



GLOBAL JOURNAL OF RESEARCHES IN ENGINEERING: F
ELECTRICAL AND ELECTRONICS ENGINEERING
Volume 17 Issue 2 Version 1.0 Year 2017
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

Modelling and Simulation of Microcode based Built-In Self Test Technology for Multiported Memory

By Prof. Dr. S. R Patil & Miss. Musle Dipali B.

Bharati vidyapeeths college of engg for women pune

Abstract- Now a day's embedded memory area and memory density is increasing. Due to this problem of fault is growing exponentially. It is necessary to detect and repair those faults. There are different methods to detect the faults present in memory. Asynchronous P-MBIST method is used to detect the faults. Simulation result includes asynchronous P-MBIST which detect mismatch of data occurred and it is shown by high fault pulse. To correct the detected fault redundancy circuit is used. This system is implemented using FPGA platform.

Keywords: built-in-self test (MBIST), built-in-self repair (BISR), asynchronous P-MBIST, microcode MBIST multiported memory, redundancy logic array, field programming gate array (FPGA).

GJRE-F Classification: FOR Code: 090699



Strictly as per the compliance and regulations of :



Modelling and Simulation of Microcode based Built-In Self Test Technology for Multiported Memory

Prof. Dr. S. R Patil^a & Miss. Musle Dipali B.^σ

Abstract- Now a day's embedded memory area and memory density is increasing. Due to this problem of fault is growing exponentially. It is necessary to detect and repair those faults. There are different methods to detect the faults present in memory. Asynchronous P-MBIST method is used to detect the faults. Simulation result includes asynchronous P-MBIST which detect mismatch of data occurred and it is shown by high fault pulse. To correct the detected fault redundancy circuit is used. This system is implemented using FPGA platform.

Keywords: built-in-self test (MBIST), built-in-self repair (BISR), asynchronous P-MBIST, microcode MBIST multiported memory, redundancy logic array, field programming gate array (FPGA).

I. INTRODUCTION

According to the ITRS 2015, Now a days the memory area is increasing than logic area in SOC's. Fig.1[1] shows that shrinking the technologies give rise to defect and new fault models. Thus, the new trends in Multiported memory testing will be driven by the following items.

BIST: The new fault and new defects eliminate using BIST. The only solution that allows at-speed testing for embedded memories, it has generate fault detection and coverage, the higher repair efficiency only detecting the faults is no longer sufficient for SOC's so memory repair is also necessary for this purpose both diagnosis and repair algorithm are required.

BISR: Combining BIST with efficient and low cost repair schemes in order to improve the yield and system reliability as well.

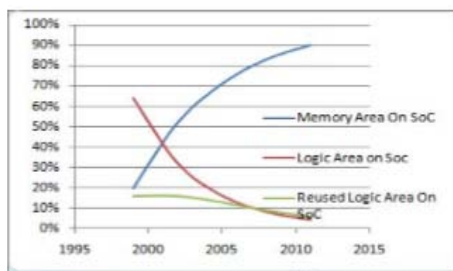


Fig.1: Embedded Memories in Future SOC's [1]

Auhtor ^a: Department of Electronics and telecommunication Bharati Vidyapeeth's College of Engineering for Women Pune, 43 Savitribai Phule Pune University, e-mails: srpatil44@gmail.com, dipali.musleb@gmail.com

There are two methods which are efficiently used. These methods are as follow:

1. **External Testing:** ATPG algorithm is used for external testing method ATPG algorithm consumes more area and to test vector generation is time consuming.
2. **Internal Testing:** BIST scheduling algorithm for internal testing it includes LFSR tuning, Test pattern generation. Whereas, BIST and Design for Test (DFT) is on same SOC and overcomes the limitation of the ATPG algorithm.

Microcode Asynchronous P-MBIST is implemented and compared with Synchronous P-MBIST. In Asynchronous M-BIST handshaking signals are used to communication between blocks of test controller. Whereas, in synchronous P-MBIST clocks are used to synchronize the test blocks on SOC. The BIST having two modes: Normal mode and scan mode The power consumption during test mode is more than the power consumption during normal mode. Hence to choose power efficient BIST method for multiported memory testing.

