Peak- and Average-Power Reduction in Check-Based BIST by using Bit-Swapping LFSR and Check-Chain Ordering

Mr.B.Baloji Naik[1], Dr.V.Venkata Rao[2]

[1](Student, Digital Electronics and Communication Systems, Narasaraopeta Engineering College/JNTU Kakinada, India)
[2](Professor & HOD, Electronics and Communication Engineering, Narasaraopeta Engineering College/JNTU Kakinada, India)

Abstract : *In this paper the proposed design, called bit-swapping LFSR (BS-LFSR), is composed of an LFSR and a 2 × 1 multiplexer. When used to generate test patterns for check-based built-in self-tests, it reduces the number of transitions that occur at the check-chain input during check shift operation by 50% when compared to those patterns produced by standard LFSR. Hence, it reduces the overall switching activity in the circuit under test during test applications. The BS-LFSR is combined with a check chain-ordering algorithm that orders the cells in a way that reduces the average and peak power (check and capture) in the test cycle or while checking out a response to a sign pattern analyzer. These techniques have a substantial effect on average- and peak-power reductions with negligible effect on fault coverage or test application time. Experimental results bench mark circuits show up to 65% and 55% reductions in average and peak power, respectively.*

Keywords: *Built-in self-test (BIST), linear feedback shift register (LFSR), low-power test, pseudorandom pattern generator, scan-chain ordering,*

I. INTRODUCTION

In recent years, the design for low power has become one of the greatest challenges in high-performance very large scale integration (VLSI) design. As a consequence, many techniques have been introduced to minimize the power consumption of new VLSI systems. However, most of these methods focus on the power consumption during normal mode operation, while test mode operation has not normally been a predominant concern. However, it has been found that the power consumed during test mode operation is often much higher than during normal mode operation [1]. This is because most of the consumed power results from the switching activity in the nodes of the circuit under test (CUT), which is much higher during test mode than during normal mode operation [1]–[3].Several techniques that have been developed to reduce the peak and average power dissipated during scan-based tests can be found in [4] and [5]. A direct technique to reduce power consumption is by running the test at a slower frequency than that in normal mode. This technique of reducing power consumption, while easy to implement, significantly increases the test application time [6]. Furthermore, it fails in reducing peak-power consumption since it is independent of clock frequency. Another category of techniques used to reduce the power consumption in scan-based built-in self-tests (BISTs) is by using scan chain-ordering techniques [7]–[13]. These techniques aim to reduce the average-power consumption when scanning in test vectors and scanning out captured responses. Although these algorithms aim to reduce average-power consumption, they can reduce the peak power that may occur in the CUT during the scanning cycles, but not the capture power that may result during the test cycle (i.e., between launch and capture). The design of low-transition test-pattern generators (TPGs) is one of the most common and efficient techniques for low-power tests [14]–[20]. These algorithms modify the test vectors generated by the LFSR to get test vectors with a low number of transitions. The main drawback of these algorithms is that they aim only to reduce the average-power consumption while loading a new test vector, and they ignore the power consumption that results while scanning out the captured response or during the test cycle. Furthermore, some of these techniques may result in lower fault coverage and higher test application time. Other techniques to reduce average-power consumption during scan-based tests include scan segmentation into multiple scan chains [6], [21], test-scheduling techniques [22], [23], static compaction techniques [24], and multiple scan chains with many scan enable inputs to activate one scan chain at a time [25]. The latter technique also reduces the peak power in the CUT.

On the other hand, in addition to the techniques mentioned earlier, there are some new approaches that aim to reduce peak-power consumption during tests, particularly the capture power in the test cycle. One of the common techniques for this purpose is to modify patterns using an X-filling technique to assign values to the don't care bits of a deterministic set of test vectors in such a way as to reduce the peak power in the test vectors that have a peak-power violation [26]–[29]. This paper presents a new TPG, called the bit-swapping linear feedback shift register (BS-LFSR), that is based on a simple bit swapping technique applied to the output

sequence of a conventional LFSR and designed using a conventional LFSR and a 2×1 multiplexer. The proposed BS-LFSR reduces the average and instantaneous weighted switching activity (WSA) during test operation by reducing the number of transitions in the scan input of the CUT. The BSLFSR is combined with a scan-chain-ordering algorithm that reduces the switching activity in both the test cycle (capture power) and the scanning cycles (scanning power).

