International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 7 (2015) pp. 16797-16813

http://www.ripublication.com

Efficient Memory Built in Self Test Address Generator Implementation

V. Suresh Kumar[1] and R. Manimegalai[2]

[1]Dept. of EIE, Valliammai Engineering College, Anna University, Chennai – 603 203, India.
[2]Dept. of Computer Science and Engineering, Park College of Engineering. and Technology, Anna University, Coimbatore - 641 659, India.
[1]suresh_vekumar@yahoo.com,[2]mmegalai@yahoo.com

Abstract

Now a days, Very Large Scale Integration (VLSI) Technology is becoming more popular in the fabrication. Field Programmable Gate Arrays (FPGA) are used to implement the complex logic operations in digital applications. Day by day the integration degree in the VLSI technology increases and due to this there is a rapid development in packaging techniques. Every VLSI application goes under testing phase. This paper describes the power efficient Memory Built-In Self Test (MBIST) Address generator using combinations of two modified Linear Feedback Shift Register (LFSR). The proposed address generator is suitable for MBIST that adopts Zero-One method for memory testing is used. This MBIST address generator module is synthesized and simulated using verilog-HDL in Xilinx ISE version 14.1. The switching activity exhibited by the proposed address generator is more efficient when compared with existing methods

Keywords — MBIST, LFSR, Power consumption, Switching activity, Zero-one memory test.

I. INTRODUCTION

Complementary Metal Oxide Semiconductor (CMOS) is used to design the VLSI circuits. CMOS is the prominent source of power dissipation in the VLSI circuits. Complex logic operations in digital applications are performed using the Field Programmable Arrays. It contains the logic modules. These logic modules communicate through interconnected lines and wires. There are two types of FPGA

platforms.

1) One time programmable FPGAs
   E.g. Antifuse type.

2) Reprogrammable FPGAs
   E.g. SRAM based FPGA

While manufacturing the VLSI devices, it undergoes through various phase like as follows.

1) Testing Phase :
VLSI circuits undergo this phase to determine whether the circuit manufactured is working properly or not. Memory Built In Self Test (MBIST) is popular method for testing the memory of VLSI circuits [1], [2], [3]. VLSI circuits are getting more and more complicated. This complex integration is built on System On chip (SoC).

2) Verification Phase :
In this phase, it is checked that whether refinement in the design process are consistent with the circuit specification.

3) Validation Phase :
In this phase, it is checked that whether the manufactured circuit is fulfilling as per the requirement to which it was designed.

This paper mainly concentrates on testing phase. As the VLSI circuits/devices are made up of semiconductor material, the power consumption is important factor to be considered. It should be as low as possible then only it can be considered power efficient. VLSI reprogrammable FPGA devices consist of Logic elements and Programmable switches. There is power consumption in the testing phase of VLSI devices. While testing the VLSI devices memory, multiple memories can be tested in parallel manner and some memories are tested in normal mode [4]. Ideally, Power consumption during testing should be twice the normal mode. If the power consumption during testing exceeds the power constraint of the chip then there is a possibility of memory and chip damage [5]. So, test power should be less than the power constraint of the chip.

The motive of this paper is to lower the power consumption during testing. From various researches, it is expected that the there will be 94% area of System On Chip (SoC) will be occupied by embedded memories by 2014. This leads to built in testing instead of external memory testing and the motive goes towards the Memory Built In Self Test (MBIST). MBIST address generator [6] mainly responsible for fault detection. Also it lowers the area overhead.

MBIST is the test circuitry that automatically tests the memory by applying certain test pattern and observes the system performance [7]. The block diagram of MBIST is shown in Fig. 1.

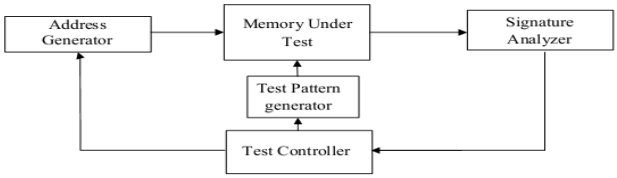

Fig. 1 Memory Built In Self Test (MBIST) block diagram

Test pattern generator is used to generate the pattern to test the memory under test [8]. It consists of test vectors that are applied to memory under test. After the memory testing, output is given to signature analyzer that analyses the memory under test output and decides whether memory under test is faulty or fault-free and gives feedback to the Test controller. The test controller executes the testing operation with memory under test and instructs the test or address pattern generator for next pattern. The input signal from the Test controller initiates the self test sequence needed for testing.