II. RELATED WORK

Yash Jyothi et al.[1] introduced Asynchronous P-MBIST method for fault detection and repair of multiported memory. The main advantage is to get power efficient P-MBIST, it is being proposed to use handshaking signals instead of using clock and implemented on FPGA. Clocks are used to synchronize the test blocks on chip and handshaking signals are used to communicate between blocks of test controller.

Dr. R.K. Sharma et al.[2] introduced Redundancy Array Logic and BISR mechanism used to low cost repair the fault and to improve the system reliability.

Bo-Cheng Charles Lai et al.[4] introduced designs of multiported memories that leverage BRAMs have been proposed to attain better utilization of FPGA resources as well as system performance. BRAMs in an FPGA can support two access ports that can be used as either a read or a write port.

III. BUILT-IN SELF TEST (BIST)

Memory Built-In Self Test is test circuitry used to test on-chip memory devices. It contains finite state machine to generate test vectors to test memory during test mode. Test coller includes Address, Data, read/write control. This address, data and read/write control signal is given as input to comparator and multiplexer. Multiplexer uses two modes of operations: *Normal mode*-In normal mode memory acts as a normal memory. *Test Mode*-During test mode test coller provides data and address to memory for testing purpose. In test mode while performing read operation data from test coller and output of memory are given to comparator for fault detection. Then, comparator compares test vectors with memory output. If, memory is working properly then, No Fault is detected and it is declared as memory is fault free but, if fault pulse becomes high it shows that memory under test is faulty.

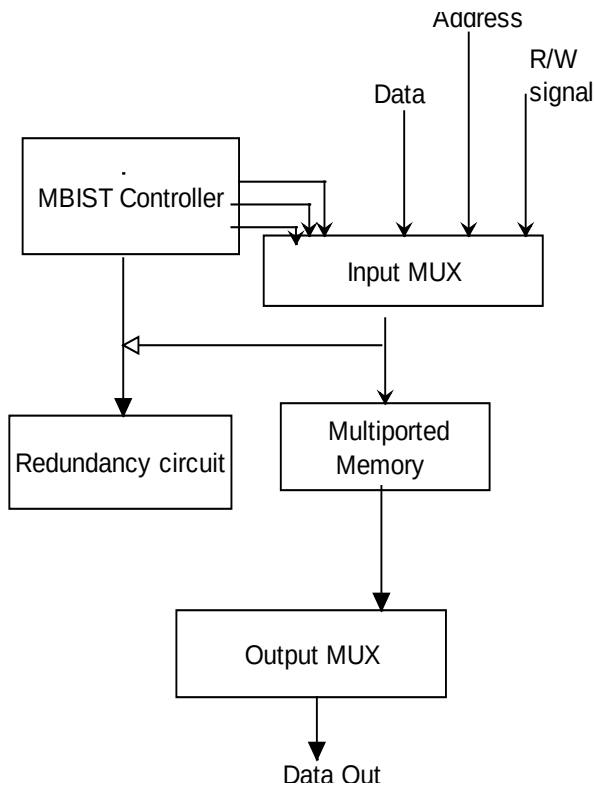


Fig. 2: Structure of P-MBIST

IV. IMPLEMENTATION OF ASYNCHRONOUS P-MBIST

The block diagram shown in Fig.3 [1] the asynchronous P-MBIST contains handshaking signals instead of using clock. Request and acknowledge signals perform communication in Asynchronous P-MBIST.

