View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by International Journal of Innovative Technology and Research (IJITR)

Ch. Prasanthi* et al.
(IJITR) INTERNATIONAL JOURNAL OF INNOVATIVE TECHNOLOGY AND RESEARCH
Volume No.5, Issue No.3, April – May 2017, 6219-6225.

Realization of Low Transition Based PRPG For Power Optimized Applications

CH. PRASANTHI
PG Scholar, Dept of ECE, Krishna Chaitanya Institute of Technology & Sciences, Markapur, AP, India

BOMMIREDDY AJANTA REDDY
Asst Professor, Dept of ECE, Krishna Chaitanya Institute of Technology & Sciences, Markapur, AP, India

Abstract: **This paper proposes low power pseudo random test pattern generator. This produces the necessary test patterns which are used for running the circuit under test for detecting faults. Power consumption of the circuit under test is measured by switching activity of the inside logic which depends on the randomness of applied stimulus. Power consumption is greatly increased due to the reduction of correlation between the successive vectors of applied stimulus. A modified conventional linear feedback shift register is implemented for reducing power of circuit under test by generating the patterns by reducing the utilization of hard ware. The main intension of producing intermediate patterns is to reduce the conventional activity of primary inputs (PI) that which reduces the switching activities inside the CUT and by this power consumption is reduced without using huge hardware.**

Keywords: **LSFR; BIST; CUT; XILINX;**

I. INTRODUCTION

The main challenges in VLSI are area, power and delay. Power is consumed due the short circuit current flow and charge produce by the switching action in the circuit under test. The usage of portable electronics is increasing rapidly day by day. 200% power is consumed by the device while testing when compared to the device in normal condition. So power optimization is an important challenge for researchers.

The main objective for manufacturing is to reduce the test power consumption and to ensure the device is reliable and high quality products. Evaluation of semi conductor devices is mainly dependent on the design characteristics and the types of the semiconductor technology used. Every new thought and every new device undergo DFT design for test methods.

It is not clear that the compression will be capable to the rapid coping rate with change in technology. The logic built in self test came to existence which makes the design easier to produce test patterns easier and the consumed power can reduced easily. Now a days the DFT came to existence and is gaining high demand for the modern test compression techniques. DFT has the potential to run with high speed and low power for good performance.

The LBIST and data compression became a vital in research and development area, for removing the bottleneck of test data bandwidths. Surveys says that there are many such techniques can be found an example, the conventional scan based test, hybrid techniques, due to high scan based test operations, may consume much power then the critical power then a circuit under test has to consume. Full toggle scan patterns are the example for the techniques that consume more power for testing, particularly for the over mission mode's peak power.

1. Test pattern generator

Test vectors are produced by the test pattern generator that which are given to the circuit under test. This patterns directly influences the fault coverage achieved by the type of pseudo random pattern generator. The most used pattern generators are the LFSR's, these ensure the test patterns are the primitives of polynomials with maximum period. It is not necessary to use primitive polynomials; they are unstable in most cases.

The paper choice is that to generate the polynomials and an LFSR seed which is shown here, by designing a mixed –mode BIST for producing ISCAS benchmarks as increase in the complexity, it makes the need of BIST. Built in self test refers to a chip that can test itself for evaluating the test response. This completely eliminates the complex ATE (automatic test equipment) for testing the design. BIST provides an easy architecture for the tested circuit which is extremely hard from the outside. A numerous BIST architectures has been proposed among them the state of the art technique is a kind that which uses pseudo random pattern generator (PRPG) for producing patterns. These patterns are applied to the circuit under test for generating test patterns. The cellular automata or the linear feedback shift register LFSR are used as pattern generator in the PRPG. These patterns generators provide necessary and required fault coverage. Somehow the patterns are modified for better results. One of the method is designing low power LFSR that produce patterns for circuit under test. The main objective of the paper is to reduce the transition activates of PI which reduces the switching activities inside the CUT, and with this power consumption is reuced.

@ 2013-2017 http://www.ijitr.com All rights Reserved.