LFSR is most commonly used address generator in MBIST applications. LFSR are more area efficient when compared to that of implemented using counters. LFSR can be implemented using two types of feedback that are external feedback or internal feedback [9]. Both types consist of same number of flip flops and XOR gates. Internal Feedback LFSR have XOR gate between any two flip-flops. So it requires higher operating frequency than External Feedback LFSR. In External Feedback LFSR feedback is linear via XOR gates. When more symmetric layout is required, external feedback LFSR is used and for high performance design, internal feedback LFSR is used.

There are many techniques developed for the memory testing as follows.

1) Zero-one
2) Walking 1/0
3) March sequences
4) Checker board

The Zero-One MBIST technique has high switching activities in address generator bus. So it increases power consumption of the testing phase. To avoid this issue, low power address generator should be used. Generally, Linear Feedback Shift Register (LFSR) is used as an address generator that lowers the hardware and ultimately the switching activities are reduced and power consumption is lowered [10], [18]. So this paper mainly constitutes the MBIST address generator with reduced switching activities.

The rest of the paper is organized as follows. Section II reviews some existing works that are related to the proposed work. Section III discusses the motivation for the proposed work along with switching activity and the methods to reduce it. Section IV explains each block of the proposed address generator and its analysis. Section V reports the simulation result and the comparison and discussion of the proposed method with other existing methods and the work concludes in section VI.

II. RELATED WORK

A number of researches are available in the literature for the MBIST Address Generator on FPGA platform. Below are the reviews of recent works from the researches.

Nilanjan Mukherjee et al. [11] presented a built-in self-test (BIST)-based scheme for fault diagnosis that can be used to identify permanent failures in embedded read-only memories. Their approach offers a simple test flow and does not require intensive interactions between a BIST controller and a tester. The scheme rests on partitioning of rows and columns of the memory array by employing low cost test logic. It was designed to meet requirements of at-speed test thus enabling detection of timing defects. Experimental results confirm high diagnostic accuracy of their presented scheme and its time efficiency.

Chun-Lung Hsu et al. [12] presented a built-in self-test (BIST) design for fault detection and fault diagnosis of static-RAM (SRAM)-based field-programmable gate arrays (FPGAs). Their approach presented an FPGA BIST structure can test both the interconnect resources [wire channels and programmable switches (PSs)] and lookup tables (LUTs) in the configurable logic blocks (CLBs). The test pattern generator and output response analyzer are configured by existing CLBs in FPGAs; thus, no extra area over-head were needed for their designed BIST structure. The target fault detection/diagnosis of their designed BIST structure were open/short and delay faults in the wire channels, stuck on/off faults in PSs, and stuck-at-0/1 faults in LUTs. The applications on XC4000-series FPGAs show that 100% fault coverage of their designed FPGA BIST structure was obtained. Additionally, the test results revealed that good performance in fault detection and fault diagnosis on both interconnect resources and CLBs can be achieved at levels similar to those required in previous works.

Mohamed H. Abu-Rahma et al. [13] presented an architecture that significantly reduces the array switching power for SRAM. Their approach combines built-in self-test and digitally controlled delay elements to reduce the wordline pulse width for memories while ensuring correct read operations, hence reducing the switching power.

Abdallatif S. Abu-Issa et al. [14] presented a low-transition linear feedback shift register (LFSR) that was based on some observations about the output sequence of a conventional LFSR. Their design, called bit-swapping LFSR (BS-LFSR), was composed of an LFSR and a 2×1 multiplexer. When used to generate test patterns for scan-based built-in self-tests, it reduced the number of transitions that occur at the scan-chain input during scan shift operation by 50% when compared to those patterns produced by a conventional LFSR. Hence, it reduced the overall switching activity in the circuit under test during test applications. The BS-LFSR was combined with a scan-chain-ordering algorithm that orders the cells in a way that reduced the average and peak power (scan and capture) in the test cycle or while scanning out a response to a signature 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 on ISCAS'89 benchmark circuits show up to 65% and 55% reductions in average and peak power, respectively.

Memory Built-In Self-Test (MBIST) has become a standard industrial practice.

Its quality is mainly determined by its fault detection capability in relationship to the area overhead. The MBIST Address Generator (AG) is largely responsible for the fault detection capability, and has a significant contribution to the area overhead. Ad J. van de Goor et al. [6] analyzed the properties and implementation aspects of several AGs. In addition, they presented a systematic, high-speed, low-power and low-overhead implementation, based on an Up-counter and a set of multiplexers.