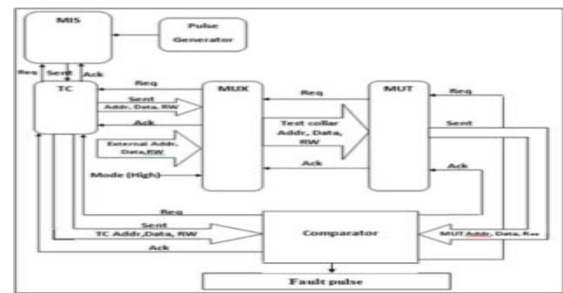


Fig. 3: Block Diagram of Asynchronous P-MBIST [1]

The control circuitry contain following blocks: Microcode Instruction storage unit, Test collar, comparator, Clock Generator. The Test Collar circuitry consists of Address Generator, RW Control and Data Control. Common Clock is used for Synchronous MBIST. In Microcode Instruction Storage (MIS), 4 instruction operations are stored that are R0, R1, W0, W1 as shown in Table I. [1]

Table No.1: Microcode Instruction [1]

Instruction	Description
R0	Read is low so no read operation occurs
R1	Read is high so read operation occurs
W0	Write is low so no write operation occurs
W1	Write is high so write operation occurs

V. FLOWCHART OF ASYNCHRONOUS P-MBIST

The flow of asynchronous P-MBIST is as shown in fig .4, fig. 5 and fig. 6. It follows the following steps:

- If there is request from Microcode Instruction Storage (MIS) to Pulse generator (PG), then PG sends Start Pulse to MIS to start the asynchronous operation. In next step MIS checks for request from Test Collar (TC), if so it sends microcode instructions to TC. TC then sends the acknowledgement signal to pulse generator indicates that TC is getting instructions from MIs.
- Output of TC Data which includes test address, corresponding data and R/W control signals send to input multiplexer (MUX) if MUX sends high pulse of request. Multiplexer is originally not a multiplexer but concept used for this block is according to working of normal multiplexer. Here mode signal is select line for MUX.
- If mode=1 then PMBIST works in test mode then it selects TC data. If mode=0 then it is in normal mode, during this mode test circuitry is in ideal condition and memory works as normal memory without under test and sends output of external Address, corresponding data and R/W control signal. When MUX is under test mode then it sends TC data to memory if request signal is high from memory to MUX.