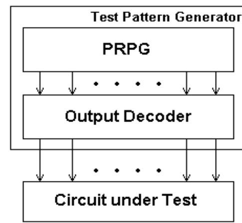


Figure.1. Test-per-clock BIST design

The width of the LFSR is arbitrarily scaled. The scaling effectiveness of the BIST equipment design is standardized by ISCAS benchmarks. The BIST architecture majorly consists of two parts: the pseudo random pattern generator which is obviously a LFSR decoder. The output of the LFSR is transformed the ATPG tool by the combinational circuit. This is the method for designing a test per clock BIST the test patterns are fed parallel to the circuit. Thus the circuit under test gents many inputs from the column matching algorithm, the inputs of the decoder PRPG are the test vectors that which fed to the CUT. The algorithm is designed only for testing the combinational circuits, which means the test patterns generated are insignificant. Thus, the vectors are needed to be reordered by which the patterns are test patterns applied to the circuit under test.

The decoder is an n-input and m-output combinational block, there will be n possibility ways for a column match for each m outputs. Which results in n X m combinations to test for obtaining matching. Thus the algorithm complexity is extremely large which has to be reduced. We have search technique called as the “thorough search” have O(n.m 2 p.s), where p is the PRPG patterns and a is the number of domestic vectors. Thus the algorithm has extended to support Mixed mode BIST. The algorithm is designed only for testing the combinational circuits, which means the test patterns generated are insignificant. Thus, the vectors are needed to be reordered by which the patterns are test patterns applied to the Circuit under test. The general scheme of column matching and mixed mode BIST is as shown in the figure. For mapping the LFSR output and the CUT inputs, the number is taken as m. i.e same bit width as in figure, CUT is the circuit under test, for column matching BIST scheme.

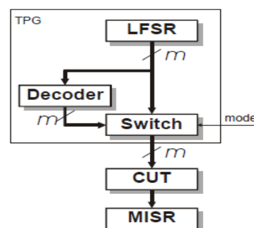


Figure.1.2. Column-matching BIST scheme

The whole mixed-mode column-matching based TPG design process can be summarized as follows:

1. Simulate several (PR) pseudo-random patterns for the CUT and determine the undetected faults (by fault simulation)

2. Compute deterministic test patterns detecting these faults by an ATPG tool

3. Perform the column-matching using the subsequent LFSR pseudo-random patterns and the deterministic tests

4. Synthesize the unmatched decoder outputs using a two-level Boolean minimizer.

II. BIST ARCHITECTURE:

A typical BIST architecture consists of

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

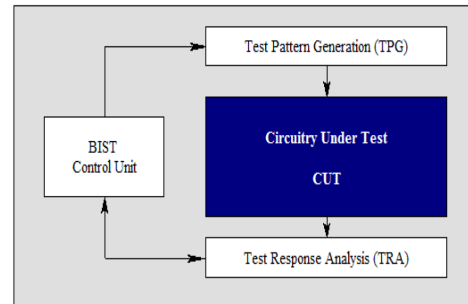


Figure.2. Test Pattern Generator

The TPG- test pattern generator produces the random pattern. The generated test patterns are the pseudo random numbers and in deterministic sequence. A new low power linear feedback shift register (LP-LFSR) for generating random numbers.

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

BIST Control Unit: Control unit is used to control all the operations. Mainly control unit will do configuration of CUT in test mode/Normal mode, feed seed value to LFSR, Control MISR & TRA. It will generate interrupt if an error occurs. You can clear interrupt by interrupt_clear_i signal.

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.

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

 @ 2013-2017 http://www.ijitr.com All rights Reserved.

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

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.

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.

An n-bit PRPG connected with a phase shifter feeding scan chains forms a kernel of the generator producing the actual pseudorandom test patterns. 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.

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. Given the phase shifter structure, one can assess then the amount of scan chains receiving constant values, and thus the expected toggling ratio.

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. 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.

 @ 2013-2017 http://www.ijitr.com All rights Reserved.

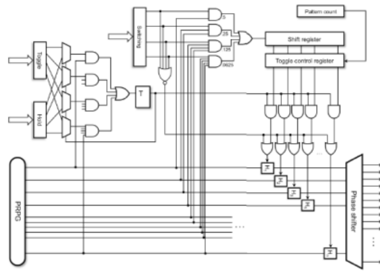


Figure.2.1. Fully operational version of PRESTO.

