ISSN: 2250–3676
[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-5, 1534 – 1538

EFFICIENT FAULT COVERAGE VLSI ARCHITECTURE FOR LFSR BASED IMAGE WATERMARKING SRAM

S.Jagadeesh[1], S. Bhargav Kumar[2], Dr.M.Ashok[3]

[1] *Associate Professor & HOD, Department of Electronics and Communication Engineering, Sri Sai Jyothi Engineering College, JNTUH, Gandipet, Hyderabad-75, (A. P.), India,* ***jaaga.ssjec@gmail.com.***

[2] *P.G. Student, M.Tech. (VLSI), Department of Electronics and Communication Engineering, Sri Sai Jyothi Engineering College, JNTUH, Gandipet, Hyderabad-75, (A. P.), India,* ***kumar.s.bhargav@gmail.com.***

[3] *Professor, Department of Computer Science and Engineering, Sri Sai Jyothi Engineering College, JNTUH, Gandipet, Hyderabad-75, (A. P.), India,* ***maram_ashokssjec@yahoo.com.***

Abstract

Recent advances in the development of image watermarking algorithms had made a rapid change in the authenticated information resource sharing. Among all techniques of image watermarking and storing watermarked image bits in SRAM (Static Random Access Memory), LFSR (Linear Feedback Shift Register) based image watermarking technique has been proposed in [1], this technique utilizes less design complexity for VLSI (Very Large Scale Integration) on-chip hardware architecture compared with the other techniques. In our paper, the FPGA (Field Programmable Gate Array) prototyping of image watermarking SRAM [1] has been presented. In our paper, we are presenting BIST-R (Build in Self Test and Repair) based efficient fault coverage VLSI architecture, where the proposed method of SRAM hardware architecture can be utilized in real time image watermarking encoder and decoder applications. In our paper, LFSR based image watermarking SRAM build in self test and repair method has been presented. The comparative results have shown efficient fault coverage in detecting the faults in less time and utilized less power during the SRAM hardware implementation. We simulated and synthesized all our proposed modules using Xilinx 9.1i and ModelSim XE III 6.4b. The layout and sketch design of our modules for MBIST (Memory Build in Self Test) and MBISR (Memory Build in Self Repair) were implemented using layout and SPICE tools. The complete layout of our proposed method has been implemented on CMOS 0.12µm technology VSLI architecture.

Index Terms: *SRAM, LFSR, on-chip hardware architecture, FPGA prototyping, BISR, fault coverage, VLSI architecture, MBISR*

--- *** ---

1. INTRODUCTION

Image watermarking technique has been evolved to protect the owner's copyright protection. During transmission of image, the encoder and decoder plays a major role in the selection of image data transmission through the communication channel. Due to the free-resource sharing through internet, the data transmission has become an open source. Decoders and encoders [5 and 7] are to be tested during the transmission of the data over the communication channel, to detect any illegal operations has been made on the data transmission. By detecting these operations, decoders and encoders become reliable and our desired data will not be encrypted by the third-party. Build in Self Test is the reliable technique to test these modules.

Recent advancements in the IC's package made testing increasingly difficult and the built-in self test (BIST) has emerged as a promising solution to the VLSI (Very Large Scale Integration) testing problem. BIST is one of the designs for testability methodology to detect faulty components in a system by incorporating in-built test logic. The main modules of BIST are the automatic test pattern generator (ATPG), the response comparator (RC), and the signature analyzer (SA). The test generator applies a sequence of random patterns to the circuit under test (CUT), the responses are compared into a signature by the response comparator, and the signature is compared to a fault-free reference value.

BIST is having numerous advantages such as no design requirement for automatic test equipment and its support, with reasonable fault coverage. Linear Feedback Shift Registers

(LFSRs) which are having simple structure [2] commonly used as pseudo-random test pattern generators in BIST schemes. The design of the LFSR pseudo-random test set varies depending on the CUT. In the design of CUT the test insertion technique and weighting is a complex task which causes performance degradation, additional area and delay overheads. By making the LFSR to select good seed value, cost and test length will decreased. LFSR seed values will be stored with watermarked bits in a SRAM (Static Random Access Memory); the same LFSR used to generate pseudo-random patterns is loaded with seeds from the previous LFSR seed value.