II.   Proposed Approach To Design The Bs-Lfsr

The proposed BS-LFSR for test-per-scan BISTs is based upon some new observations concerning the number of transitions produced at the output of an LFSR.

Definition: Two cells in an *n*-bit LFSR are considered to be adjacent if the output of one cell feeds the input of the second directly (i.e., without an intervening XOR gate).

Lemma 1: Each cell in a maximal-length *n*-stage LFSR (internal or external) will produce a number of transitions equal to $2^n - 1$. After going through a sequence of 2^n. Clock cycles.

Proof: The sequence of 1s and 0s that is followed by one bit

position of a maximal-length LFSR is commonly referred to as an m-*s*equence. Each bit within the LFSR will follow the same *m*-sequence with a one-time-step delay. The *m*-sequence generated by an LFSR of length *n* has a periodicity of $2^n - 1$. It is a well-known standard property of an *m*-sequence of length *n* that the total number of runs of consecutive occurrences of the same binary digit is $2^n - 1$. [3], [30]. The beginning of each run is marked by a transition between 0 and 1; therefore, the total number of transitions for each stage of the LFSR is $2^n - 1$. This lemma can be proved by using the toggle property of the XOR gates used in the feedback of the LFSR [32].
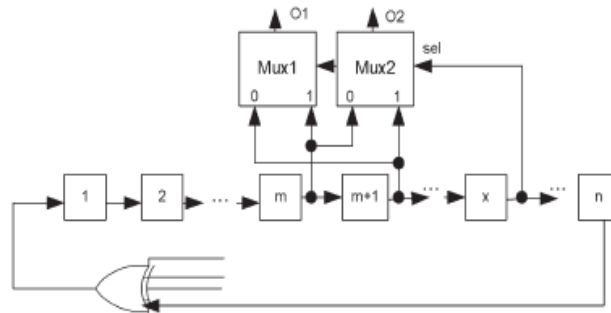

Fig. 1. Swapping arrangement for an LFSR.

Lemma 2: Consider a maximal-length *n*-stage internal or external LFSR ($n > 2$). We choose one of the cells and swap its value with its adjacent cell if the current value of a third cell in the LFSR is 0 (or 1) and leave the cells unswapped if
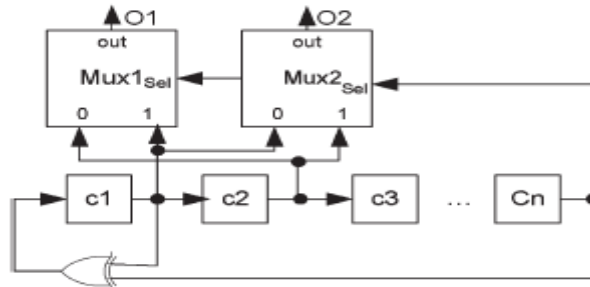

Fig. 2. External LFSR that implements the prime polynomial $x^n + x + 1$ and the proposed swapping arrangement.

The third cell has a value of 1 (or 0). Fig. 1 shows this arrangement for an external LFSR (the same is valid for an internal LFSR). In this arrangement, the output of the two cells will have its transition count reduced by $T_{saved} = 2^{n-2}$ transitions. Since the two cells originally produce $2 \times 2^{n-1}$ transitions, then the resulting percentage saving is $T_{saved}\% = 25\%$ [32].

In Lemma 2, the total percentage of transition savings after swapping is 25% [31]. In the case where cell *x* is not directly linked to cell *m* or cell *m*+ 1 through an XOR gate, each of the cells has the same share of savings (i.e., 25%).   Lemmas 3–10 show the special cases where the cell that drives the selection line is linked to one of the swapped cells through an XOR gate. In these configurations, a single cell can save 50% transitions that were originally produced by an LFSR cell. Lemma 3 and its proof are given; other lemmas can be proved in the same way.

