International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 6 (2015) pp. 14553-14562
© Research India Publications
http://www.ripublication.com

Memory Testing on FPGA Board using BIST

A V Pradeep Chandra, M Santosh Kumar, G Surya Teja, D SaiVamsi Krishna

School of Electronics Engineering, VIT University, Vellore-632014.
Tamil Nadu, India, email:avpchandra@gmail.com

Abstract

In this paper we acquired the concept of BIST using which, we planned to perform the memory testing on FPGA board. The testing components involve Complete LFSR, Data LFSR and Memory address register (MAR). Testing the memory with this method avoids the need of external ATPG. So it reduces cost as well as the complexity of the testing circuitry. For this method we are expecting the low power along with the less occupation on the chip.

INDEX Terms- CLFSR, MAR, SRAM cell, Testing

I. Introduction

Built in self test (BIST) is a testing technique used for on chip testing. The general testing scheme for testing the memory cores is shown in Fig.1. BIST provides more flexibility as the maintenance and repair of circuit becomes easier. It is not possible to break the large VLSI circuits for testing, in case of cascading two or more devices it is difficult to find out the test for a complete system from the tests used in the individual parts of the circuits[1]. Here BIST provides a way to decompose the large circuit to be tested. In this project memory testing is done using BIST algorithm. A basic BIST should contain a test pattern generator, a memory under test and a test pattern compactor and a comparator. Implemented circuit consists of a hardware pattern generator (LFSR), MUX, circuit under test (memory), output response compactor followed by comparator to compare the output of memory with the correct output. Here pseudorandom pattern BIST is used which means the test patterns are generated using linear feedback shift registers (LFSR). Its benefit is that it requires very little hardware overhead which is the major concern in the implementation of BIST. In case of stored pattern BIST the hardware requirement is increased because of memory devices used to store the test patterns [2]. But in pseudorandom LFSR the fault coverage has to be compromised because it will not generate completely random patterns pseudo random pattern means the sequence will be predictable and will be repeated hence it depends on the circuit that how much fault coverage can be achieved.

II. COMPLETE LINEAR FEEDBACK SHIFT REGISTER

Linear feedback shift register(LFSR) consists of D-Flip flops and exclusive OR gates(XOR). In LFSR the exclusive OR network forms the feedback network. There are two types of LFSR namely external and internal LFSR. In internal LFSR the feedback network has the two extreme D-flip-flops connected with each other and accordingly each exclusive OR gate is connected to a common feedback line through one input and to the output of D-flip-flop through other input as shown in figure.For external LFSR the extreme flip-flop is not connected to each other directly instead they are connected through exclusive or gates.