R.Muthammal et al. [9] presented a low power efficient Built in Self Test (BIST) with Test Pattern Generation (TPG) technique, which reduces power dissipation during testing. In general, the correlations between the consecutive test patterns are higher during normal mode than during testing mode. Their approach uses the concept of reducing the transitions in the test patterns generated by conventional Linear Feedback Shift Register (LFSR). The transitions are reduced by increasing the correlation between the successive bits in the test pattern, which is done with the help of their modified LFSR. This approach eliminates the need for an external tester. The simulation result showed that the power dissipated during testing is reduced in modified LFSR than in conventional LFSR.

T. Nandha Kumar et al. [15] presented a method for generating configurations for application-dependent testing of a SRAM-based FPGA interconnect. Their method connects an activating input to multiple nets, thus generating activating test vectors for detecting stuck-at, open, and bridging faults. Their arrangement permitted a reduction in the number of redundant configurations, thus also achieving a reduction in test time for application-dependent testing at full fault coverage. As the underlying solution requires an exponential complexity, a heuristic algorithm that is polynomial and greedy in nature (based on sorting) is used for net selection in the configuration generation process. Extensive logic-based simulation results were provided for ISCAS89 sequential benchmark designs implemented on Xilinx Virtex4 FPGAs; their results presented a considerable reduction in the number of test configurations compared with methods found in the technical literature (on average, a reduction of 49.5 percent)

III. RESEARCH METHODOLOGY

Out of the number of memory testing techniques mentioned in above sections couple of them focus on reducing the test power which plays an important role in evaluating the effectiveness of the test. VLSI circuits are based on Complementary Metal Oxide Semiconductor (CMOS) technology in which dynamic power is the dominant source of power dissipation. The dynamic power is related to the number of Switching Activities (SA) during testing. For this reason, reducing the switching activity during testing is a basic technique for power saving. Most of the time in electronic gadgets such as mobile phones and digital cameras, the embedded memory is tested for stuck at faults since these faults are the most common specially during online testing. Thus, the test power has to be reduced since these small devices have low power envelop and they may be damaged if the power consumed exceeds the power constraint. Since dynamic power is the major source of power dissipation, in this paper the reduction in number of Switching Activities (SA) in the address generator of the testing circuit is

targeted for power saving

Motive of this work is to design a power efficient MBIST Address Generator which shows effective improvement in the power consumption compared to the existing techniques. Major work covered in this article is designing an MBIST address generator with reduced SA, so that the total power is reduced. To do so the Zero-one MBIST technique to test the memory for stuck at fault will be utilized. Zero-One testing algorithm contains high switching activity in address bus that will increase the testing power consumption. And to overcome this problem the sequence of the addresses to be accessed is not important and a low power address generator providing appropriate address transitions has to be selected. Usually, the Linear Feedback Shift Register (LFSR) is used as an address generator due to its little overhead in the hardware area but SA is its concern. Hence in this work a power efficient Address Generator module using modified LFSR in Verilog is designed and this module is implemented on FPGA platform and simulated in Xilinx ISE 14.1, which will reduce the Switching Activity (SA) compared to the existing Address Generators.

IV. PROPOSED METHOD

The basic idea behind the design of low power address generator for MBIST is to reduce the bit transition between two successive address patterns that are generated by the address generators at each clock cycle during the test mode operation of the circuit under test and thereby contributing efficiently for the reduction of switching activity in the MBIST. Let us consider a *n*-bit address generator with *n*-bits. A *n*-stage conventional LFSR when used as address generator exhibits high transitions between the successive address patterns. In the proposed address generation technique for reducing the switching activity we have adopted two LFSR's the LFSR's are provided with a two separate clock signals with two different frequencies. In the proposed address generator *n-2* bits of the n-bit address are generated by the modified complete LFSR architecture and the last two LSB bits of the address pattern are generated by 2-bit complete LFSR. The schematic block diagram of the proposed address generator for MBIST is shown in Fig. 2.

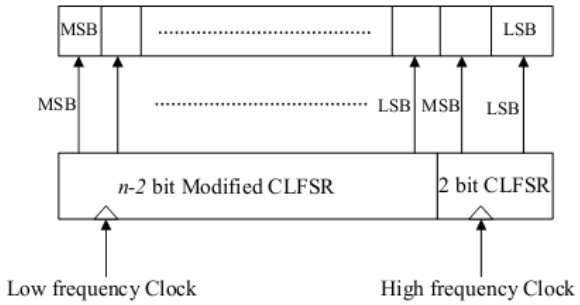

Fig. 2 Block schematic of the proposed low power address generator

The proposed address generator includes 1) *n-2* bit modified CLFSR 2) 2-bit CLFSR and 3) A Dual clocking scheme. Let us discuss these in the below sections in detail.