Lemma 3: For an external *n*-bit maximal-length LFSR that implements the prime polynomial $x^n + x + 1$ as shown in Fig. 2, if the first two cells (c_1 and c_2) have been chosen for swapping and cell *n* as a selection line, then o_2 (the output of MUX2) will produce a total transition savings of 2^{n-2} compared to the number of transitions produced by each LFSR cell, while o_1 has no savings (i.e., the savings in transitions is concentrated in one multiplexer output, which means that o_2 will save 50% of the original transitions produced by each LFSR cell).

Proof: There are eight possible combinations for the initial state of the cells c_1, c_2, and c_n. If we then consider all possible values of the following state, we have two possible combinations (not eight, because the value of c_2 in the next state is determined by the value of c_1 in the present state; also, the value of c_1 in the next state is determined by "c_1 xor c_n" in the present state). Table I shows all possible and subsequent states.

It is important to note that the overall savings of 25% is not equally distributed between the outputs of the multiplexers as in Lemma 2.This is because the value of c_1 in the present state will affect the value of c_2 and its own value in the next state (c_2 (Next) = c_1 and c_1 (Next) = "c_1 xor c_n"). To see the effect of each cell in transition savings, Table 1 shows that o_1 will save one transition when moving from state (0,0,1) to (1,0,0), from (0,1,1) to (1,0,0), from (1,0,1) to (0,1,0), or from (1,1,1) to (0,1,0). In the same time, *o*1 will increase one transition when moving from (0,1,0) to (0,0,0), from (0,1,0) to (0,0,1), from (1,0,0) to (1,1,0), or from (1,0,0) to (1,1,1). Since *o*1 increases the transitions in four possible scenarios and save transitions in other four scenarios, then it has a neutral overall effect because all the scenarios have the same probabilities. For o_2, one transition is saved when moving from (0,1,0) to (0,0,0), from (0,1,0) to (0,0,1), from (0,1,1) to (1,0,0), from (1,0,0) to (1,1,0), from (1,0,0) to (1,1,1), or from (1,0,1) to (0,1,0). At the same time, one additional transition is incurred when moving from state (0,0,1) to (1,0,0) or from (1,1,1) to (0,1,0). This gives o_2 an overall saving of one transition in four possible scenarios where the initial states has a probability of 1/8 and the final states of probability 1/2;

| LFSR outputs of m, m+1 | | | | | | | | | Multiplexers outputs O₁, O₂ | | | | | | |
| States | | | Next states | | | transition | | | states | | Next States | | transition | | |
c_1	c_2	c_n	c_1	c_2	c_n	c_1	c_2	Σ	O_1	O_2	O_1	O_2	O_1	O_2	Σ
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			0	0	1	0	0	0			0	0	0	0	0
0	0	1	1	0	0	1	0	1	0	0	0	1	0	1	1
			1	0	1	1	0	1			1	0	1	0	1
0	1	0	0	0	0	0	1	1	1	0	0	0	1	0	1
			0	0	1	0	1	1			0	0	1	0	1
0	1	1	1	0	0	1	1	2	0	1	0	1	0	0	0
			1	0	1	1	1	2			1	0	1	1	2
1	0	0	1	1	0	0	1	1	0	1	1	1	1	0	1
			1	1	1	0	1	1			1	1	1	0	1
1	0	1	1	0	0	1	1	2	1	0	1	0	0	0	0
			0	1	1	1	1	2			0	1	1	1	2
1	1	0	1	1	0	0	0	0	1	1	1	1	0	0	0
			1	1	1	0	0	0			1	1	0	0	0
1	1	1	1	0	0	1	0	1	1	1	1	0	0	1	1
			0	1	1	1	0	1			0	1	1	0	1
ΣTransitions						8	8	16					8	4	12

Table 1: POSSIBLE AND SUBSEQUENT STATES FOR CELLS c_1, c_2, AND c_3 (SEE FIG. 2)

Hence, *P* save is given by
*P*save = 1/8 × 1/2 + 1/8 × 1/2 + 1/8 × 1/2 + 1/8 × 1/2 = 1/4.