In this paper, we propose a fault coverage-based method to compute an efficient seed of a given LFSR-ATPG. Our method is intended to produce a one-seed test sequence of a given test length that achieves a high global fault coverage. Moreover, it concentrates on the hardest [3] to detect faults of the CUT. In our proposed method of fault testing, the CUT is memory under test (MUT). The main feature of the proposed method is that it allows the hardware overhead area is reduced. Our simulation and synthesis report shows efficient fault coverage architecture for any image watermarking SRAM.

The rest of the paper is organized as follows. In the next section, we give details about our proposed method of fault coverage. In section 3 and 4, we present the simulation and synthesis results of our method of memory testing. Concluding remarks are given in section 5

2. PROPOSED EFFICIENT FAULT COVERAGE MODEL

In our proposed method, the watermarking technique is implemented on cover image [4] (JPEG 2000 standard) as shown in the Fig: 2.1. The watermark bits are generated by LFSR of 8 bits. Using AOI logic [1] equation 2.1, the bits of cover image will be watermarked with the watermark bits generated by LFSR.

Fig: 2.1: Cover image.

In AOI logic circuit, the implementation [1] will be done based on the

$$W_d = w_i * En\_1*Clk\_1+c_i* En\_1 * Clk\_1 +w_i c_i *Clk\_1+w_i c_i *En\_1*Clk\_1......................Eqn.(2.1)$$

logic operation on cover image (c_i) bit patterns and watermark image (w_i) bit patterns. The bit patterns are taken individually form original and watermarked image. In the equation 1, the subscripts are : W_d is the watermarked bit pattern, w_i is watermarking image bit, c_i is cover or original image bit, En_1 is enable input and Clk_1 is clock input.

The watermarked bits will be stored in the SRAM memory address locations. These address locations will be generated by another LFSR of 8 bits. The watermarked bits which are stored in SRAM will be considered as an encoder, a prototype is made. And this SRAM as an encoder will be tested using MBIST technique. The implementation of our proposed method of storing the watermark bits in SRAM is shown in the Fig: 2.2.

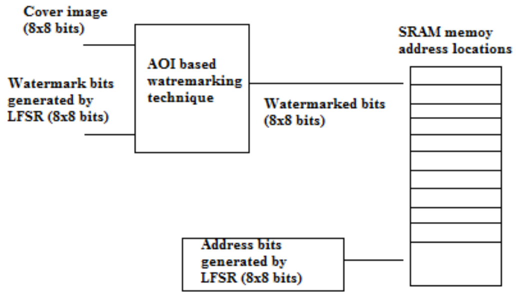

Fig: 2.2: Proposed method of watermarking

In Fig: 2.2 the watermarked AOI bits will be stored in the SRAM memory location of 256x8 size. The address locations will be stored in the temporary 8 bit register of SRAM. These address locations will be occupied by the image bits of 32x32 to accommodate 256x256 size cover image of 8 bit pattern. Now SRAM will be tested using MBIST (Memory Build-in Self Test), to check whether the bits are arranged in their memory locations or not, which is shown in Fig 2.3.

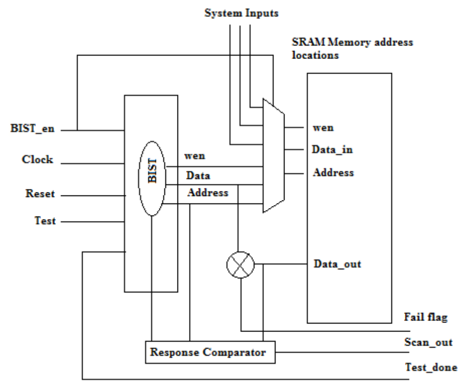

Fig: 2.3: Block diagram of MBIST

In Fig: 2.3 SRAM will be tested under system inputs and RC. ATPG is used in MBIST, LFSR will generate the random test pattern generation for ATPG. In MBIST, the device under test is SRAM array. If any address location is detected with fault data bits, MBIST using ATPG will detect the type of fault and given to the MBISRA controller. The detected faults and its type will be repaired using MBISRA (Memory Build-in Self Repair Analyzers) which is shown in Fig 2.4.