A. *n-2 Bit Modified CLFSR*

The proposed address generator adopts a *n-2* bit modified CLFSR which is of a external type. The modified *n-2* bit CLFSR consists of positive edge triggered *n-2* D-flip flops arranged as shown in Fig.3. Based on the primitive polynomial for the selected *n-2* value the taps b_0 to b_{n-2} values switches between '0' and '1', if its value is '1' then fed to the reconfigured full adder as input if not its left without connection. The output of the full adder is then fed as feedback to the first flip flop. Conventional LFSR stuck at all zero states, hence to overcome this, all zero state detector and a feedback of '1' generation logic are needed. In the modified CLFSR design this logic can be replaced by feeding all the individual flip flop outputs to a NOR gate and then the output to the feedback as shown in Fig 3.

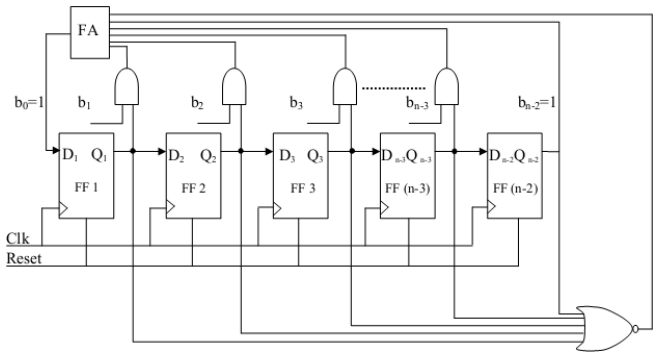

Fig. 3 Architecture for Proposed *n-2* bit Modified CLFSR

The working of the modified CLFSR can be analyzed clearly by considering a primitive polynomial $P(x)$ expressed as follows:

$$P(x) = \sum_{i=0}^{n-2} b_i * x^i, \tag{1}$$

The value of 'b_i'is '0'if for the selected *n-2* value the primitive polynomial does not include a corresponding 'x_i' value. For example if the selected value of *n-2* =4, then the corresponding primitive polynomial is given as (refer TABLE I),

$$P(4) = x^4 + x + 1$$
$$= 1x^4 + 0x^3 + 0x^2 + 1x^1 + 1x^0 \tag{2}$$

Comparing (1) and (2),

$b_0 = 1, b_1 = 1, b_2 = 0, b_3 = 0$ and $b_4 = 1$

By feeding the above 'b' values, the proposed Complete LFSR can be made to generate all possible 4-bit combinations including '0000' states randomly. The random pattern thus generated contains a high bit transition and hence the switching activity is more thereby increasing the dynamic power. This can be reduced by using 2 separate clocks with a signal output as shown in Fig. 3 and thus the output bit pattern generated by this proposed architecture reduces the switching activity.

TABLE I B-VALUE FOR PRIMITIVE POLYNOMIAL OF DEGREE *'n-2'*

n-2	***B***
2,3,4,6,7,15,22	b_{n-2}=1, b_1=1, b_0=1
5,11,21,29	b_{n-2}=1, b_2=1, b_0=1
8,19	b_{n-2}=1,b_6=1,b_5=1,b_1=1,b_0=1
9	b_{n-2}=1, b_4=1, b_0=1
10,17,20,25,28	b_{n-2}=1, b_3=1, b_0=1
12	b_{n-2}=1,b_7=1,b_4=1,b_3=1,b_0=1
13,24	b_{n-2}=1,b_4=1,b_3=1,b_1=1,b_0=1
14	b_{n-2}=1,b_{12}=1,b_{11}=1,b_1=1,b_0=1
16	b_{n-2}=1,b_5=1,b_3=1,b_2=1,b_0=1
18	b_{n-2}=1, b_7=1, b_0=1
23	b_{n-2}=1, b_5=1, b_0=1
26,27	b_{n-2}=1,b_8=1,b_7=1,b_1=1,b_0=1
30	b_{n-2}=1,b_{16}=1,b_{15}=1,b_1=1,b_0=1

Let us consider that the task is to test a memory having 32 locations, and then the bit length '*n*' needed for addressing these locations is 5 (2^5=32). According to proposed architecture the first '*n-2*' bits of the 'n' bits in the address pattern is produced by modified '*n-2*' bit CLFSR i.e.) *n-2*=3. Hence for designing a MBIST for 32 memory locations the proposed low power address generator consists of 3-bit modified CLFSR and a 2-bit CLFSR.