If the LFSR is allowed to move through a complete cycle of 2^n states, then Lemma 1 shows that the number of transitions expected to occur in the cell under consideration is 2^{n-1}. Using the swapping approach, in 1/4 of the cases, a saving of one transition will occur, giving a total saving of 1/4 × 2^n = 2^{n-2}. Dividing one figure by the other, we see that the total number of transitions saved at o_2 is 50%.

In the special configurations shown in Table 2 (i.e. Lemmas 3–10), if the cell that saves 50% of the transitions is connected to feed the scan-chain input, then it saves 50% of the transitions inside the scan chain cells, which directly reduces the average power and also the peak power that may result while scanning in a new test vector.

Table 3 shows that there are 104 LFSRs (internal and external) whose sizes lie in the range of 3–168 stages that can be configured to satisfy one or more of the special cases in Table II to concentrate the transition savings in one multiplexer output.

Lemmas	LFSR Polynomial	LFSR Type	Swapped cells 1st	2nd	Selection Line	MUX Out 50% Save
Lemma 3	x^n+x+1	External	C_1	C_2	C_n	O_2
Lemma 4	x^n+x+1	Internal	C_1	C_n	C_2	O_2
Lemma 5	$x^n+x^{n-1}+1$	External	C_{n-1}	C_n	C_1	O_1
Lemma 6	$x^n+x^{n-1}+1$	Internal	C_1	C_n	C_{n-1}	O_1
Lemma 7	x^n+x^2+1	External	C_1	C_2	C_n	O_1
Lemma 8	$x^n+x^{n-2}+1$	Internal	C_{n-1}	C_n	C_{n-2}	O_1
Lemma 9	$x^n+x^{n-1}+x^{ym}+\ldots+x^{y2}+x^{y1}+1$	Internal	C_1	C_n	C_{n-1}	O_1
Lemma 10	$x^n+x^{n-2}+x^{ym}+\ldots+x^{y2}+x^{y1}+1$	Internal	C_{n-1}	C_n	C_{n-2}	O_1

Table 2:SPECIAL CASES WHERE ONE CELL SAVES 50% OF THE TRANSITIONS

# of LFSR Stages	LFSR settle one or more of Lemmas 3 to 10 in table 2
3-20	3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15, 16, 19
21-40	21, 22, 24, 26, 27, 29, 30, 32, 34, 35, 37, 38, 40
41-60	42, 43, 44, 45, 46, 48, 50, 51, 53, 54, 56, 59, 60
61-80	61, 62, 63, 64, 66, 67, 69, 70, 74, 75, 76, 77, 78, 80
81-100	83, 85, 86, 88, 90, 91, 92, 93, 96, 99
101-120	101, 102, 104, 107, 109, 110, 112, 114, 115, 116, 117
121-140	122, 123, 125, 126, 127, 128, 131, 133, 136, 138
141-160	141, 143, 144, 146, 147, 149, 152, 153, 154, 155, 156, 157, 158, 160
161-168	162, 163, 164, 165, 166, 168
Total	**104**

Table 3: LFSRS THAT SATISFY ONE OR MORE OF LEMMAS 3–10

III. Important Properties Of The Bs-Lfsr

There are some important features of the proposed BS-LFSR that make it equivalent to a conventional LFSR. The most important properties of the BS-LFSR are the following.

1) The proposed BS-LFSR generates the same number of 1s and 0s at the output of multiplexers after swapping of two adjacent cells; hence, the probabilities of having a 0 or 1 at a certain cell of the scan chain before applying the test vectors are equal. Hence, the proposed design retains an important feature of any random TPG. Furthermore, the output of the multiplexer depends on three different cells of the LFSR, each of which contains a pseudorandom value. Hence, the expected value at the output can also be considered to be a pseudorandom value.

2) If the BS-LFSR is used to generate test patterns for either test per- clock BIST or for the primary inputs of a scan-based sequential circuit (assuming that they are directly accessible) as shown in Fig. 3, then consider the case that c_1 will be swapped with c_2 and c_3 with c_4, . . . , c_{n-2} with c_{n-1} according to the value of c_n which is connected to the selection line of the multiplexers (see Fig. 3). In this case, we have the same exhaustive set of test vectors as would be generated by the conventional LFSR, but their order will be different and the overall transitions in the primary inputs of the CUT will be reduced by 25% [32].

