,854	2,473	13,327	3.97
	F2-64	11,096	2,894	13,990	4.16
	F8-64	11,177	2,985	14,162	4.22

The silicon real estate taken up by the proposed test logic amounts to an equivalent area of 2-input NAND gates, as shown in Table VI. It provides the actual area costs computed with a commercial synthesis tool for three architectures by using 32- and 64-bit ring generators (in the table denoted as F1-32, F2-64, and so on) feeding either n = 100 (the upper part) or n = 500 (the lower part) scan chains. All components of our test logic were synthesized using a 90-nm CMOS standard cell library under 3.5-ns timing constraint. The table reports the resultant silicon area with respect to combinational and non combinational devices. The total area is then compared with the corresponding area occupied by a conventional PRPG (typically, the XOR network of a phase shifter consists of n 3-input gates in addition to m flip-flops forming the ring generator—this reference area is reported in rows labeled as PRPG). For example, a 64-bit LP generator is 4.57 times larger than its standard counterpart, whereas it offers exceptional LP features. Consequently, the numbers of Table VI make the proposed scheme attractive as far as its silicon cost is concerned.

V. CONCLUSION

The proposed approach shows the concept of reducing the transitions in the test pattern generated. The transition is reduced by increasing the correlation between the successive bits. The simulation results shows that how the patterns are generated for the applied seed vector. This paper presents the implementation with regard to Verilog language. Synthesizing and implementation of the code is carried out on Xilinx - Project Navigator, ISE suite. The power reports show that the proposed architecture consumes less power during testing by taking the benchmark circuit C14. In future there is a chance to reduce the power somewhat more by doing modifications in the proposed architecture.

VI. REFERENCES

[1] Michał Filipek, Grzegorz Mrugalski, Senior Member, IEEE, Nilanjan Mukherjee, Senior Member, IEEE, Benoit Nadeau-Dostie, Senior Member, IEEE, Janusz Rajski, Fellow, IEEE, Je˛drzej Solecki, and Jerzy Tyszer, Fellow, IEEE, "Low-Power Programmable PRPG With Test Compression Capabilities", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 23, No. 6, June 2015.

[2] A. S. Abu-Issa and S. F. Quigley, "Bit-swapping LFSR for low-power BIST," Electron. Lett., vol. 44, no. 6, pp. 401–402, Mar. 2008.

[3] C. Barnhart et al., "Extending OPMISR beyond 10x scan test efficiency," IEEE Design Test, vol. 19, no. 5, pp. 65–73, Sep./Oct. 2002.

[4] S. Bhunia, H. Mahmoodi, D. Ghosh, S. Mukhopadhyay, and K. Roy, "Low-power scan design using first-level supply gating," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 13, no. 3, pp. 384–395, Mar. 2005.

[5] M. Chatterjee and D. K. Pradham, "A novel pattern generator for near perfect fault-coverage," in Proc. 13th IEEE Very Large Scale Integr. (VTSI) Test Symp., Apr./May 1995, pp. 417–425.

[6] F. Corno, M. Rebaudengo, M. S. Reorda, G. Squillero, and M. Violante, "Low power BIST via non-linear hybrid cellular automata," in Proc. 18th IEEE Very Large Scale Integr. (VTSI) Test Symp., May 2000, pp. 29–34.

[7] D. Das and N. A. Touba, "Reducing test data volume using external/LBIST hybrid test patterns," in Proc. Int. Test Conf. (ITC), 2000, pp. 115–122.

[8] R. Dorsch and H. Wunderlich, "Tailoring ATPG for embedded testing," in Proc. Int. Test Conf. (ITC), 2001, pp. 530–537.
[9] M. Filipek et al., "Low power de-compressor and PRPG with constant value broadcast," in Proc. 20th Asian Test Symp. (ATS), Nov. 2011, pp. 84–89.
[10] S. Gerstendorfer and H. Wunderlich, "Minimized power consumption for scan-based BIST," in Proc. Int. Test Conf. (ITC), 1999, pp. 77–84.
[11] V. Gherman, H. Wunderlich, H. Vranken, F. Hapke, M. Wittke, and M. Garbers, "Efficient pattern mapping for deterministic logic BIST," in Proc. Int. Test Conf. (ITC), Oct. 2004, pp. 48–56.
[12] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudo vitch, "A test vector inhibiting technique for low energy BIST design," in Proc. 17th IEEE VLSI Test Symp. (VTS), Apr. 1999, pp. 407–412.

Author's Profile:

P. Sujatha, (Student), ES & VLSI, Princeton Institute of Engineering and Technology for Women, Hyderabad, India,
Email Id: suji.padmam@gmail.com.

Mr.M.Moshe, received the Master of Technology degree in Communication and Signal Processing from the Bapatla Engineering College-ANU, he received the Bachelor of Engineering degree from QIS College of Engineering and Technology-JNTUH. He is currently working as Associate Professor and a Head of the Department of ECE with Princeton Institute of Engineering and Technology for Women, Ghatkesar, Hyd. His interest subjects are Microwave Engineering, Signals and Systems, Digital Signal Processing Communication Systems, Digital Electronics and etc, Email: moshe.nani@gmail.com.