International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 1, January 2014)

Delay Adjusted Low Power Pattern Generator for BIST

B.Syed Ibrahim[1], M.Rajendiran[2], D.Sankareshwaran[3]
[1]*Sr.Asst Professor,Department of ECE,Chettinad College of Engineering & Technology,Karur, India*
[2]*Asst Professor,Department of ECE,Chettinad College of Engineering & Technology,Karur, India*
[3]*Asst Prof,Department of ECE,Chettinad College of Engineering & Technology,Karur, India*

***Abstract*— Most of the pattern generators used for BIST are sequential circuits consisting of flip-flops. LFSR is one of the most widely used pattern generators comprising of a series of flip-flops with additional xor taps. The use of flip-flops in circuits increases circuit complexity, propagation delay, power consumption and requirement of additional external signals such as reset and clock. Here in this paper we propose a new combinational architecture for pattern generator which constitutes only few logic gates and not flip-flops. Thus this method aids in providing an efficient and effective way in greatly reducing power consumption, delay and complexity.**

Keywords-power consumption;

I. INTRODUCTION

As the VLSI designs increase in integration density, associated modules becomes inaccessible and testing of the chip becomes more challenging. VLSI processors are still required for higher speed and less-power consumption. The test industry is facing with varied problems such as increase in test time and test volume when external ATE is employed for testing. Power consumption in test mode is usually higher than that in normal mode, and it has been shown that power consumption during test mode is as high as 200% of the power consumed in the normal mode [1].To deal with this, in-system test generation and test application were introduced through built in self test architectures by the testing community. Test generation hardware and embedded algorithms reduce the need for test access for system-wide built-in self-test (BIST), which is a preferred DFT approach.

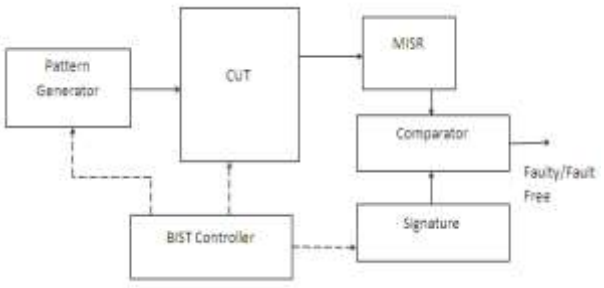

Figure 1. A simple architectural representation of BIST

A general form of BIST structure is given in Figure.1. Test pattern generation and output response analyzer are the two important functions that help in self testing of circuit under test (CUT) by BIST. Test pattern generator produces sequence of patterns to be applied to CUT for testing. Output response analyzer (also known as signature analyzer) compares the encoded output of CUT with expected output and indicates whether the circuit is faulty or fault free. Input isolation circuitry prevents the real time system inputs to enter the CUT, when the system is in test mode. Test controller takes care of initialization of the modules before test vectors are applied as well as before output responses are given for compaction.

The test generation module that has been widely used in BIST design is the linear feedback shift register (LFSR). Linear Feedback Shift Registers (LFSR) are widely used in BCH encoders and CRC operations [2]–[4]. LFSRs are also used in conventional Design for Test (DFT) and Built-in Self-Test (BIST) [5]. LFSRs are used to carry out response compression in BIST, while for the DFT, it is a source of pseudorandom binary test sequences.

LFSRs can be implemented in hardware, and this makes them useful in applications that require very fast generation of a pseudo-random sequence, such as direct-sequence spread spectrum radio. LFSRs have also been used for generating an approximation of white noise in various programmable sound generators. The repeating sequence of states of an LFSR allows it to be used as a clock divider, or as a counter when a non-binary sequence is acceptable as is often the case where computer index or framing locations need to be machine-readable. LFSR counters have simpler feedback logic than natural binary counters or Gray code counters, and therefore can operate at higher clock rates. LFSRs have long been used as pseudo-random number generators for use in stream ciphers (especially in military cryptography), due to the ease of construction from simple electromechanical or electronic circuits, long periods, and very uniformly distributed output streams. However, an LFSR is a linear system, leading to fairly easy cryptanalysis. A sequential LFSR circuit cannot meet the speed requirement when high-speed data transmission is required.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 1, January 2014)

Because of this limitation, parallel architectures must be employed in high-speed applications such as optical communication systems where throughput of several Gigabit/s is required [6].