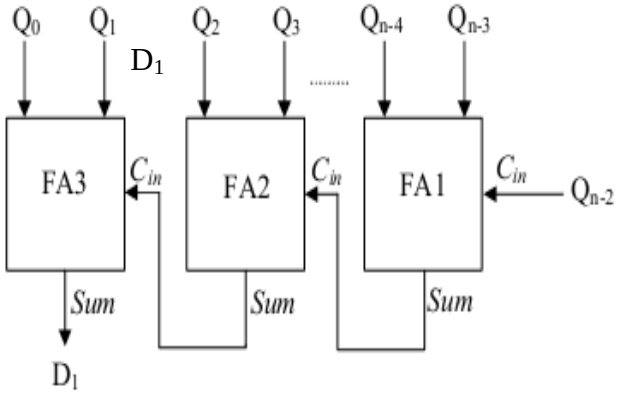

Fig. 4 Reconfigured Full adder

The reconfigured full adder provides a perfect feedback to proposed CLFSR. The input to the full adder circuit is the outputs from the D-flip flops based on the '*b*' value in the primitive polynomial chosen. The block schematic diagram for the feedback network in the proposed *n-2* CLFSR is exhibited in Fig. 4.

TABLE II TRUTH TABLE FOR THE RECONFIGURED FULL ADDER

Input-1	**Input-2**	**Input-3(C_{in})**	**Output(*sum*)**
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

The full adder adapted in this work is a reconfigured architecture of the conventional full adder architecture. In the conventional design the C_{out} of the previous full adder cell is fed to the C_{in} of the successive full adder cell but in the case of Reconfigured Full adder exhibited in Fig. 4, the C_{in} of the previous full adder cell act as a third input which is fed by the output (sum) of the previous full adder and the C_{out} of each full adder cell is left without any connection. The output for each full adder cell with all possible input combinations are tabulated in TABLE II.

B. *2-Bit CLFSR*

The operation of 2-bit CLFSR is same as that of a normal CLFSR containing two D-flip flops with external type feedback provided by using a single full adder cell with inputs Q_1, Q_2 and NOT gate output. Then the output (sum) of full adder is fed to the

FF1 as shown in Fig. 5.

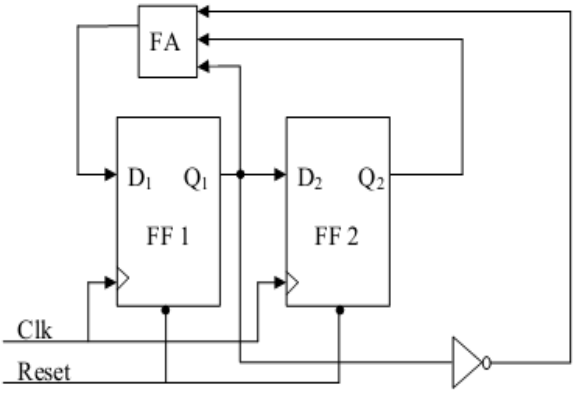

Fig. 5 2-bit CLFSR

TABLE III 2-BIT CLFSR DATA FLOW

Clk	**Q_1**	**Q_2**	**NOT**	**FA**	**D_1**	**D_2**
0	1	1	0	0	0	1
1	0	1	1	0	0	0
2	0	0	1	1	1	0
3	1	0	0	1	1	1

The data flow through the 2-Bit CLFSR for each clock cycle is tabulated in TABLE III. This data flow can be clearly analyzed through a state diagram shown in Fig. 6.

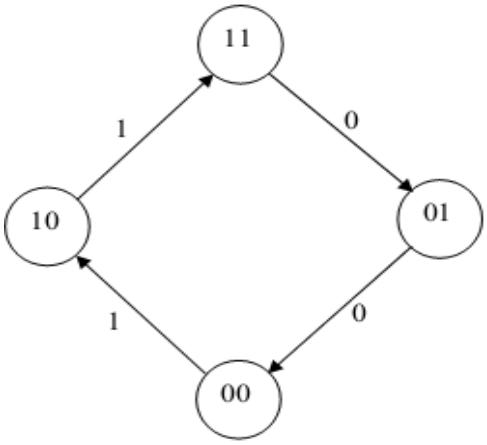

Fig. 6 State diagram for a 2-bit CLFSR

If the initial state of the 2 bits is '11' then the feedback produced by the FA is '0',which in turn is stored in the FF 1 by shifting the present state to right and hence in the next positive clock the output from the CLFSR is '01'. Hence whenever the positive clock pulse is applied a transition of state with 1 bit difference from the present value is obtained.

C. *Dual Clocking Scheme*

Conventional CLFSR with D-flip-flops when triggered with normal clocks generates bit patterns randomly with high bit transitions. The dual clocking scheme used in the proposed method contributes in low switching activity by reducing the transition between the patterns generated. The *n-2* bit modified CLFSR is triggered by a low frequency clock so that the transition from one pattern to the other pattern takes place after all the 2 bit CLFSR patterns are generated (since it is triggered by a high frequency clock). The variation of the two separate clocks signals comparing with the normal clock signal is shown in Fig. 7.

