

FPGA based complex test pattern generation for high speed fault diagnosis in memory blocks

S. Charitha Pal and Mrs. J. Shailaja

¹Spoorthy Engineering College, Student, M. Tech, ECE, E.C.E Dept., Hyderabad, India. ²Spoorthy Engineering College, Asst. Prof. E.C.E Dept., Hyderabad, India

¹Email:charitha.srivari@gmail.com ² Email:shailu.boddupally@gmail.com

ABSTRACT:

The memory blocks testing is a separate testing procedure followed in VLSI testing. The memory blocks testing involves writing a specific bit sequences in the memory locations and reading them again. This type of test is called March test. For example the March C- test has the following test pattern. $\{\uparrow(w0); \uparrow(r0, w1); \uparrow(r1, w0); \downarrow(r0, w1); \downarrow(r1, w0); \uparrow(r0)\}$ However no test setup is developed so far for testing the memory blocks inside the FPGA. The BRAM blocks of FPGA are designed to work at much higher frequency than the FPGA core logic. Hence testing the BRAMs at higher speed is essential. Hence the proposed work develops a memory testing tool based on March tests for FPGA based BRAM (block RAM testing). A PC based GUI tool shall send command to FPGA using serial port for selecting the type of test. The FPGA core gets the command through UART and performs the appropriate and sends the test report back to PC.

Keywords: BRAM-Block Ram, FPGA, VLSI-Very Large Scale Integration.

I INTRODUCTION:

The cost of verification and test for nowadays circuits represents an important part of the total IC final price. Hence, the domain of test represents a cornerstone for the industry. Recently, the advances in semiconductor memory technologies have become more complex resulting in a rapidly increasing transistor per memory design.

New design techniques enable a higher memory capacity implementation on a fixed die size. However, larger memory capacities require more extensive testing. Inevitably memory testing time increases the fault model to effectively model the variety of physical failures that could occur because of interference between closely packed cells. On the other hand, the more compact size will produce more defects during chip manufacturing, pushing yields down.

Therefore, memory testing and repairing will be more important in the future memory chips. Testing and diagnosis techniques play a key role in the advancement of semiconductor memory technologies. The challenge in obtaining reliable failure detection has created intensive investigation

Copyright © 2013 SciResPub.

on efficient testing and diagnosis algorithms for better fault coverage and diagnostic resolution.

There are a number of test techniques that have been well studied. Many test algorithms were proposed based on functional-level fault models using manual technique analysis. Although March Test Algorithm (MTA) is one of the best solutions and widely use in testing memories, due to the technology advance there are more new fault models will be introduced. Therefore, to overcome this coming fault models a new technique is required to develop which can be automatically analyzed and added the new technique.

This aims to propose a new solution for researchers to find an efficient test and diagnosis algorithm in short time. A combination march-based test algorithm will be implemented for this purpose. Universities and industry involved in memory Built-in-Self test, Built-in- Self repair and Built-in-Self diagnose will benefit by saving a few years on research and development due to the fact that the manual and automatic test procedure developed in this work is compatible, expandable for BRAM memory testing.

II MARCH TEST GENERATOR:

Memory testing may be considered as a full disciplinary subject. Commonly, test sequences or test algorithms for memories are known under the name of March tests. Every March test has specific capabilities that allow revealing the typical defects of memories [3]. A typical didactic test bench has to allow not only the implementation of March tests existing in the literature but also the creating of new test algorithms. A March test consists of a sequence of March elements. A March element has a certain number of operations (or March primitive) that must be applied to all memory cells of an array. Thus, $\uparrow(r0;w1)$ is a March element and $r0$ and $w1$ are March primitives. The addressing order of a March element can be done in an up (\uparrow), down (\downarrow) way or (\updownarrow) if the order is not significant. A March primitive can be a write 1 ($w1$), write 0 ($w0$), read 1 ($r1$) and read 0 ($r0$) that can be performed in a memory cell.

Here, we introduce, for example, the March C- :

$\{\uparrow(w0); \uparrow(r0,w1); \uparrow(r1,w0); \downarrow(r0,w1); \downarrow(r1,w0); \uparrow(r0)\}$

This well-known March test allows to detect all the stuck@ and transition fault of a memory cell array, as well as all address decoder faults and coupling (interaction between two cells) faults. A way to create a function allowing the implementation of any March test. A 274 bits register is needed to memorize the march test data

III FPGA THEORETICAL ANALYSIS:

FPGAs comprise an array of uncommitted circuit elements, called *logic blocks*, and interconnect resources, but FPGA configuration is performed through programming by the end user.

An illustration of a typical FPGA architecture appears in Figure 2.3. As the only type of FPD that supports very high logic capacity, FPGAs have been responsible for a major shift in the way digital circuits are designed.