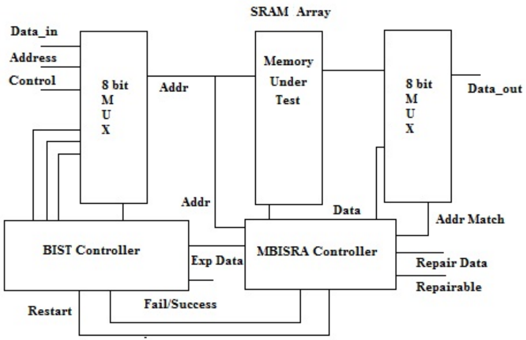

Fig: 2.4: Block diagram of MBIST and MBISRA Controller

In Fig 2.4 MBISRA controller is presented, the type of fault will be detected by the expected data with its memory location. The type of fault we detected in our method is stuck at and bit oriented faults. A check bit named repairable will decide whether the fault can be detected or not. The detected fault memory location will be replaced with the correct data by repair data bit. And the address before fault and after fault will be checked by address match bit. After rectifying the type of fault, the desired data will be placed in its memory location of SRAM. The data will be displayed at data out bits, which will be used for reverse watermarking of the cover image. In reverse watermarking [6], LFSR is used to decrypt the original cover image. Decrypting bits will be stored in SRAM, which will undergo same process of MBIST and MBISRA at the receiver section of the communication system. At receiver, our proposed method had shown good results which made our technique of memory testing an efficient.

Above method of memory testing clearly explains the method of watermarking, SRAM testing and repair for both encryption and decryption of the data bits. Our results showed efficient fault coverage architecture for SRAM under test.

3. SIMULATION RESULTS

We had tested our approach on 256x8 size SRAM. The results obtained by applying our approach to the ATPG of SRAM using MBIST and MBISRA is shown in the Fig 3.1, Fig 3.2, Fig 3.3 and Fig 3.4 respectively.

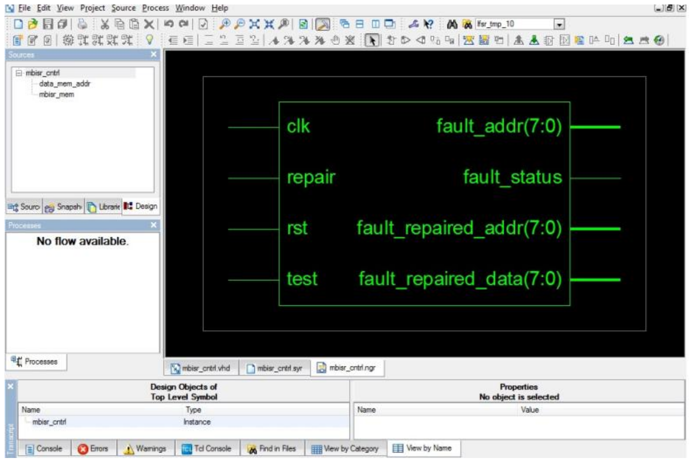

Fig 3.1: Technology schematic of memory test Controller.

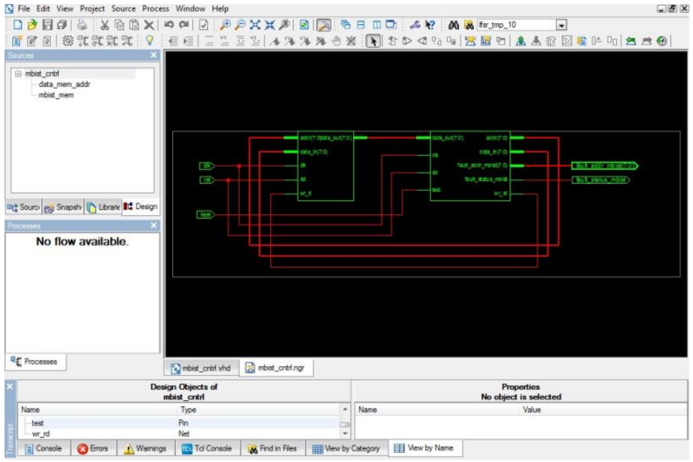

Fig 3.2: Technology schematic of MBIST and MBISRA Controller.