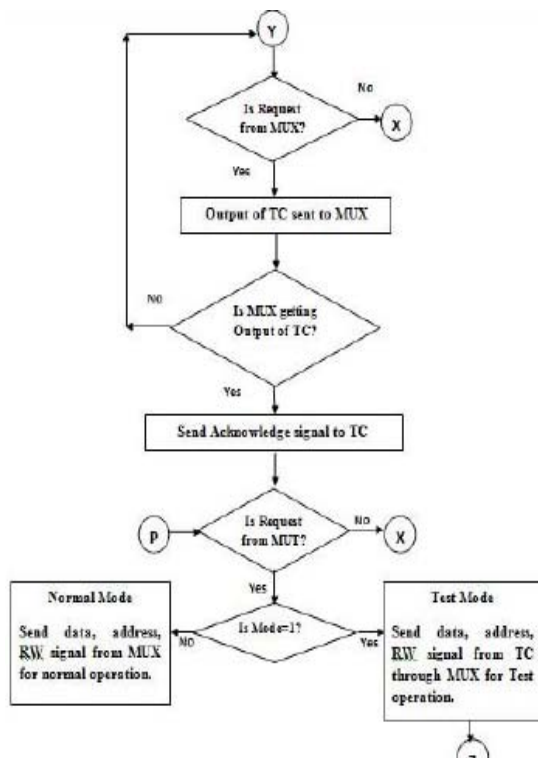


Fig. 4: Flowchart Step-I

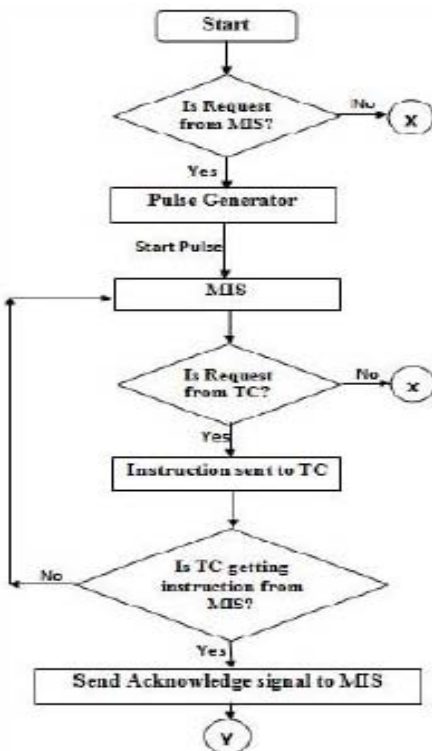


Fig. 5: Flowchart Step-II

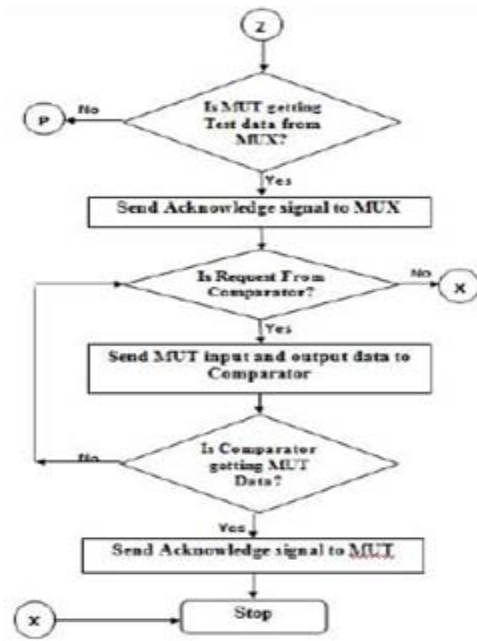


Fig. 6: Flowchart Step-III

- VI. Comparator compares inputs provided by TC and memory output data during test mode and gives high fault pulse if any mismatch happens between these inputs

V. ASYNCHRONOUS P-MBIST SIMULATION

The simulation of asynchronous P-MBIST is depending upon request signal and mode. Output Response: Normal Mode: When mode=0, MUX selects normal mode, then test controller is off so as comparator is off. At this moment memory acts as normal memory without under test and generates external address and data if rw=0 (i.e. reads address and corresponding data). During this mode as comparator is off, no fault pulse is generated. In this mode, memoutdata (memory output) data is data (External data) provided for normal memory operation. Test Mode: If mode=1, Memory is under test. If req=1, microcode instruction storage generates Read/Write control for test collar. Test collar writes data if tcw=0 otherwise read to memory through MUX. The data from test collar and output of memory is compared in comparator. If mismatch happens at output of comparator, comparator generates fault pulse=1 otherwise fault pulse=0 [1].

VI. REPAIR MODULE

Fig.8. Shows the Repair Module including the redundancy array and output multiplexer and its interfacing with the existing BIST module.

Fig.7 An array of redundant words placed in parallel with the memory. The following interface signals are taken from the MBIST logic: 1) A fault pulse indicating a faulty location, 2) Fault address, 3)

Correct data that is compared with the results of Memory under test. The MBISR logic used here can function in two modes.

a) Mode 1: Test & Repair Mode

In this mode the input multiplexer connects test collar input for memory under test as generated by the BIST controller circuitry. As faulty memory locations are detected by the fault diagnosis module of BIST Controller, the redundancy array is programmed. A redundancy word is as shown in Figure. 7 The fault pulse acts as an activation signal for programming the array. The redundancy word is divided into three fields. The FA (fault asserted) indicates that a fault has been detected. The address field of a word contains the faulty address, whereas the data field is programmed to contain the correct data which is compared with the memory output. The IE and OE signals respectively act as control signals for writing into and reading from the data field of the redundant word.

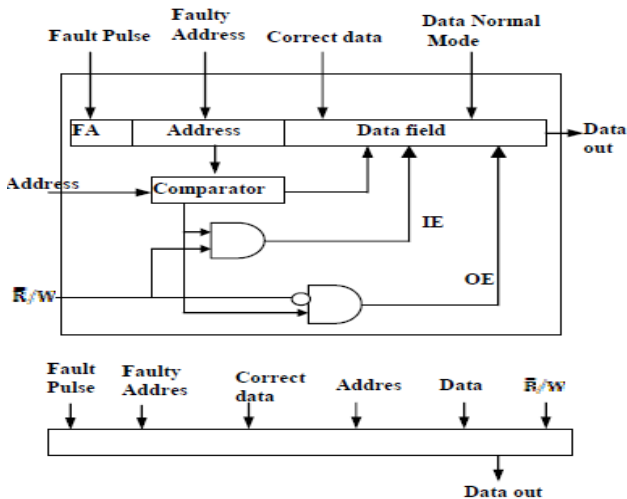


Fig. 7: Redundancy Word Line [2]