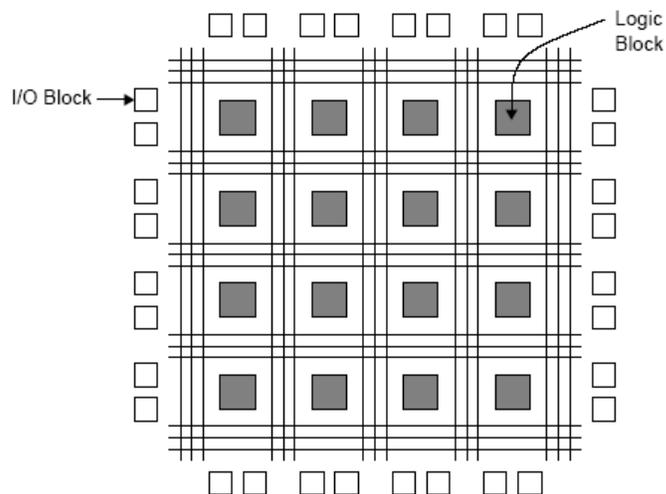


Figure 1: structure of an FPGA

Field programmable gate arrays (FPGAs) are a popular choice among VLSI devices, any logical circuit can be implemented into the FPGA at low cost. It consists of an array of configurable logic blocks (CLBs), programmable interconnect and programmable Input/output blocks, Block Random Access Memories (BRAMs), a multiplier, a Digital Clock Manager. Many methods have been proposed to test FPGAs. In some works, the circuits under consideration are programmed FPGAs, in which logic circuits have been implemented. Since an FPGA can be programmed in many different ways, this method is not applicable to manufacturing time testing, as we do not know the final configuration. The FPGA architecture consists of five fundamental programmable functional elements:

1. Configurable Logic Blocks (CLBs) contain flexible Look-Up Tables (LUTs) that implement logic plus storage elements used as flip-flops or latches. CLBs perform a wide variety of logical functions as well as store data.

2. Input/output Blocks (IOBs) control the flow of data between the I/O pins and the internal logic of the device. Each IOB supports bidirectional data flow plus 3-state operation. It supports variety of signal standards, including high-performance differential standards. Double Data-Rate (DDR) registers are included.

3. Block RAM provides data storage in the form of 18-Kbit dual-port blocks.

4. Multiplier Blocks accept two 18-bit binary numbers as inputs and calculate the product. The Spartan-3E devices provide 4 to 36 dedicated multiplier blocks per device. The multipliers are located together with the block RAM in one or two columns depending on device density.

5. Digital Clock Manager (DCM) Blocks provide self-calibrating, fully digital solutions for distributing, delaying, multiplying, dividing, and phase-shifting clock signals.

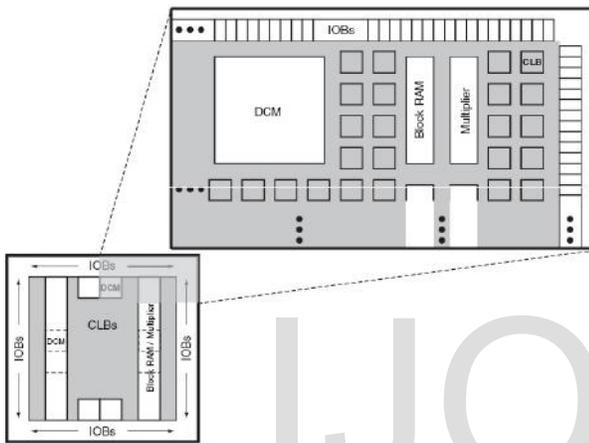


Figure 1.2: Programmable elements of FPGA.

IV BLOCK RAM (BRAM):

Spartan-3E devices incorporate 4 to 36 dedicated block RAMs, which are organized as dual-port configurable 18 Kbit blocks. Functionally, the block RAM is identical to the Spartan-3 architecture block RAM. Block RAM synchronously stores large amounts of data while distributed RAM, previously described, is better suited for buffering small amounts of data anywhere along signal paths. This section describes basic block RAM functions

4.1 The Internal Structure of the Block RAM:

The block RAM has a dual port structure. The two identical data ports called A and B permit independent access to the common block RAM, which has a maximum capacity of 18,432 bits, or 16,384 bits with no parity bits. Each port has its own dedicated set of data, control, and clock lines for synchronous read and

write operations. There are four basic data paths, as shown in Figure:

1. Write to and read from Port A
2. Write to and read from Port B
3. Data transfer from Port A to Port B
4. Data transfer from Port B to Port A

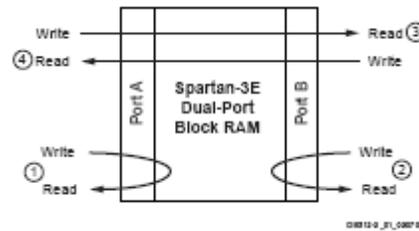
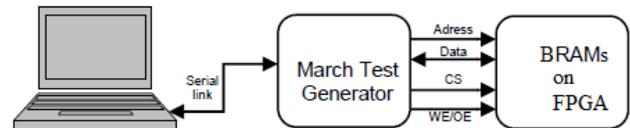


Figure 2: Block RAM Data Paths

V TEST BENCH ARCHITECTURE:

Our test bench architecture for memories is composed on one computer, a versatile March test generator a serial interface (for the communication between the generator and the computer) and a deck containing memories under test,

Figure 3: Test bench architecture



A user interface, presented in Fig. allows students to choose or set a specific March test of the literature (March A, March C-, Matt, Matt +) or introduce a new one custom) The chosen March test is uploaded through The serial connection to the programmable test generator and then applied to each memory on the deck.