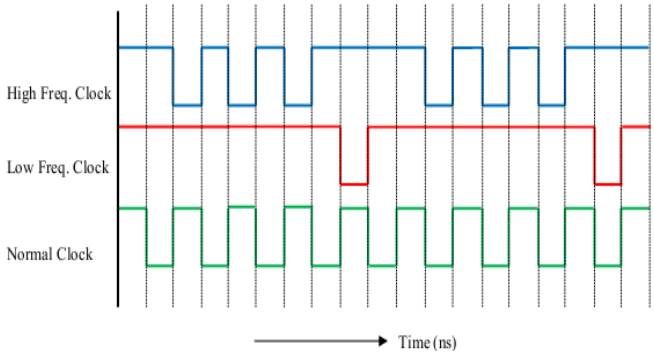

Fig. 7 Proposed two clock signals compared with normal clock

An example for the proposed address generator for 5-bit pattern generation is tabulated in TABLE IV. The proposed address generators working with each clock cycle can be understood clearly from the table. Let us consider that the initial seed be '00000'

TABLE IV PROPOSED ADDRESS GENERATOR FOR 5-BIT ADDRESS GENERATION

Time (ns)	**Low Freq. clock**	**High Freq. clock**	**Modified CLFSR**	**2 bit CLFSR**	**Output**
0	0	1			
10	1	1	000	00	00000
20	1	0			
30	1	1		10	00010
40	1	0			
50	1	1		11	00011

60	1	0			
70	0	1		01	00001
80	0	1			
90	1	1	100	01	10001
100	1	0			
110	1	1		00	10000
120	1	0			
130	1	1		10	10010
140	1	0			
150	0	1		11	10011
160	0	1			
170	1	1	010	11	01011
180	1	0			
190	1	1		01	01001
200	1	0			
210	1	1		00	01000
220	1	0			
230	0	1		10	01010
240	0	1			
250	1	1	110	10	11010
260	1	0			
270	1	1		11	11011
280	1	0			
290	1	1		01	11001
300	1	0			
310	0	1		00	11000
320	0	1			
330	1	1	001	00	00100
340	1	0			
350	1	1		10	00110
360	1	0			
370	1	1		11	00111
380	1	0			
390	0	1		01	00101
400	0	1			
410	1	1	101	01	10101
420	1	0			
430	1	1		00	10100
440	1	0			
450	1	1		10	10110
460	1	0			
470	0	1		11	10111
480	0	1			

490	1	1	011	11	01111
500	1	0			
510	1	1		01	01101
520	1	0			
530	1	1		00	01100
540	1	0			
550	0	1		10	01110
560	0	1			
570	1	1	111	10	11110
580	1	0			
590	1	1		11	11111
600	1	0			
610	1	1		01	11101
620	1	0			
630	0	1		00	11100
No. of Transitions					(7)(3)(1)(12)(12)=35

From the above table it is clear that for 5-bit address generation, the proposed address generator exhibits only '35' switching activities which is much low when compared to the patterns generated by other methods.

V. RESULTS AND COMPARISON

In this section, the focus is on FPGA implementations of the proposed architecture in the previous sections.

The proposed architecture is coded in verilog HDL and are synthesized using Xilinx ISE version 14.1 and are implemented on Xilinx Virtex-5 xc5vlx20t-2ff323 FPGA. The RTL schematic for proposed address generator implemented for 5 bit and 10 bit address generation is shown in Fig.8 and Fig. 9 respectively

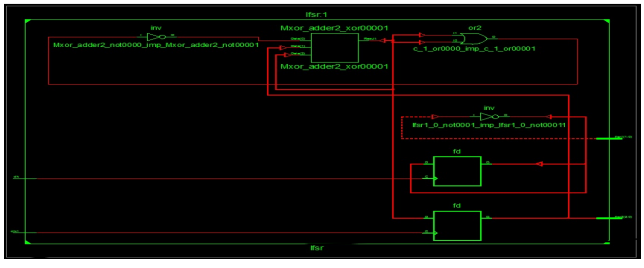

Fig 8 RTL schematic for proposed 5-bit address generator

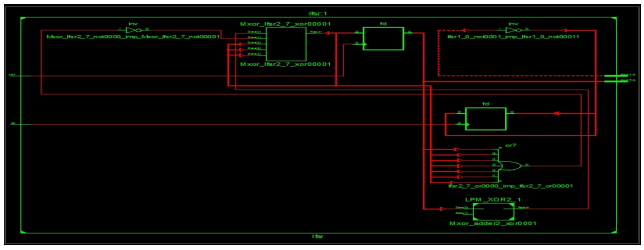

Fig 9 RTL schematic for proposed 10-bit address generator

The Simulation result for proposed address generator implemented for 5 bit and 10 bit address generation is shown in Fig.10 and Fig. 11 respectively