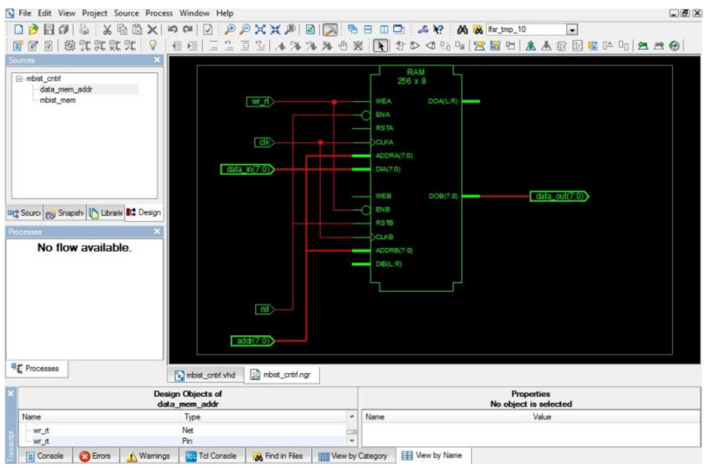

Fig 3.3: Technology schematic of MBIST.

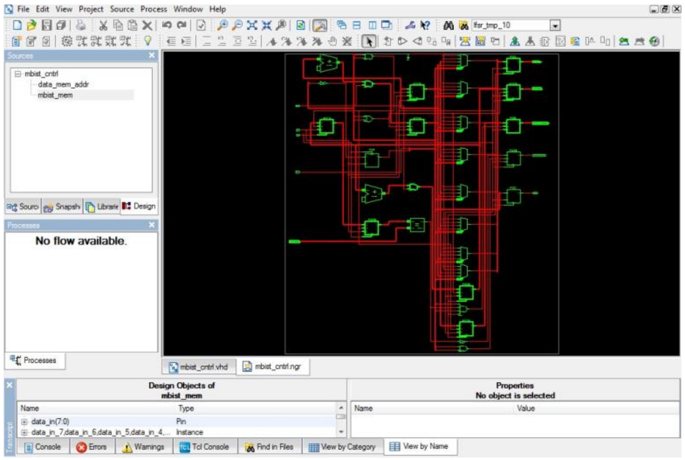

Fig 3.4: Technology schematic of MBISRA Controller.

The simulation results of the SRAM under test are shown in Fig 3.5; here the address bits will be decoded using addr pin of MBIST.

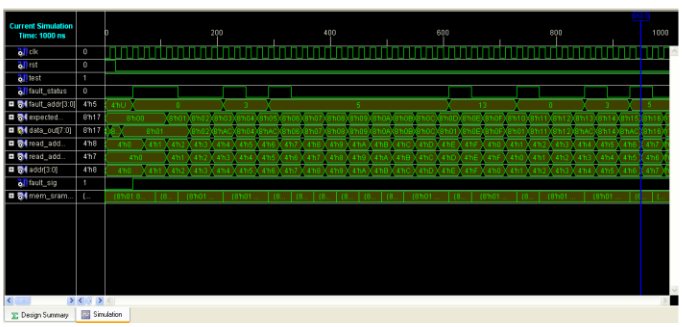

Fig 3.5: Simulation waveform of MBIST and MBISRA modules.

Fig: 3.5 show exactly the MBIST and MBISRA simulation. In MBIST module, during normal mode the watermarked bits will be stored in the address locations of the SRAM. In test mode, the watermarked bits will be compared with ATPG, if the data present in the SRAM memory address locations which we are testing is having any fault data fault_status pin will be 1 and if the data present in the address is correct in the address location of SRAM fault_status pin will be 0.

If fault_status is 1, then MBISRA module will be activated. Type of stuck-at fault in the SRAM memory location will be detected and correct data bits will be stored in the correct address location.