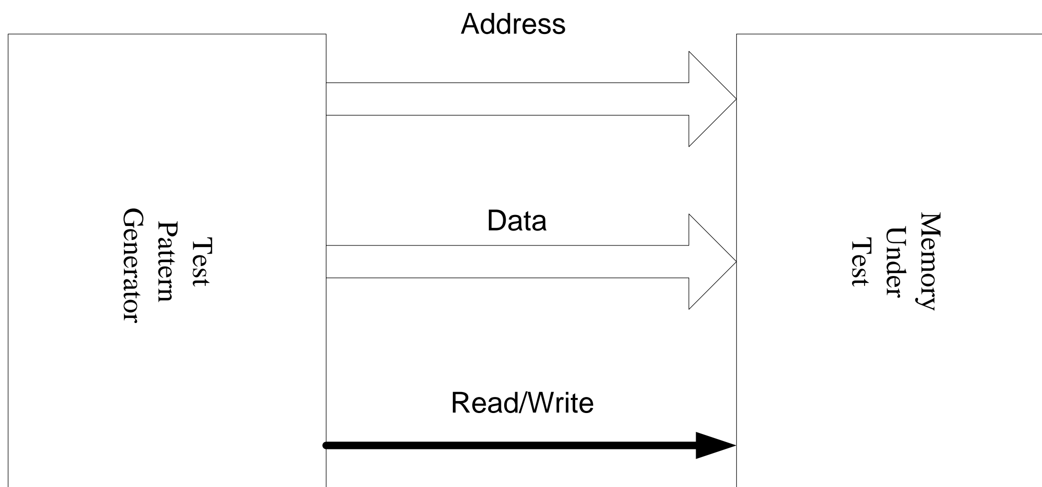


Fig 1. The general scheme for Memory Testing

Here the output of consecutive exclusive or gates are connected to the successive XOR gate and the remaining input is taken from the output of the flip flop.Here external LFSR is used but the circuit is possible also using the internal LFSR also. The LFSR is used in two forms as shown below. There is a need to generate a sequence for data input and the other sequence to generate for the memory address generation. For memory address generation complete LFSR(CLFSR) [3] is used and to generate data sequence data LFSR is used. In CLFSR along with the D-flip flop and exclusive OR gates, OR gates, NOR gates multiplexer are added to generate up-down sequence according to the requirement. According to the MUX select input up or down sequence is selected.

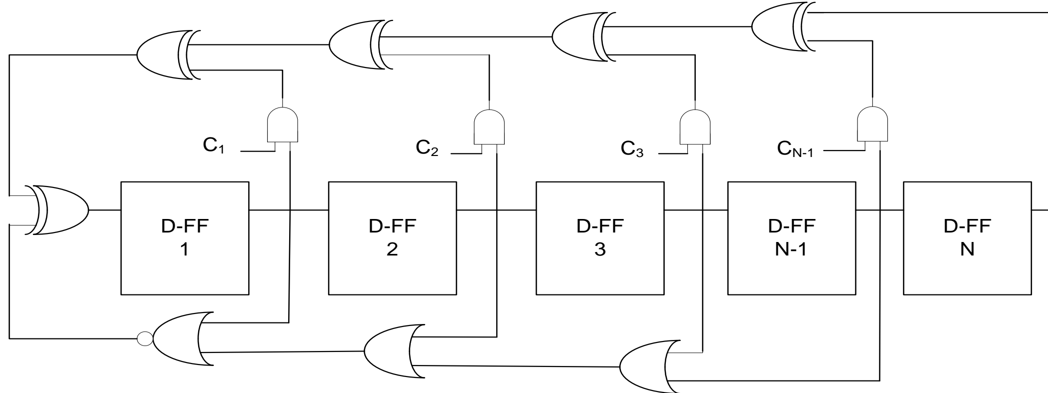


Fig 2 .The N stage Up CLFSR with U(z)

An n-bit LFSR generates 0 to 2N-1 pseudo-randomly sequences [4]. Now a days LFSR is widely used in testing the memory chips. Here we are using complete LFSR(CLFSR) as a memory address generator. It will generate up/down sequences.Fig.2. shows an n stage CLFSR with a primitive polynomial [5].

$$U(z) = \sum_{i-0}^{N} C_i * z^i \ ; C_0 = C_N = 1 \qquad (1)$$

The sequence generated are

$$S_i = \{Q_1, Q_2, Q_3 \ldots Q_{N-1}, Q_N\}_I \quad \text{for } 0 \leq i \leq 2^N\text{-}1$$

The characteristic matrix is expressed as

$$U = \begin{bmatrix} C1 & 1 & 0 & ---- & 0 \\ C2 & 0 & C1 & ---- & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ Cn\text{-}1 & 0 & 0 & ---- & 1 \\ Cn & 0 & 0 & ---- & 0 \end{bmatrix}$$

The relation between S_i andS_j is

$$S_i = S_j * U^{i\text{-}j} + P_j \ , \ i \geq j \qquad (2)$$

Here i and j are internal stages of the generated sequences
Where $P_i = (p;0;0;\ldots0)$ andp=1, if $S_i=(0;0;0\ldots.0)$; p=0, elsewhere