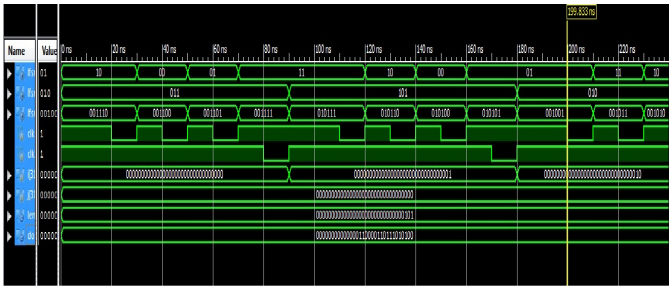

Fig 10 Simulation for proposed 5-bit address generator

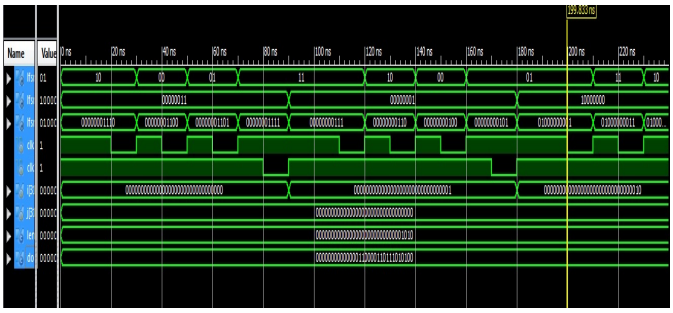

Fig 11 Simulation for proposed 10-bit address generator

The output switching activity of various address generators that are computed in [16] is compared with the switching activity computed from the proposed Address generator and tabulated in TABLE V. Comparing all other address generator's it is identified that the switching activity exhibited by proposed address generator performs low switching activity and thus making the system more efficient in dynamic power consumption. It is also observed that if the seed value is changed, the switching activity changes, hence by selecting a proper method for seed selection the proposed address generator's dynamic power dissipation can be much reduced.

TABLE V SWITCHING ACTIVITY COMPARISON

Seed	n	LFSR [16]	BSLFSR [16]	DS-LFSR [16]	Bipartite LFSR [16]	BS & DS LFSR [16]	Proposed
111….1	5	85	69	70	45	64	36
	10	5130	4106	2904	2462	2376	1783
010….1	5	85	69	55	48	57	32
	10	5130	4106	2643	2428	2315	1777

TABLE VI POWER COMPARISON

Address Generator [5-bit]	Total Dynamic Power(mW)	Total Power (mW)
LFSR [17]	222	402
BS-LFSR [17]	199	378
PROPOSED	12	333

The power comparison of the proposed method with conventional LFSR and BS-LFSR are tabulated in TABLE VI. Conventional LFSR when used as address generator consumes a 222mW dynamic power and BS-LFSR when used consumes 199mW saving a dynamic power consumption of 10.36% comparing with conventional LFSR but proposed low power address generator saves up to 94.59% of total dynamic power. From the above comparison it is obvious that the proposed address generator shows low dynamic power dissipation because of reduced switching activity.

Area:

TABLE VII AREA OCCUPIED BY THE PROPOSED ADDRESS GENERATOR

n	Slices (3120)	LUT (12480)	Registers (12480)	IOBs (172)
5	3	1	5	7
10	4	2	10	12

From the Table VII, For 5-bit and 10-bit address generator the proposed architecture occupies 3 and 4 slices out of 3120 slices, 1 and 2 LUT out of 12480 LUT's, 5 and 10 Registers among 12480 available registers. All these resources utilize only about 1% of the total resources. Among the total 172 IOB's, 5-bit and 10-bit address generator occupies only 7 and 12 IOB's thereby contributing 4% and 6% to the total resource utilization.

VI. CONCLUSION

Power consumption is a serious concern during the testing of memory and the main source of this power consumption is the bit transition between the generated bit patterns leading to high switching activity. In this work an efficient address generator that works for MBIST with Zero-One testing algorithm with reduced switching activity is implemented. The proposed address generator was synthesized and simulated in Xilinx ISE version 14.1 in verilog-HDL with Virtex-5 xc5vlx20t-2ff323 FPGA. The switching activity of the proposed address generator is compared with other existing works and it is confirmed that the proposed method shows reduced switching activity over all other methods. The power report was obtained using XPower analyzer tool and compared with the existing address generators, shows that an average dynamic power reduction of about 94.26. %. The area report is also generated and is tabulated for 5-bit and 10-bit address generators

REFERENCES

[1] Mohammed Niamat, Manoj Lalla, and Junghwan Kim, "Testing Faults in SRAM Memory of Virtex-4 FPGA," 52nd IEEE International Midwest Symposium on Circuits and Systems, pp:965-970, 2009.