III. LOW POWER PATTERN GENERATION

One way to avoid the correlation between the successive vectors is to avoid frequent logic transitions from primary inputs. Our approach inserts 3 intermediate vectors between every two successive vectors. The number of transients between the 2 successive vectors generated is equal to the total number of single transitions between the 5 vectors. This reduces the transition activity of the primary inputs which reduces the switching activity inside the CUT. This leads the reduction in power consumption for the design under test.

By modifying the conventional LFSR circuit leads to the new technique of inserting 3 intermediate values in between the logic levels of the conventional flip flop outputs and the outputs as shown in the figure.

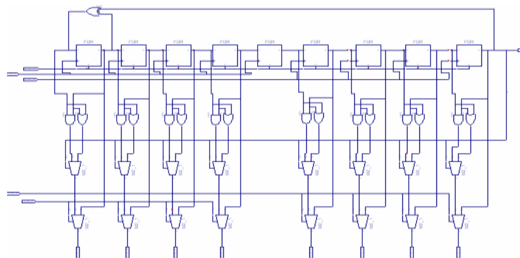


Figure.3. LP-LFSR

In the simulation environment, the outputs of the flip-flops are loaded with the seed vector. The feedback taps are selected pertinent to the characteristic polynomial $x^8 + x + 1$. Only 2 inputs pins, namely test enable and clock are required to activate the generation of the pattern as well as simulation of the design circuit. It is also noteworthy here that the intermediate vectors in addition to aiding in reducing the number of transitions can also empirically assist in detecting faults just as good as the conventional LFSR patterns. Description of the technique to produce low power pattern for BIST The following is a description of a low power test pattern generation technique as depicted in the 9-bit LFSR based schematic in Figure 4.1. Verilog based test bench as shown in Appendix B is used in assigning the initial output states (0100 1011) of the 9-bit LFSR. The feedback taps are designed for maximal length LFSR generating all zeros and all one's as well. The first step is to generate T1, the first vector by enabling (clocking) the first 4-bits of the LFSR and disabling (not clocking) the last 4 bits. This Shifts the first 4 bits to the right by one bit. The feedback bits of the LFSR are the outputs of the 8th and the first flip-flop. The output of the 8th flip-flop is 1 and the output of the first flip-flop is 0. The exclusive-or of the 8th -flip-flop (logic 1 in this case) and the first flip-flop(logic 0 in this case) is input (1 EXOR 0 = 1 into the first D flip-flop. The new pattern in the first four bits of the LFSR is 1010. Note that the shaded register is clocked along with the first 4 bits of the LFSR. So the input of the shaded flip-flop is the output of the 4th flip-flop which in this case is 0. Also note that prior to the first clock, the input of the shaded register was the seed value of the 4th flip-flop at the output of the 4th flip-flop which in this case is 0. So after the first clock this value of 0 will now appear at the output of the shaded flip-flop. In other words the value of the 4th output is stored in this shaded register and is used in the next few steps. The first 4 shifted bits of the LFSR and the last 4 un-shifted bits (i.e. the seed value) are propagated as T1 (1010 1011) to the final outputs. Next few steps involve generating the 3 intermediate patterns from T1. These patterns are defined as Ta, Tb and Tc shown in below flow.

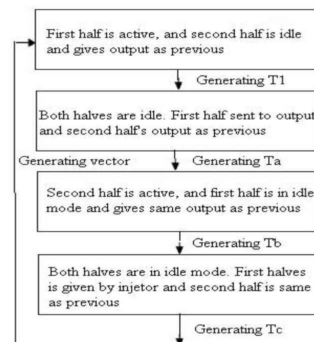


Figure.3.1. Proposed algorithm for low power LFSR

Ta is generated by maintaining (disabling the clock to the first 4 bits) the first four bits of the LFSR outputs (as is from T1) as the final first four low power outputs 1010. Note that the clock to the last four bits of the LFSR is also disabled.