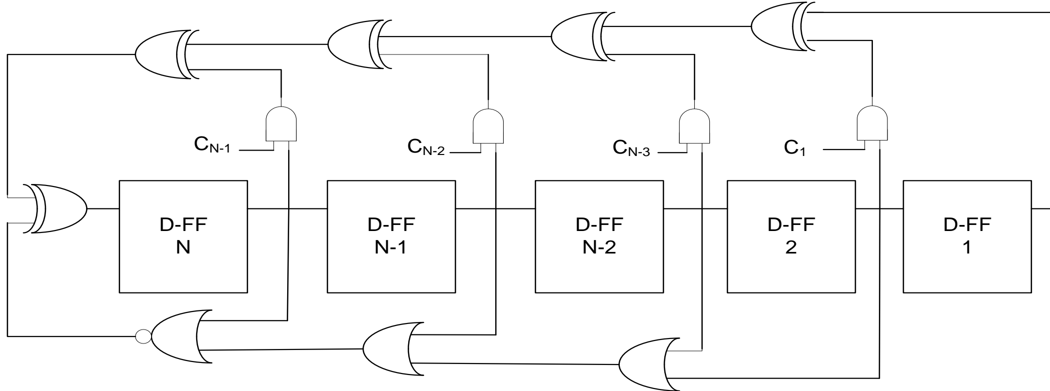


Fig 3. The N stage Down CLFSR with V(z)

The N stage CLFSR with U(z) generates the up sequences [5]. Then the corresponding down sequences also generated by the CLFSR with primitive polynomial V(z). The V(z) is reciprocal to the U(z). The N stage down CLFSR with V(z) is shown in Fig.3

$$V(z) = \sum_{i-0}^{N} C_i * z^{n-i} \; ; C_0 = C_N = 1 \qquad (3)$$

The sequence generated are

$$Y_i = \{Q_1, Q_2, Q_3 \ldots Q_{N-1}, Q_N\}_I \quad \text{for } 0 \leq i \leq 2^N - 1$$

The characteristic matrix is expressed as

$$V = \begin{bmatrix} 0 & 0 & ----- & 0 & 1 \\ 1 & 0 & ----- & 0 & C1 \\ 0 & 1 & ----- & 0 & C2 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & ----- & 1 & Cn-1 \end{bmatrix}$$

The relation between Y_i andY_j is

$$Y_i = Y_j * V^{i-j} + R_j \, , \; i \geq j \qquad (2)$$

Here i and j are internal stages of the generated sequences

Where $R_i = (0,0,0\ldots r)$ andr=1, if Y_i=(0;0;0….0) ; r=0, elsewhere

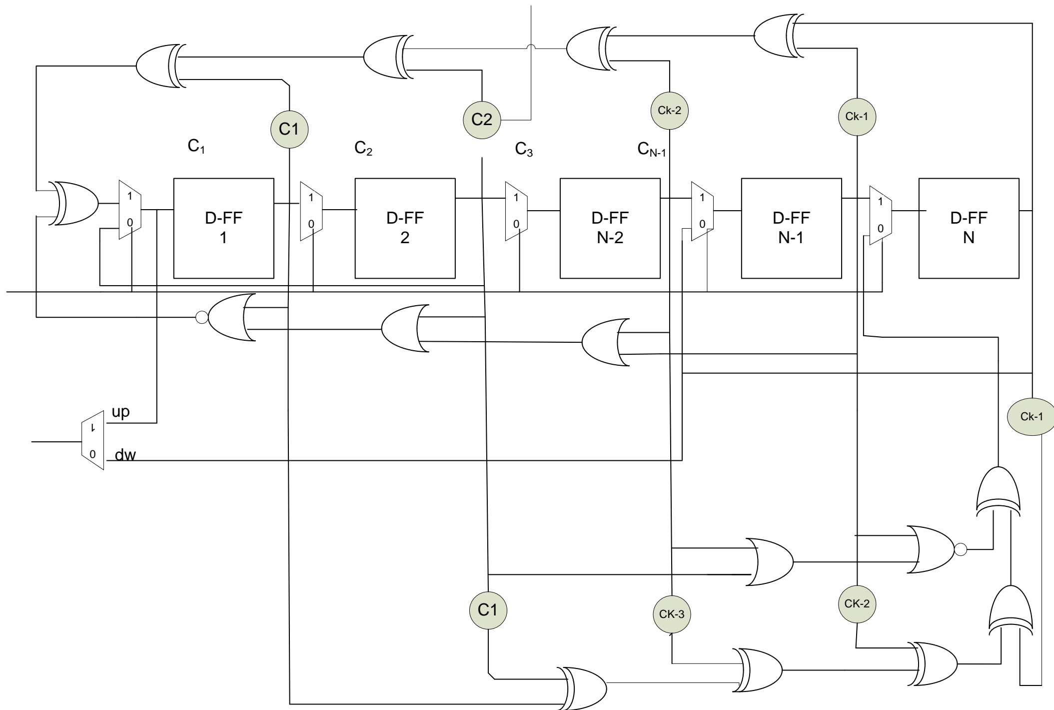


Fig 4. The N stage Up/Down CLFSR with U(z) and V(z)