IV. Cell Reordering Algorithm

Although the proposed BS-LFSR can achieve good results in reducing the consumption of average power during test and also in minimizing the peak power that may result while scanning a new test vector, it cannot reduce the overall peak power because there are some components that occur while scanning out the captured response or while applying a test vector and capturing a response in the test cycle. To solve these problems, first, the proposed BS-LFSR has been combined with a cell-ordering algorithm presented in [11] that reduces the number of transitions in the check chain while checking out the captured response. This will reduce the
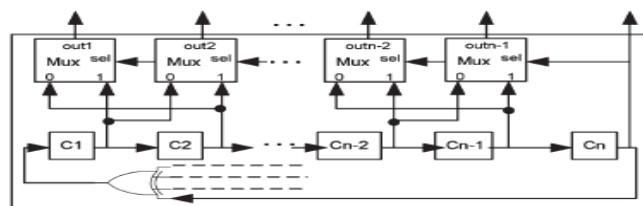

Fig. 3. BS-LFSR can be used to generate exhaustive patterns for test-per clock BIST.

overall average power and also the peak power that may arise while scanning out a captured response. The problem of the capture power (peak power in the test cycle) will be solved by using a novel algorithm that will

reorder some cells in the scan chain in such a way that minimizes the Hamming distance between the applied test vector and the captured response in the test cycle, hence reducing the test cycle peak power (capture power). In this scan-chain-ordering algorithm, some cells of the ordered scan chain using the algorithm in [11] will be reordered again in order to reduce the peak power which may result during the test cycle. This phase mainly depends on an important property of the BS-LFSR. This property states that, if two cells are connected with each other, then the probability that they have the same value at any clock cycle is 0.75. (In a conventional LFSR where the transition probability is 0.5, two adjacent cells will have the same value in 50% of the clocks and different values in 50% of the clocks; for a BS-LFSR that reduces the number of transition of an LFSR by 50%, the transition probability is 0.25, and hence, two adjacent cells will have the same value in 75% of the clock cycles.) Thus, for two connected cells (cells j and k), if we apply a sufficient number of test vectors to the CUT, then the values of cells j and k are similar in 75% of the applied vectors. Hence, assume that we have cell x which is a function of cells y and z. If the value that cell x will have in the captured response is the same as its value in the applied test vector (i.e., no transition will happen for this cell in the test cycle) in the majority of cases where cells y and z have the same value, then we connect cells y and z together on the scan chain, since they will have the same value in 75% of the cases. This reduces the possibility that cell x will undergo a transition in the test cycle. The steps in this algorithm are as follows.

1) Simulate the CUT for the test patterns generated by the BS-LFSR.
2) Identify the group of vectors and responses that violate the peak power.
3) In these vectors, identify the cells that mostly change their values in the test cycle and cause the peak-power violation.
4) For each cell found in step 3), identify the cells that play the key role in the value of this cell in the test cycle.
5) If it is found that, when two cells have a similar value in the applied test vector, the concerned cell will most probably have no transition in the test cycle, then connect these cells together. If it is found that, when two cells have a different value, the cell under consideration will most probably have no transitions in the test cycle, then connect these cells together through an inverter.

It is important to note that this phase of ordering is done when necessary only, as stated in step 2 of the algorithm description that the group of test vectors that violates the peak power should be identified first. Hence, if no vector violates the peak power, then this phase will. not be done. In the worst case, this phase is performed in few subsets of the cells. This is because, if this phase of ordering is done in all cells of the scan chain, then it will destroy the effect of algorithm found in [11] and will substantially increase the computation time.