This paper presents a parallel high speed parallel architecture for pattern generation where the use of flip-flop registers are completely eliminated thereby decreasing power consumption [7], area overhead and delay that are associated with sequential circuits. This is because the proposed circuit doesn't use pipeline registers that are triggered by a system clock and consumes much power.

II. Types Of Pattern Generation

A. *Pseudo-Exhaustive*

Pseudo exhaustive testing partitions the CUT in to several smaller sub circuits and tests each of them exhaustively. All detectable faults within the sub circuits can be detected. However, such a method requires extra design effort to partition the circuits and deliver the test patterns and test response.

B. *Pseudorandom*

Pseudorandom testing applies a certain length of test patterns with certain randomness property. The sequences of test patterns are in a deterministic order. The fault coverage is determined by the test length and the contents of the pattern.

C. *Weighted Pseudorandom*

Weighted pseudorandom testing applied pseudorandom patterns with certain 0s and 1s distribution to handle the random pattern resistant faults undetectable by the pseudorandom testing. It can effectively shorten the test length.

III. Existing System

To observe the

A. *Conventional LFSR*

A linear feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state. The most commonly used linear function of single bits is XOR. Thus, an LFSR is most often a shift register whose input bit is driven by the exclusive-or (XOR) of some bits of the overall shift register value thus forming a feedback loop.

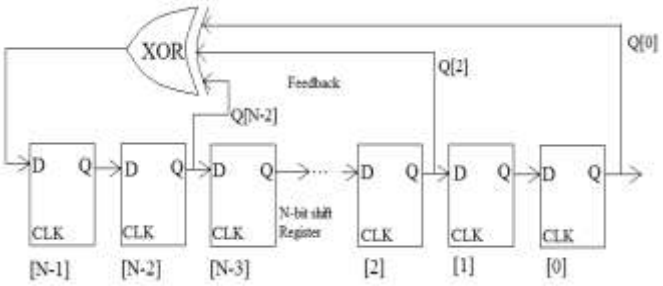

Figure 2. General structure of LFSR

Figure.2 shows a general structure of conventional LFSR. The initial value of the LFSR is called the seed, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its current (or previous) state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random and which has a very long cycle.

It will produce a pseudorandom sequence of length 2^n-1 states (where n is the number of stages) if the LFSR is of maximal length. It does not generate all zeros pattern. If so, it gets stuck up in that state. The sequence will then repeat from the initial state for as long as the LFSR is clocked.

One of the reasons is that an LFSR is more area efficient than a counter, requiring less combinational logic per flip-flop. The whole structure of LFSR can be reduced to a single polynomial equation. The connections to the feedback loop indicate the placeholder names which are powers of X. One end is always X^n. The others are X^k if there is an XOR connection at k. The polynomial which generates all the 2^n-1 patterns except all zeros pattern is called primitive polynomial. Others are non-primitive and they are not of much use as they have only shorter sequences.

The demerits of conventional LFSR are more power consumption and area overhead due to the presence of flip-flops since it requires n number of flip-flops for n bit pattern generation.

B. *Maximal Length LFSR*

A conventional LFSR can be modified quite easily to generate the *"*all zero*"* state. The resulting sequence is then called a De Bruijn code. The normal LFSR circuit cannot generate the "all zero" state or the trivial state, because it will get stuck in this state forever.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 1, January 2014)

The idea is to insert a '1' into the circuit by NORing the outputs of flip- flops when they are stuck up in all zeros state. Since maximal length LFSR also has the same architecture as the conventional LFSR but with one additional NOR gate to pull the circuit to get out of the stuck up. Hence the detriments of conventional LFSR also apply to Maximal length LFSR except the missing of all zeros pattern.