The CLFSR generates all 2N sequences including all zero sequence. The relationship between U(z), V(z),S_i,S_j,$Y_i$and Y_j is

$$S_i = S_j{*}U^{i-j}$$

$$Y_i = Y_j{*}V^{i-j}$$

The multiplication of U and V matrices resuIt to an idempotent matrix I i,e., U*V=I. Therefore U is the inverse of V i.e., U=V=1. To reduce the area constraints and to save the cost of the memory address generator here we are using an integrated up/down CLFSR with both U(z) and V(z). The Fig.4 shows N-stage Complete up/down CLFSR.

Here multiplexer is used to select the up or down sequences. The up/dw signal controls the multiplexer. If up/dw controls 1 the CLFSR will generate up sequence and the internal state transitions are going up. If up/dw signal is 0 the CLFSR will generate down sequences and internal state transitions are going down .The output of the CLFSR taken serially through the ‘seq’ signal. The seq output is D1 if the up/dw control is 1 and it is Q_N if the up_dwcontrol is 0.

In this project we designed an 8-bit complete up/down LFSR with polynomial U(z)=1+z2+z3+z4+z8. Fig.5 shows the CLFSR with this polynomial.

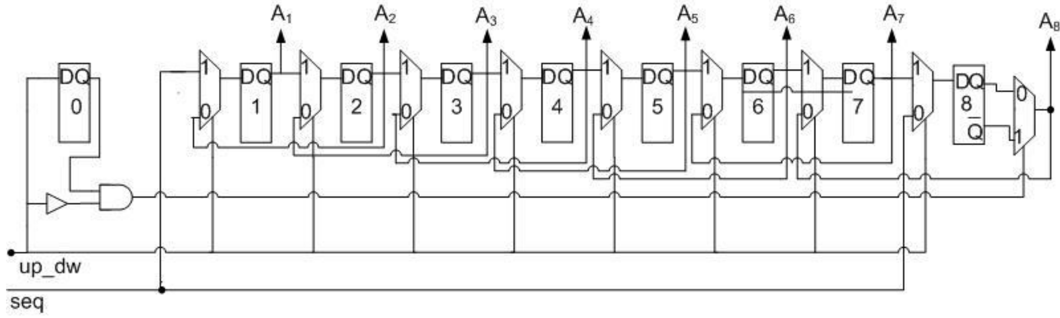


Fig 5. The 8 stage up/down CLFSR with U(z) = 1+Z^2+Z^3+Z^4+Z^8

A. Data LFSR

In this project we are using an 8 bit Data Linear feedback shift register with polynomial D(z)=1+z+z2+z8. The DLFSR is used generated the data randomly at every clock cycle. This generated data is directly given to the Data-w pin of the SRAM memory as explained in section 4. The circuitry of DLFSR is same as up CLFSR with respect to the corresponding polynomial.

III. The Memory Address Register (MAR)

Memory address register is a register that stores memory address from which data is sent to CPU or address to which data will be sent and stored. MAR holds the memory address location from where next instruction is to be executed. It generates proper up/down memory addresses of scan based algorithm to test memory blocks.

Up to now we designed an up/down CLFSR which acts as memory address generator (MAG), which generates up/down sequences and apply them serially to MAR. Now we have to pass address to the SRAM chip, for that we are using MAR taking sequence generated by CLFSR as input to generate N-bit up/down addresses (A1 A2 A3…..An-1 An). In Fig.6 the process of applying up/down sequence to the MAR is shown.