Circuit	n	m	PI	RF%	FC%	Test Length			
						Det.	LFSR	BS-LFSR no order	BS-LFSR with order
S641	32	19	35	0	98.0	53	5120	4910	4970
S838	32	32	35	0	86.5	90	8160	8460	7910
S1196	30	18	14	0	97.0	131	3750	3680	3370
S1238	30	18	14	5.09	91.3	141	3890	3560	3610
S5378	40	179	35	0.88	98.0	244	30110	33700	28900
S9234	40	228	19	6.52	90.0	367	397800	401930	398170
S13207	60	669	31	1.54	95.0	455	49660	47400	48110
S35932	64	1728	35	10.19	89.8	63	18700	16640	16520
S38417	64	1636	28	0.53	96.5	849	118580	125520	117080
S38584	64	1452	12	4.15	94	632	43530	39660	40090

Table 4: TEST LENGTH NEEDED TO GET TARGET FAULT COVERAGE FOR LFSR AND BS-LFSR

Circuit	TL	LFSR			BS-LFSR with cell ordering			%Savings of BS-LFSR	
		FC%	WSA_{avg}	WSA_{pk}	FC%	WSA_{avg}	WSA_{pk}	WSA_{av}	WSA_{pk}
S641	3000	97.84	97.78	153	97.54	42.20	84	57	45
S838	20000	96.15	81.91	151	96.21	33.14	83	60	45
S1196	2000	95.33	53.18	74	95.51	21.52	42	60	43
S1238	3000	91.11	61.20	97	90.97	34.80	59	43	39
S5378	40000	98.42	1143.24	1639	98.40	625.28	993	45	39
S9234	100000	87.27	2817.45	3988	87.28	1108.93	2197	61	45
S13207	100000	96.45	4611.67	7108	96.39	1897.33	4172	59	41
S35932	200	87.88	7945.81	12592	87.89	2793.16	5723	65	55
S38417	100000	95.73	10965.50	16380	95.68	5022.30	10017	54	39
S38584	100000	94.46	11194.65	15974	94.48	5682.72	7851	49	51

Table 5: EXPERIMENTAL RESULTS OF AVERAGE- AND PEAK-POWER REDUCTION OBTAINED BY USING THE PROPOSED TECHNIQUES

V.   Experimental Results

A group of experiments was performed on full-scan ISCAS'89 benchmark circuits. In the first set of experiments, the BS-LFSR is evaluated regarding the length of the test sequence needed to achieve a certain fault coverage with and without the scan-chain-ordering algorithm. Table 4 shows the results for a set of ten benchmark circuits. The columns labeled n, m, and PI refer to the sizes of the LFSR, the number of flip-flops in the scan chain, and the number of primary inputs of the CUT, respectively. The column labeled RF indicates the percentage of redundant faults in the CUT, and fault coverage (FC) indicates the target fault coverage where redundant faults are included. The last four columns show the test length needed by a deterministic test (i.e., the optimal test vector set is stored in a ROM), a conventional LFSR, a BS-LFSR with no scan-chain ordering, and the BS-LFSR with scan-chain ordering, respectively. The results in Table 4 show that the BS-LFSR needs a shorter test length than a conventional LFSR for many circuits even without using the check chain-ordering algorithm. It also shows that using the scan-chain ordering algorithm with BS-LFSR will shorten the required test length. The second set of experiments is used to evaluate the BS-LFSR together with the proposed scan-chain-ordering algorithm in reducing average and peak power. For each benchmark circuit, the same numbers of conventional LFSR and BS-LFSR patterns are applied to the full scan configuration. Table 5 shows the obtained results for the same benchmark circuits as in Table 4. The column labeled test length (TL) refers to the number of test vectors applied to the CUT. The next three columns show the FC, average WSA per clock cycle (WSAavg), and the maximum WSA in a clock cycle (WSApeak) for patterns applied using the conventional LFSR. The next three columns show FC, WSAavg, and WSApeak for the BS-LFSR with ordered scan chain. Finally, the last two columns show the savings in average and peak power by using the BS-LFSR with the scan-chain-ordering algorithm.

In order to provide a comparison with the techniques published previously by other authors, Table 6 compares the results obtained in [15].