b) Mode 2: Normal Mode

During the normal mode each incoming address is compared with the address field of programmed redundant words. If there is a match, the data field of the redundant word is used along with the faulty memory location for reading and writing data. The output multiplexer of Redundant Array Logic then ensures that in case of a match, the redundant word data field is selected over the data read out (= 0) of the faulty location in case of a read signal. This can be easily understood by the redundancy word detail shown in Figure 7.

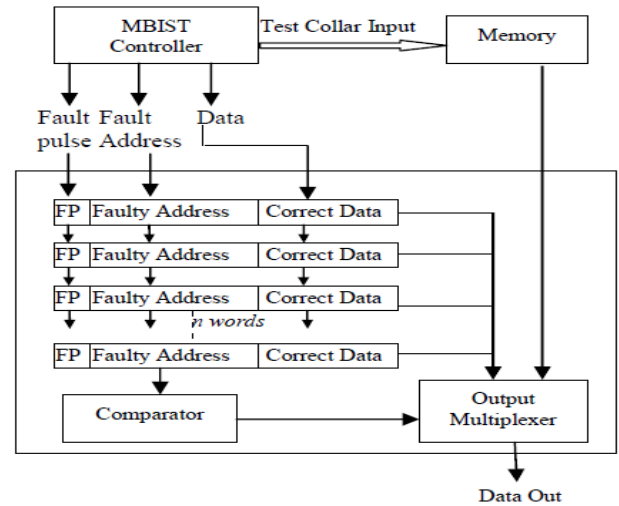


Fig. 8: Redundancy Array Logic [2]

Fig.8.Shows the simulated waveform of fault diagnosis module, magnified at seventh pulse to indicate how signals like 'fault pulse', 'faulty location addresses' and 'correct data' are generated by this module for successful interfacing with the Redundancy Array logic.[2]

VII. RESULTS

1. Single port Memory

The single port memory support single read and write operation.



Fig.9: Result of Single port Memory

2. Dual port Memory

The single port memory support dual read and write operation.



Fig.10: Result of Dual port Memory

3. Multiported Memory

It requires significant amount of BRAMs to implement a memory module that supports multiple read and write ports.