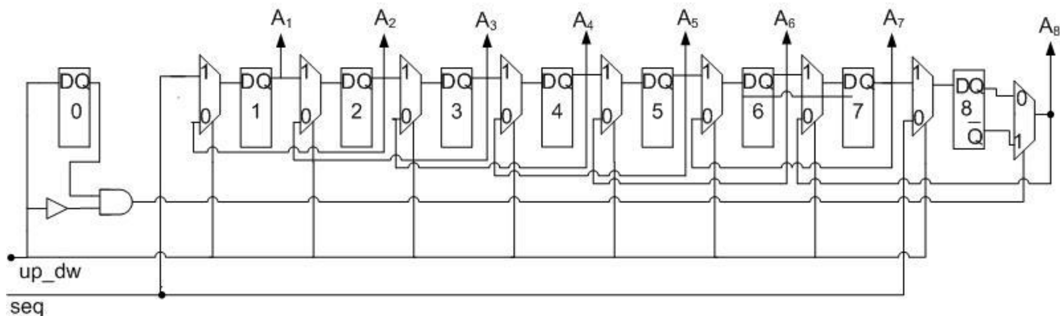


Fig 6. The 8 stage MAR

IV. SRAM

Here we are using SRAM cell as a memory for testing. SRAM stands for static random access memory. It is one form of the semiconductor memory normally used in many microprocessors and computing applications [6]. It is volatile which means when power is switched off the data is lost from the memory. SRAM is used to store the data. It stores the data in static manner in contrast to the DRAM where the data is held in dynamic fashion. In SRAM the data can be written to or read from memory randomly. The main advantages of SRAM compared to DRAM are SRAM is faster than DRAM and cost is very low compared to DRAM [7].

The basic SRAM cell consists of four transistors as a cross coupled inverters. In addition to this we are using another two transistors for controlling inputs to operate the memory. So totally it is a six transistors memory cell. We may also use more transistors like eight and ten for the other applications like implementing additional ports.

SRAM array is represented as 2^mx n. Here 'm' denotes number of address bits and 'n' denotes number of data bits. This means that a SRAM cell has 2^m address locations and in each location we can store n bits of data. The basic block diagram of SRAM is given in Fig.7.

Address line is used to select the memory location of the chip. The number of address lines represents the size of the memory. For example a 256K X 8 memory has 18 address lines (2^{18}=256K) and 8 data input and output lines. Data-w pins are used to write the data into memory and Datar pins are used to read the data from the memory. During write operation the data is applied to Dataw pins. This data is translated and store at a particular memory location. During read operation the data from the selected memory location is appeared at Datar pins. To select chip we use an active low control signal CS.

Read operation: CS=0, WE=1, RE=0

Write operation: CS=0, WE=0, RE=1

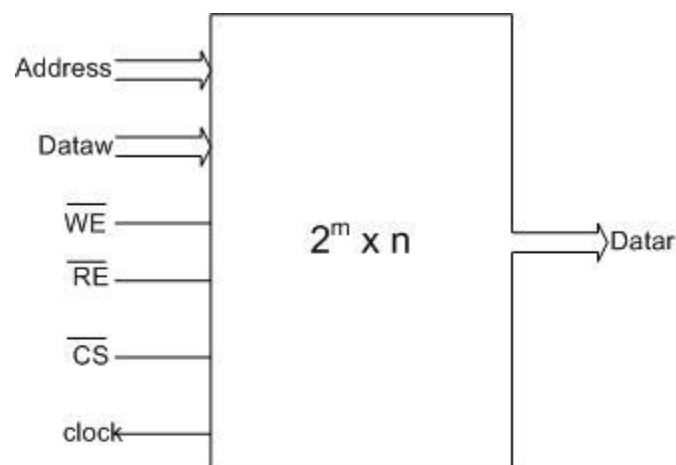


Fig 7. The SRAM cell

In this paper we use the output of MAR is connected to address pins of the SRAM. Initially data is written into the cell using the addresses generated by MAR. The data to be written into address pointed by the MAR is generated here using an 8-bit data LFSR.