The last four bits however are the outputs from the injector circuits. The injector circuit compares the next value (the input of the D-flip-flop) with the current value (the output of the D-flip-flop). According to T1, the outputs (current values) of the last 4 bits of the LFSR are 1011. The next values are the values at the inputs of the D-flip-flops which in this case are 0101**.** **C**ompare the current values (1011) bit by bit with the next values (0101). If the values bit by bit are not the same then

 @ 2013-2017 http://www.ijitr.com All rights Reserved.

use the random generator feedback R (in this case is logic 1) as the bit value as shown in the schematic above. If however both values bit by bit are the same then propagate that bit value to output as opposed to the R bit. This bit by bit comparison gives us the last four bits of Ta to be 1111. Therefore Ta = 1010 1111. Next step is to generate Tb. Shift the last 4 flip-flops to the right one bit but do not shift the first 4 flip-flops to the right. The clock to the first 4 bits plus the shaded flip-flop is disabled. The clock to the last 4 bits is enabled. Propagate the outputs of the flip-flops of the entire LFSR as opposed to the outputs of the injection circuit to the outputs (low power). The injection circuits are disabled. As in Ta, maintain the first four LFSR outputs (1010) as the low power outputs. Again from Ta, the inputs of the last four D flip-flops from the previous step (generating Ta) are 0101. Also note that the output of the shaded register is 0 from the previous step (generating Ta). Therefore the input of the 5th flip-flop is a 0. The outputs of the last 4 flip-flops are 0101 resulting in Tb = 1010 0101. The 3rd intermediate vector Tc is generated via disabling the clock to the entire LFSR. Propagate the first 4 outputs from the injection circuit as the first 4 low power outputs and maintain the last 4 low power outputs the same as Tb. Generating injection circuit outputs for Tc is conceptually the same as explained above in generating Ta. Current values (the outputs of the flip-flops) of the first four flip-flops are compared with the next values (the inputs of the flip-flops) of the flip-flops. The feedback from the 8th flip-flop is 1 (please see generating Tb). Therefore the logical feed forward value of R is 1. The feedback value from the first flip-flop is also 1 as per the current values above. The exclusive or of two ones is a 0. Therefore the input to the first flip-flop is a 0 which is also the next state of the first flip-flop. Hence the next values are 0 for the first flip-flop and 101 for the 2nd, 3rd and 4th flip-flop respectively. The next values are 0101. The first four outputs from the injection circuit are 1111. The last 4 outputs are the same as Tb which are 0101 resulting in the 3rd and final intermediate vector Tc = 1111 0101. Generating T2 is quite similar to generating T1. As in Tc the outputs of the last four LFSR flops are 0101. The outputs of the first 4 flip-flops of the LFSR are the current values which are 1010. Therefore the seed vector for generating T2 is 1010 0101. Shift the first four bits of the LFSR plus the shaded flip-flop. Do not clock the last four flip-flops. Propagate the outputs of the entire LFSR to the final low power outputs. The output of the 8th flip-flop from the previous step (generating Tc) is a 1 and the output of the first flip-flop from the previous step (generating Tc) is also a 1. The exclusive or of the output of the 8th flip-flop and the first flip-flop is 0. Therefore the input to the first flip-flop will be a 0. The inputs to the 2nd, 3rd, 4th and the shaded flip-flops are 1010. These are also the current values from the previous step (generating Tc). Shifting the first four flip-flops of the LFSR to the right by one bit results in 0101 as the outputs of the first four flip-flops. Therefore T2 generated is 0101 0101.

C17 bench mark circuit:

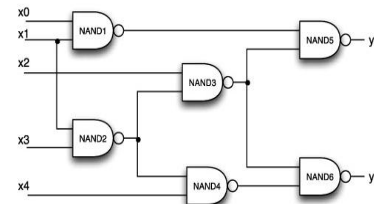
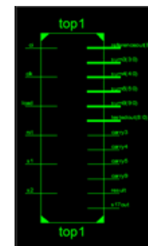


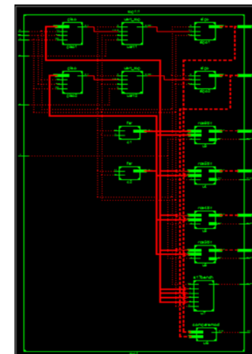
Figure.3.2. C17 Benchmark Circuit

IV. RESULT

RTL SCHEMATIC DIAGRAM:



INTERNAL RTL SCHEMATIC:



Simulation Results of BIST using Low Power LFSR:

 @ 2013-2017 http://www.ijitr.com All rights Reserved.