Circuit	Results in [15]			Results of proposed method		
	TL	FC	%WSA$_{av}$	TL	FC	%WSA$_{av}$
S641	4096	97.21	38	3000	97.54	57
S838	4096	95.46	50	20000	96.21	60
S1196	4096	95.59	17	2000	95.51	60
S1238	4096	89.41	17	3000	90.97	43
S5378	65536	96.54	43	40000	98.40	45
S9234	524288	90.89	62	100000	87.28	61
S13207	132072	93.66	45	100000	96.39	59
S35932	128	87.84	56	200	87.89	65
S38417	132072	94.99	56	100000	95.68	54
S38584	132072	93.35	59	100000	94.48	49
AVG	100255	93.49	44	46820	94.04	55

Table 6: COMPARISON WITH RESULTS OBTAINED IN [15]

Circuit	Results in [25] WSA$_{pk}$ Savings %	Proposed Method WSA$_{pk}$ Savings %
S5378	36.6	39
S9234	38.9	45
S13207	46.1	41
S38417	40.1	39
S38584	35.9	51
AVG	39.5	43.0

Table7: COMPARISON OF PEAK-POWER REDUCTIONS WITH RESULTS IN [25]

Table 6 compares the TL, FC, and average-power reduction (WSAavg). It is clear that the proposed method is much better for most of the circuits, not only in average-power reduction but also in the test length needed to obtain good fault coverage. Finally, Table 7 compares the results obtained by the proposed technique for peak-power reduction with those obtained in [25]. It is clear from the table that the proposed method has better results for most of the benchmark circuits.

VI.   Conclusion

A low-transition TPG that is based on some observations about transition counts at the output sequence of LFSRs has been presented. The proposed TPG is used to generate test vectors for test-per check BISTs in order to reduce the switching activity while scanning test vectors into the scan chain. Furthermore, a novel algorithm for scan-chain ordering has been presented. When the BS-LFSR is used together with the proposed scan-chain-ordering algorithm, the average and peak powers are substantially reduced. The effect of the proposed design in the fault coverage, test-application time, and hardware area overhead is negligible. Comparisons between the proposed design and other previously published methods show that the proposed design can achieve better results for most tested benchmark circuits.

References

[1]   Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," in *Proc. 11th IEEE VTS*, Apr. 1993 pp. 4–9.

[2]   A. Hertwig and H. J. Wunderlich, "Low power serial built-in self-test," in *Proc. IEEE Eur. Test Workshop*, May   1998, pp. 49–53..

[3]   P. H. Bardell, W. H. McAnney, and J. Savir, *Built-in Test for VLSI: Pseudorandom Techniques*. New York: Wiley, 1997.

[4]   P. Girard, "Survey of low-power testing of VLSI circuits," *IEEE Des. Test Comput.*, vol. 19, no. 3, pp. 80–90,   May/Jun. 2002.

[5]   K. M. Butler, J. Saxena, T. Fryars, G. Hetherington, A. Jain, and J. Lewis "Minimizing power consumption in   scan testing: Pattern generation and DFT techniques," in *Proc. Int. Test Conf.*, 2004, pp. 355–364.

[6]   J. Saxena, K. Butler, and L. Whetsel, "An analysis of power reduction techniques in scan testing," in *Proc. Int. Test Conf.*, 2001, pp. 670–677.

[7]   V. Dabhholkar, S. Chakravarty, I. Pomeranz, and S. M. Reddy, "Techniques for minimizing power dissipation in   scan and combinational   circuits during test applications," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 17, no. 12, pp. 1325–1333, Dec. 1998.

[8]   Y. Bonhomme, P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and V. Virazel, "Design of routing- constrained low power scan chains," in *Proc. Des. Autom. Test Eur. Conf. Exhib.*, Feb. 2004, pp. 62–67.

[9]   W. Tseng, "Scan chain ordering technique for switching activity reduction during scan test," *Proc. Inst. Elect.   Eng.—Comput. Digit. Tech.*, vol. 152, no. 5, pp. 609–617, Sep. 2005.

[10]   C. Giri, B. Kumar, and S. Chattopadhyay, "Scan flip-flop ordering with delay and power minimization during   testing," in *Proc. Annu. IEEE INDICON*, Dec. 2005, pp. 467–471

[11]   Y. Bonhomme, P. Girard, C. Laundrault, and S. Pravossoudovitch, "Power driven chaining of flip-flops in scan   architectures," in *Proc. Int. Test Conf.*, Oct. 2002, pp. 796–803.