The algorithm of fault diagnosis and coverage is as follows:

```
// if (normal_mode = 1 and test_mode=0) then
// data_in_SRAM <= watermarked_bits;
// end if;
// if( test_mode=1 and normal_mode=0) then
// MBIST<=1;
// if(expected_value=data_in_SRAM) then
// fault_status<='0';
// else
// fault_status<='1';
// fault_addr<=read_addr_reg;
// end if;
// end if;
// if (fault_status<='1') then
// MBISRA <=1;
// fault_repaired_addr<=correct_addr_of_SRAM;
// fault_repaired_data<=correct_data_into_SRAM;
// end if;
// MBIST<=0;
// MBISRA <=0;
```

4. SYNTHESIS RESULTS

	Image watermarking MBIST-R module	**Complete watermarking with MBIST-R**
	HDL Synthesis Report	
RAMs	1	3
Adders/Subtractors(8-bit)	2	4
Registers	91	182
Comparators	1	2
Multiplexers	8	16
	Cell Usage	
BELS	48	151
Clock Buffers	1	1
IO Buffers	27	399
	Device utilization summary	
Selected Device	SPARTAN 3s1600efg484-5	SPARTAN 3s1600efg484-5
Number of Slices	57 out of 14752	135 out of 14752
Number of Slice Flip Flops	91 out of 29504	218 out of 29504
Number of 4 input LUTs	45 out of 29504	130 out of 29504
Number of bonded IOBs	28 out of 376	294 out of 376
Timing Summary : Minimum period	5.400ns	6.046ns
Total memory usage	206640 kilobytes	213236 kilobytes

Table-4.1: Synthesis Report of Our Proposed Method of Memory Testing.

Table 4.1 shows the synthesis report of our proposed method of memory testing. Synthesis results show that our MBIST technique has very high fault coverage and supports the theoretical analysis. The proposed method affects three aspect yield, area of chip & delay. By this approach we get the maximum (100%) yield whereas area is concern, as we are using spare device area will increase and due to increase in area the affect time delay of various nets.

CONCLUSIONS

Our proposed method of MBIST and MBISRA has shown good results in performing the testing process to an encoder in communication channel for image transmission through watermarking. Algorithm used to implement the testing phase utilized less time and less memory compared with the total watermarking technique. Simulated results had shown a high-end method of MBISRA to test any SRAM memory with less complexity. From synthesis results we can say that our proposed method of MBISRA utilized less space in SPARTAN 3s1600efg484-5 and fault coverage is improved alot comparably. From simulation and synthesis results we can say that our proposed method of memory fault coverage and its design VLSI architecture is efficient than the previous proposed method of MBIST for image watermarking RAM's.

FUTURE RESEARCH

Our proposed method of MBIST can be further extended to multiple-core RAM's and dual SRAM's. The design complexity of these in-build SRAM's will be reduced with our proposed method. Moreover, our technique is able to deal with combinational RAM's of great size. Further studies will be conducted to adapt it to the test-per-scan scheme.

REFERENCES

[1].S.Bhargav Kumar, S.Jagadeesh, Dr.M.Ashok, "*LFSR Based Watermark And Address Generation For Digital Image Watermarking*", International Journal of Computer & Organization Trendz- Volume2 Issue 3-2012, pp 73-79.

[2].W.A.S Wijesinghe, M.K Jayananda and D.U.J Sonnadara, *"Hardware Implementation of Random Number Generators"*, Proceedings of the Technical Sessions, 22 (2006) 28-38, Institute of Physics – Sri Lanka, pp.28-38.

[3].B. Rajan and S.Ravi, "*FPGA Based Hardware Implementation of Image Filter With Dynamic Reconfigurable Architecture*", in IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.12, December 2006, p. 121-127.

[4].Pankaj U.Lande, Sanjay N. Talbar and G.N. Shinde, "*FPGA Prototype of Robust Image Watermarking For JPEG 2000 With Dual Detection*", International Journal of Computer Science & Security (IJCSS), Volume (4) : Issue (2), pp-226-236.

[5].Yonatan Shoshan, Alexander Fish, Xin Li, Graham Jullien, Orly Yadid-Pecht, "*VLSI Watermark Implementations And Applications*", International Journal "Information Technologies and Knowledge" Vol.2 / 2008, pp-379-386.

[6].A.Mohamed Zuhair M.E., Lecturer,C. Mohamed Yousuf M.E., Lecturer, "*FPGA Based Image Security And Authentication In Digital Camera Using Invisible Watermarking Technique*", International Journal of Engineering Science and Technology Vol. 2(6), 2010, pp-1745-1751,

[7].Saraju P. Mohanty, Renuka Kumara C, and Sridhara Nayak, "*FPGA Based Implementation of an Invisible-Robust Image Watermarking Encoder*," CIT 2004, LNCS 3356, pp. 344–353, 2004.