C. Leap Forward LFSR

Leap-forward LFSR method utilizes only one LFSR and shifts out several bits. All shifts are performed in one clock cycle; i.e., multiple steps are done in the recurrence equation [8]. This method is based on the observation that an LFSR is a linear system and the register state can be written in vector format:

$$q(i+1) = A \cdot q(i) \qquad (1)$$

In this equation, q(i +1) and q(i) are the content of shift register at (i+1)th and ith steps, and A is the transition matrix. Leap-forward LFSR method achieves its goal by utilizing extra combinational circuit instead of duplicated LFSRs. For small *k*, the combinational circuit is not very complex. It is ideal for FPGA devices since it balances register and combinational circuitry and fully utilizes the resource of logic cells. However, for very large k, the XOR structure grows very large and becomes the dominant factor.

IV. PROPOSED SYSTEM

We adopted a new technique for generating test patterns named technique. The previous test pattern generating methods which we have dealt with consists of two or more flip-flops thus consuming power greatly. The proposed circuit consists of only logic gates and not flip-flops. Hence the power consumption is reduced with this technique. It comprises only less number of transistors which in turn reduces the area overhead. Moreover it is a high speed circuit and operates in GHz. The parallel architecture of the proposed circuit also contributes to the increase in speed of pattern generation.

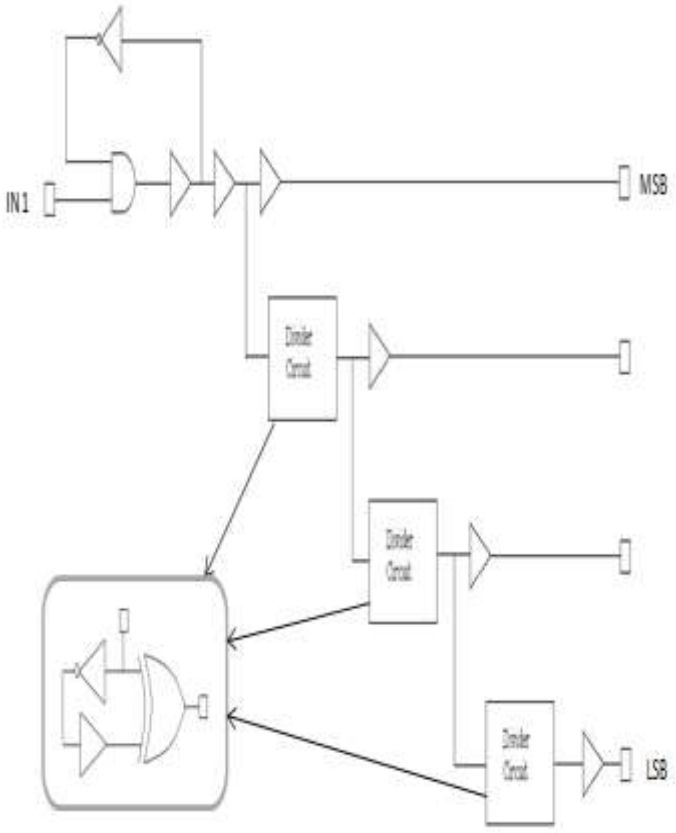

Figure 3. Proposed TPG

V. EXPERIMENTAL RESULTS

Experimentation was performed on conventional, maximal length and leap forward LFSRs and also on the proposed method. Power consumption is analyzed by using. Running time is calculated with the help of the tool Modelsim Altera 6.5b from Mentor Graphics. The synthesis report on total number of gates and flip-flops are obtained from the tool Xilinx. The proposed circuit is simulated in Microwind and Modelsim and the area overhead is evaluated from its layout.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)) Volume 2, Issue 1, January 2014)