COMPARISON RESULTS:

Proposed Design

Device Utilization Summary (estimated values)			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	56	4656	1%
Number of Slice Flip Flops	76	9312	0%
Number of 4 input LUTs	81	9312	0%
Number of bonded IOBs	55	232	23%
Number of GCLKs	1	24	4%

Delay: 3.710ns

Power : (81*0.076)/9312= 0.0006612 = 66.12 micro watts

Existing Design

Device Utilization Summary (estimated values)			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	57	4656	1%
Number of Slice Flip Flops	81	9312	0%
Number of 4 input LUTs	88	9312	0%
Number of bonded IOBs	61	232	26%
Number of GCLKs	1	24	4%

Delay : 3.710ns

Power : (88*0.076)/9312 = 71.821 micro watts

Comparison table

	no. of LUT's	power (micro watts)	delay (nano sec)
existing	88	71.821	3.71
proposed	81	66.12	3.71

By using low power LFSR, the area utilization is decreased when compared to the PRESTO method then the number of look up tables is decreased to 81 than existing method of 88 look up tables. Hence the Power is also decreased to 66.12 micro watts from existing method of 71.821 micro watts power and memory utilization is 187084kb with compared to memory of existing PRESTO 187724kb and the delay is 3.710ns for existing prpg and 3.710ns for lowpower LFSR.

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 (i.e. Translate, Map and Place and Route) of the code is carried out on Xilinx - Project Navigator, ISE 12.3i suite. The power reports shows that the proposed low power LFSR consumes less power during testing by taking the benchmark circuit C17. 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",2014.

[2]. Michael L.Bushnell, Vishwani D.Agawal," Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits," Kluwer Academic Publishers, 2000.

[3]. Mohammad Tehranipoor, Mehrdad Nourani, Nisar Ahmed," Low- Transition LFSR for BIST-Based Applications," 14th Asian Test Symposium, pp. 138- 143, 18-21 Dec. 2005.

[4]. F. Corno, P. Prinetto, M. Rebaudengo, M. Sonza Reorda," A Test Pattern Generation methodology for low power onsumption," pp.1-5, 2008.

[5] P.Glard et al "Survey 0 F Low-Power Testing Of Vlsi Circuits"IEEE Design & Test Of Computers, vol. 19, no. 3. (2002), pp. 80-90.

[6]. Dr.K.Gunavathi, Mr. K. ParamasivaM, Ms.P.Subashini Lavanya, M.Umamageswaran," A novel BIST TPG for testing of VLSI circuits", IEEE International Conference on Industrial and Information Systems, pp. 8 - 11, August 2006.

[7] Y.-C. Lin, F. Lu, K. Yang, and K.-T. Cheng, "Constraint extraction for pseudo-functional scan-based delay testing," in Proc. Asia South Pacific Design Autom. Conf., 2005, pp. 166–171.

[8] M.E. Amyeen, A. Jayalakshmi, S. Venkataraman, S.V. Pathy, and E.C. Tan,"Logic BIST Silicon Debug and

 @ 2013-2017 http://www.ijitr.com All rights Reserved.

Volume Diagnosis Methodology," Proc. Int'lTest Conf., pp. 1-10, 2011.

[9] M. Abramovici, M.A. Breuer, and A.D. Friedman, Digital Systems Testing and Testable Design. IEEE Press, 1995.

[10] P. Girard, "Survey of Low-Power Testing of VLSI Circuits," IEEE Design and Test of Computers, vol. 19, no. 3, pp. 80-90, May/June 2002.

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

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

[13] 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.

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

[15] 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.

AUTHOR's PROFILE

Ch.Prasanthi is a student pursuing M.Tech(ECE) in Krishna Chaitanya Institute of Technology and Sciences, Markapur, Prakasam district, India.

Bommireddy Ajanta Reddy obtained his Bachelor's degree from Prakasam engineering college, Kandukur and Masters Degree from Lakireddy Balireddy College of engineering, Mylavaram(systems and signal processing). Areas of interest are Image processing, signal processing, datamining and computer vision. He has published my research papers in various national and international journals. Presently working as Asst Professor of ECE Department, KITS, Markapur, A.P, India.

 @ 2013-2017 http://www.ijitr.com All rights Reserved.