[12]   M. Bellos, D. Bakalis, and D. Nikolos, "Scan cell ordering for low power BIST," in *Proc. IEEE Comput. Soc.   Annu. Symp. VLSI*, Feb. 2004,pp. 281–284.

[13]   K. V.A. Reddy and S. Chattopadahyay, "An efficient algorithm to reduce test power consumption by scan cell and scan vector reordering," in *Proc. IEEE 1st India Annu. Conf. INDICON*, Dec. 2004, pp. 373–376.

[14]   S. Wang, "A BIST TPG for low power dissipation and high fault coverage," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15,   no. 7,pp. 777–789, Jul. 2007.

[15]   S.Wang and S. Gupta, "LT-RTPG: A new test-per-scan BIST TPG for low switching activity," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 8, pp. 1565–1574, Aug. 2006.

[16]   S. Wang and S. K. Gupta, "DS-LFSR: A BIST TPG for low switching activity," *IEEE Trans. Comput.-Aided   Design Integr. Circuits Syst.*, vol. 21, no. 7, pp. 842–851, Jul. 2002.

[17]   H. Ronghui, L. Xiaowei, and G. Yunzhan, "A low power BIST TPG design," in *Proc. 5th Int. Conf. ASIC*, Oct. 2003, vol. 2, pp. 1136–1139.

[18]   L. Jie, Y. Jun, L. Rui, and W. Chao, "A new BIST structure for low power testing," in *Proc. 5th Int. Conf. ASIC*, Oct. 2003, vol. 2, pp. 1183– 1185.

[19]   M. Tehranipoor, M. Nourani, and N. Ahmed, "Low transition LFSR for BIST-based applications," in *Proc. 14th ATS*, Dec. 2005, pp. 138– 143.

[20]   I. Pomeranz and S. M. Reddy, "Scan-BIST based on transition Probabilities for circuits with single and   multiple scan chains," *IEEE Trans.Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 3, pp.591–596, Mar. 2006.

[21]   N. Nicolici and B. Al-Hashimi, "Multiple scan chains for power minimization during test application in sequential circuits," *IEEE Trans. Comput.*, vol. 51, no. 6, pp. 721–734, Jun. 2002.

[22]   V. Iyengar and K. Chakrabarty, "Precedence-based, preemptive, and power-constrained test scheduling for system-on-a-chip," in *Proc. IEEE VLSI Test Symp.*, 2001, pp. 368–374.

[23]   R. Chou, K. Saluja, and V. Agrawal, "Power constraint scheduling of tests," in *Proc. IEEE Int. Conf. VLSI   Des.*, 1994, pp. 271– 274.

[24]   R. Sankaralingam, R. Oruganti, and N. Touba, "Static compaction techniques to control scan vector power dissipation," in *Proc. IEEE VLSI Test Symp.*, 2000, pp. 35–42.

[25]   S. Wang and W. Wei, "A technique to reduce peak current and average   power dissipation in scan designs by limited capture," in *Proc. Asia South Pacific Des. Autom. Conf.*, Jan. 2007, pp. 810–816.

[26]   N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault,   A. Virazel, and H. Wunderlich, "Minimizing   peak power consumption during scan testing: Test pattern modification with X filling heuristics,"   in *Pro Des. Test Integr. Syst. Nanoscale Technol.*, 2006, pp. 359–364.

About the Authors:

Mr. B.Baloji Naik is currently pursuing M.Tech in the stream of Digital Systems and Computer Electronics (DSCE) in the Narasaraopet Engineering College, Narasaraopeta, Guntur (Dist), Andhra Pradesh, India. He is interested to do research in VLSI based design systems.

Dr. V.Venkata Rao has his M.E in Microwave and Radar Engineering from Osmania University Hyderabad. He obtained his PHD from JNTU Hyderabad. Presently he is working as Professor and HOD of ECE (Electronics and Communication Engineering) in Narasaraopeta Engineering College, Narasaraopet, Guntur, Andhra Pradesh, India. He has over 18 years of experience in industry R&D and teaching. He has published and presented 35 research papers in respective international /national journals and conferences .He is a life member of ISTE. His research area includes JPS image processing and embedded systems.