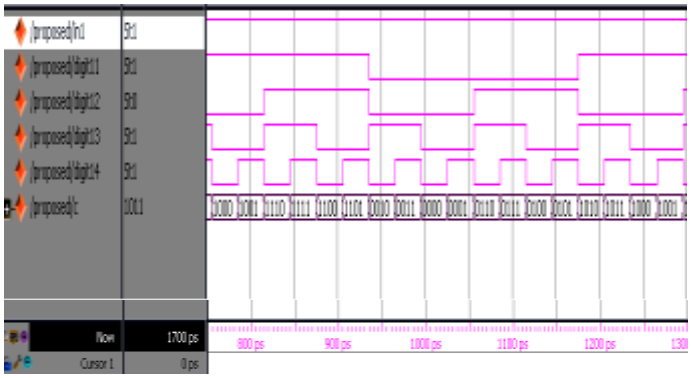

Figure 4. Simulation Result of Figure 3

The analysis and calculation is carried out for 4 bit for the following TPGs listed in Table.1

TABLE I. PERFORMANCE COMPARISON

Type / Parameters	Conventional LFSR	Max Length LFSR	Leap LFSR	Proposed Method DALP
Running Time (ns)	630	694	320	7
Number of FFs used	4	4	4	none
Number of gates used	21	23	23	17
Power Consumption (mW)	15.2	17.4	19.7	4
Number of patterns generated	15	16	15	16

VI. CONCLUSION

The paper proposed a new combinational logic design for pattern generation in BIST thereby eliminating the use of flip-flops which consumes most of the power during testing. The reduction in power consumption is accomplished by the use of invertors and buffers which consume very little amount of power.

The experimental results guarantee up to % reduction of power consumption,% reduction of running time when compared with that of the sequential LFSR circuit. In addition, the parallel architecture of the proposed LTPG ensures minimized propagation delay and equal test pattern length in comparison to the previous pattern generators.

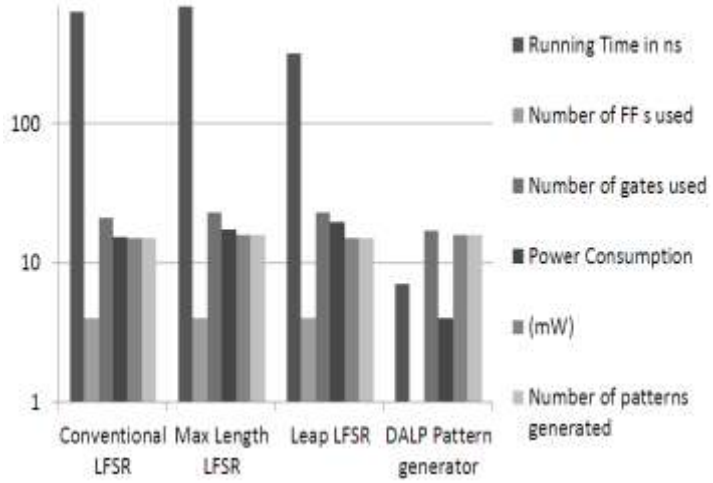

Figure 5. Performance chart

REFERENCES

[1] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," Proc. 11th IEEE VLSI Test Symp., Atlantic City, New Jersy, pp.4-9, Apr. 1993.

[2] T. V. Ramabadran and S. S. Gaitonde, "A tutorial on CRC computations," *IEEE Micro.*, Aug. 1988.

[3] R. E. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley,1984.

[4] W. W. Peterson and D. T. Brown, "Cyclic codesfor error detection," *Proc. IRE*, vol. 49, pp. 228–235, Jan. 1961.

[5] N. Oh, R. Kapur, and T. W. Williams, "Fast speed computation for reseeding shift register in test pattern compression," *IEEE ICCAD*, pp. 76–81, 2002.

[6] Manohar Ayinala and Keshab K. Parhi, "High-Speed Parallel Architectures for Linear Feedback Shift Registers," IEEE Transactions On Signal Processing, vol. 59,pp. 4459-4469, Sep. 2011.

[7] Tomoaki Sato Rena Sakuma, Daisuke Miyamori, and Masa-aki Fukase, "Waved-PRNG for a Wave-Pipelining Test Circuit," 12th NASA Symposium on VLSI Design, Oct. 2005.

[8] Pong P. Chu and Robert E. Jones, "Design Techniques of FPGA Based Random Number Generator," Nasa Grant NAG3-2040.