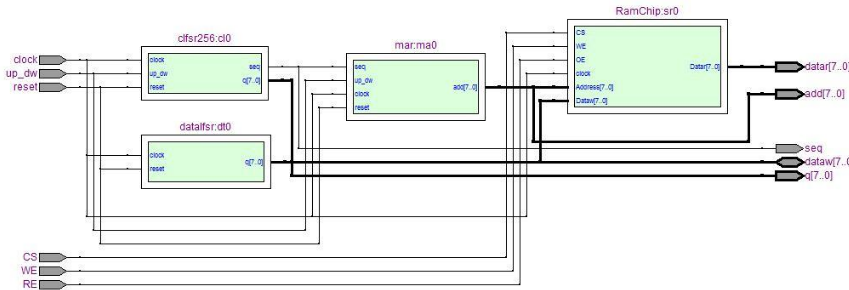


Fig 8. The SRAM testing architecture

The overall architecture of SRAM cell testing by using CLFSR, DLFSR and MAR is shown in the Fig.8. This architecture is the Netlist viewer of Quartus-II tool.

V. RESULTS AND DISCUSSIONS

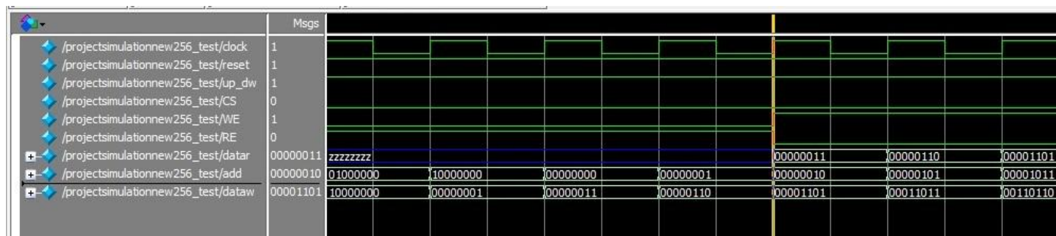


Fig 9.Testing of SRAM cell with Up-CLFSR

Fig 9 and Fig 10 shows the testing of SRAM cell with UP and DOWN CLFSR respectively. The waveforms are showing the read and write operations of SRAM cell.

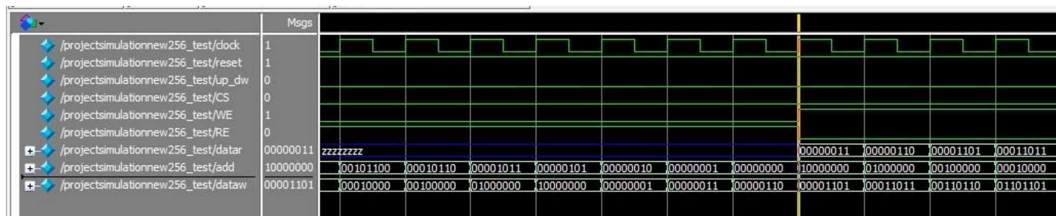


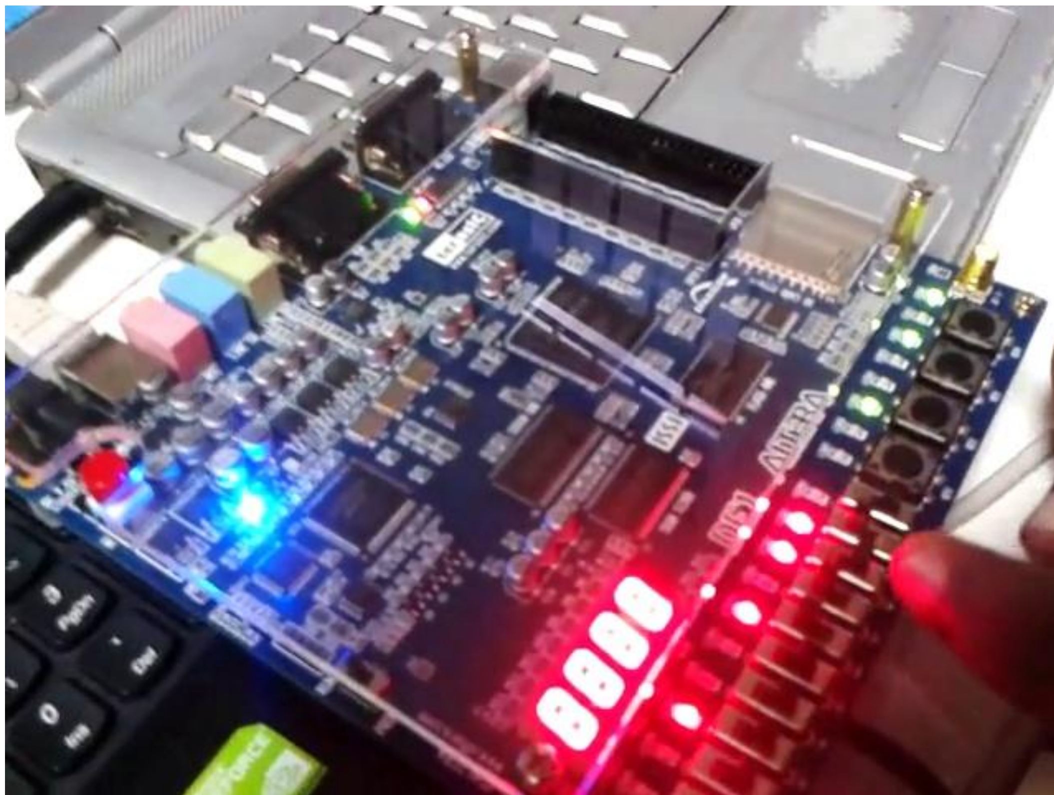
Fig 10. Testing of SRAM cell with Down-CLFSR