Fig. 11: Result of Multiported Memory

4. Synchronous P-MBIST

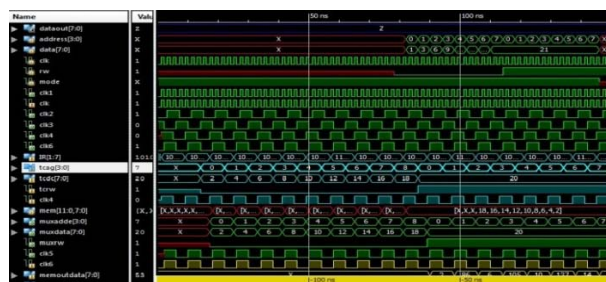


Fig. 12: Simulation Result of Synchronous P-MBIST

5. Asynchronous P-MBIST

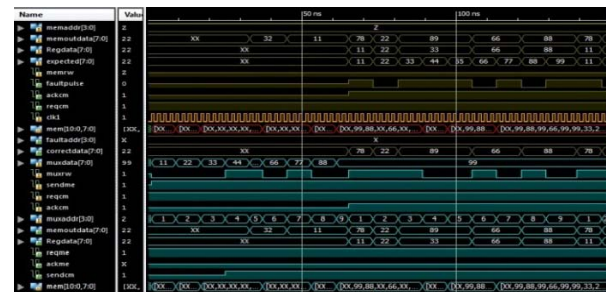


Fig. 13: Simulation Result of Asynchronous P-MBIST

6. Redundancy Array Logic

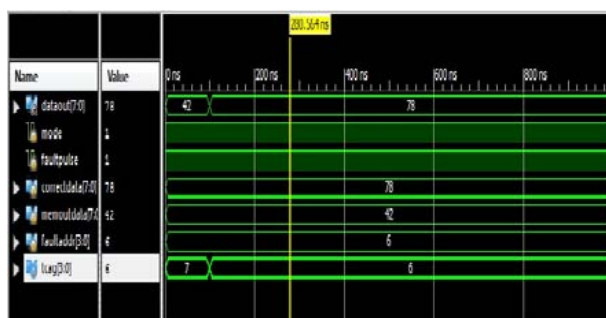


Fig. 14: Simulation Result of Redundancy Logic Array

VIII. CONCLUSION

In simulation results of different types memories where implemented. The efficient BRAM-based multiported memory designs on FPGAs. The existing design methods require significant amounts of BRAMs to implement a memory module that supports multiple read and write ports. In simulation results of different methodology of P-MBIST where implemented. To synchronize the blocks synchronous PMBIST uses clock due to this it consume more power because of excessive switching activity and to communicate between the blocks of MBIST asynchronous MBIST uses handshaking signals and it consume less power. The word redundancy uses spare words in place of spare rows and columns. It stores faulty location address immediately supporting on-the-fly fault repair memory.

REFERENCES RÉFÉRENCES REFERENCIAS

1. HassenMestiri, FatmaKahri BelgacemBouallegue, Mohsen Machhout, "A high speed AES design resistant to fault injection attacks", Microprocessors and Microsystems journal,2016 Elsevier, pp.47-55
2. Dr. R.K. Sharma Aditi Sood, "Modeling and Simulation of Multi-Operation Microcode based Built-In Self Test for Memory Fault Detection and Repair", IEEE Annual Symposium on VLSI, 2010
3. Prof.Vinod Kapse, Mohammed Arif, "Optimization of Microcode Built-In Self Test By Enhanced Faults Coverage for Embedded Memory", IEEE Students' Conference on Electrical, Electronics and Computer Science, 2012.
4. Bo-Cheng Charles Lai, Member, IEEE, and Jiun-Liang Lin,"Efficient Designs of Multiported Memory on FPGA",ieee transactions on very large scale integration (vlsi) systems,2016.
5. G. A. Malazgirt, H. E. Yantir, A. Yurdakul, and S. Niar,"Application specific multi-port memorycustomization in FPGAs," in Proc. IEEE Int.Conf. Field Program. Logic Appl. (FPL), Sep. 2014, pp. 1–4.
6. Sandra Irobi Zaid Al-Ars Said Hamdioui.Memory Test Optimizationfor Parasitic Bit LineCoupling in SRAMs. IEEE International Test Conference, 2010.
7. N. Z. Haron, S.A.M. Junos, A.S.A. Aziz, "Modelling and Simulation of Microcode Built-In Self test Architecture for Embedded Memories", In Proc. of IEEE International Symposium on Communications and Information Technologies pp. 136-139, 2007.
8. S. Hamdioui, G.N. Gaydadjiev, A.J. van de Goor, "State-of-art and Future Trends in Testing Embedded Memories", International Workshop on Memory Technology, Design and Testing (MTDT'04),2004.
9. P. Bernardi, M. Grosso, M. S. Reorda, and Y. Zhang, "A programmable BIST for DRAM testing

and diagnosis," in Test Conference (ITC), 2010 IEEE International, 2010, pp. 1-10.

10. C. E. LaForest and J. G. Steffan, "Efficient multi-ported memories for FPGAs," in Proc. 18th Annu. ACM/SIGDA Int. Symp. Field Program. Gate Arrays, 2010, pp. 41-50.
11. Xilinx. 7 Series FPGAs Configurable Logic Block User Guide, accessed on May 30, 2016. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf.
12. C. E. LaForest, M. G. Liu, E. Rapati, and J. G. Steffan, "Multi-ported memories for FPGAs via XOR," in Proc. 20th Annu. ACM/SIGDA Int. Symp. Field Program. Gate Arrays (FPGA), 2012, pp. 209-218.