[2] Nur Q.Noor, Yusrina Yusof, and Azilah Sparon, "Low Area FSM-Based Memory BIST for Synchronous SRAM", proceeding of the international colloquium of Signal Processing and Its application, pp. 409-412, 2009.

[3] Miron Abramovici and Charles Stroud, "BIST-Based Test and Diagnosis of FPGA Logic Blocks," IEEE Trans. on VLSI Systems,Vol. 9, No. 1, pp:159-172, Jan. 2001.

[4] Tzu F.Chien, Wen C.Chao, Chein M.Li, Yao W.Chang, Kuan Y.Laio, Ming T.Chang Min H.Tsai an Chih M.Tseng, "BIST Design Optimization for Large-Scal Embedded Memory Cores", ICCAD, SanJose, California, USA, 2009.

[5] Luigi Dilillo, Paul Rosinger, and Bashir M.Al-Hashimi, "Minimizing Test Power in SRAM through Reduction of Pre-charge Activity", proceeding the Design, Automation, and Test in Europe (DATE) conference, pp.1-6, 2006.

[6] Ad J. van de Goor, Halil Kukner and Said Hamdioui, "Optimizing Memory BIST Address Generator Implementations," 6th International Conference on Design & Technology of Integrated Systems in Nanoscale Era, pp:1-6, april 2011.

[7] Nor Z.Haron, Siti A.Junos, Abdul Hadi A.Razak, and Mohd Y.Idris, "Modeling and Simulation of Finite State Machine Memory Built-in Self Test Architecture for Embedded Memories", Asia-Pacific Conference on Applied Electromagnetics", 2007.

[8] T. Rudnicki and A. Háawiczka, "Test Pattern Generator for Delay Faults," IEEE conference on Design and Diagnostics of Electronic Circuits and Systems, pp:1-4, 2007.

[9] R.Muthammal, and Dr K O Joseph, "Low Power Efficient Built in Self Test," IEEE International Conference on Microwaves, Communications, Antennas and Electronics Systems, pp:1-5, 2011.

[10] Seongmoon Wang, and Sandeep K.Gupta, "DS-LFSR:A BIST TPG for Low Switching Activity, IEEE Transactions On Computer-Aided Design of Integrated Circuits And Systems, 7, July, 2002.

[11] Nilanjan Mukherjee, Artur Pogiel, Janusz Rajski, and Jerzy Tyszer, "BIST-Based Fault Diagnosis for Read-Only Memories," IEEE Transactions On Computer-Aided Design of Integrated Circuits And Systems, Vol. 30, no. 7, July 2011.

[12] Chun-Lung Hsu and Ting-Hsuan Chen, "Built-in Self-Test Design for Fault Detection and Fault Diagnosis in SRAM-Based FPGA," IEEE Transctions On Intrumentation And Measurement, Vol. 58, no. 7, July 2009.

[13] Mohamed H. Abu-Rahma, Mohab Anis, and Sei Seung Yoon, "Reducing SRAM Power Using Fine-Grained Wordline Pulsewidth Control," IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 18, no. 3, March 2010.

[14] Abdallatif S. Abu-Issa and Steven F. Quigley, "Bit-Swapping LFSR and Scan-Chain Ordering: A Novel Technique for Peak- and Average-Power Reduction in Scan-Based BIST," IEEE Transactions On Computer-Aided Design of Integrated Circuits And Systems, Vol. 28, no. 5, May 2009.

[15] T. Nandha Kumar and Fabrizio Lombardi, "A Novel Heuristic Method for Application-Dependent Testing of a SRAM-Based FPGA Interconnect," IEEE Transactions On Computer, Vol. 62, no. 1, Jan 2013.

[16] Ahmed N.Awad & Abdallatif S.Abu-Issa, "Low Power Address Generator for Memory Built-In Self-Test," The Research Bulletin of Jordan ACM, Vol. II (III), pp:52-56, 2011.

[17] C.ravishankarreddy, shaikzilani, V.sumalatha, "Low Power, Low-Transition Random Pattern Generator", International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 5, July 2012.

[18] Said Hamdoui, and Zaid Al-Ars, "Scan More with Memory Scan Test", proceedings of 4th International DTIS conference, 2009, pp.204-209.