Upto 256 clock cycles it is writing the data into memory from Data LFSR and after 256 clock cycles it can read data from the memory. The table 1 shows the details of power, area required to this implementation [8].

Table I

	Area(um^2)	Power(uW)	Logic elements
SRAM testing using BIST	52503	5712.15	2757

The complete design is verified with the ALTERA DE-1 board. The following Fig 11 shows the implementation of the design on the kit.



VI. CONCLUSION

In this paper we designed internal testing circuitry for SRAM chip on the basis of system on chip design methodology. This confronts the use of external ATPG for testing the memory chips. So, it adds cost effective feature in the testing field. For generating up/down memory addresses with less routing area, CLFSR as a memory address generator by using scan based algorithms was proposed in this paper. We verified the CLFSR with an 8-bit primitive polynomial for both up/down sequence to test the SRAM cell. The outputs of CLFSR and MAR are verified in this paper.

ACKNOWLEGMENT
The authors are greatly supported by Prof. Sivasankaran K for ideas to present this paper.

REFERENCES

[1] E.Marinissen, B.Prince, D. Keltel-Schulz, and Y. Zorian, "Challenges in embedded memory design and test," in Design, Automation and Test in Europe, 2005. Proceedings , 2005 , pp. 722-727 vol.2.

[2] B. Nadeau-Dostie, A. Silburt, and V. Agarwal, "Serial interfacing for embedded-memory testing," Design Test of Computers, IEEE, vol.7, no.2, pp.52-63, 1990.

[3] E.McCluskey and S. Bozorgui-Nesbat, "Design for autonomous test," Circuits and Systems, IEEE Transactions on, vol.28, no. 11 , pp. 1070-1079, 1981.

[4] M. Nicolaidis, V. Alves, and O. Kebichi, "Trade-offs in scan path and bist implementations for rams," in European Test Conference, 1993.

[5] W.-L. Wang and K.-J. Lee, "A Complete memory address generator for scan based march algorithms," in Memory Technology, Design, and Testing, 2005. MTDT 2005. 2005 IEEE International Workshop on, Aug 2005, pp. 83-88.

[6] P.K. Veenstra, F.P.M. Beenker, and J.J.M.Koomen,"Testing of random access memories: theory and practice," Electronic Circuits and Systems,IEEE Proceedings G, vol. 135, no. 1, pp.24-28, 1988.

[7] J.Zhao, S.Irrinki, M.Puri, and F.Lombardi, "Testing sram-based content addressable memories,"Computers, IEEE Transactions on, vol.49,no. 10, pp.1054-1063, 2000.

[8] V. Kalyanam and N. Touba, "Htpg: hybrid test pattern generation for reducing test storage," in Autotestcon, 2005.IEEE, Sept2005,pp.759-765.