If no fault is detected, the programmable generator returns a positive acknowledgement on the four me the opposite case occurs, i.e. when a reading1) does not return the expected data, the test Bench returns the following data: the failing address , failing memory, the failing march element and operation. Only the knowledge of this information allows the identification of physical defects beside the observed fault, or at least to Make reasonable suppositions.

5.1 Need for testing BlockRAMs:

Memory blocks are fabricated using high dense circuits, transistors are closely packed, and intentionally they are very sensitive devices and work with very highspeed read & write sequences. When shipping out from the factory, that time they definitely do testing. When deployed into some field, there years together these have to work. Block RAM is a block on FPGA which will be always in usage. First keeping in BRAM and reading in an FPGA. Hence BRAM is always active. Here there is a need to test BRAM logic whether it is working correctly or not FPGA core is capable of running at 50MHz but the BRAM capable of running at 200MHz. something running at higher frequency means it is sensitive and assertable to errors. Certain errors will come out when you run at higher clock frequency. Over aging errors are introduced in BRAM.

Table 1: March Test Sequences

MarchC-	$\downarrow(W0)\uparrow(R0,W1)\uparrow(R1,W0)$
MarchSR	$\downarrow(W0)\uparrow(R0,W1,R1,W0)\downarrow(R0,R0)$
MarchB	$\downarrow(W0)\uparrow(R0,W1,R1,W0,R0,W1)$

The above table shows the implemented March test sequences. If a memory block gets through all these three test sequences, we can say that the memory block is fault free. Here the BlockRAM is treated as PCore of 16bit width and depth of 256, which is generated by Xilinx Core Gen with Single port configuration.

5.2 Marchtest Generator:

The March test generator is used to test the block RAMs by generating the appropriate test sequences. This block generates all the above said March test sequences to detect functional faults in the Block RAMs on FPGA. The test sequences are generated so as to satisfy the conditions needed to detect the functional faults like SAFs, TFs, CFs, and AFs in Block RAMs.

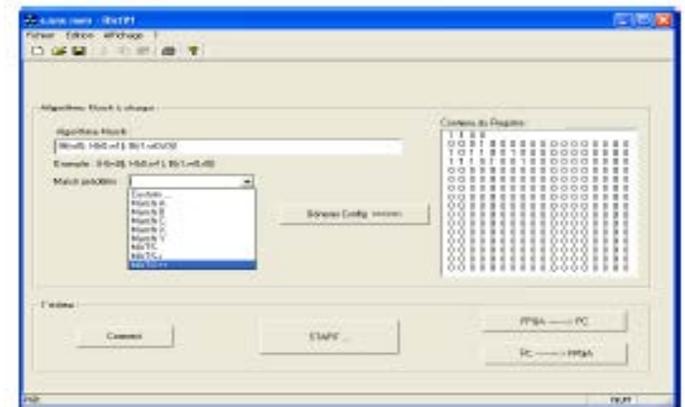


Figure 4: User interface

VI EXPERIMENTAL RESULTS:

The written input values into BRAM are read with expected values. Here the contents of BRAM are read without any failure. Hence March test C- is Successful and BRAM is fault Free.

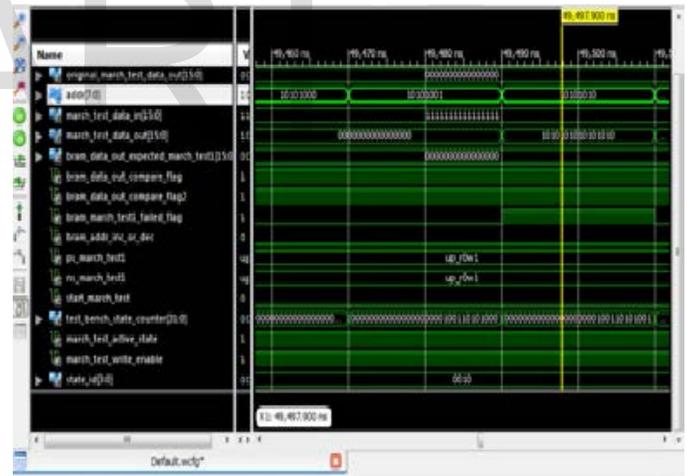


Figure 5: Data output of BRAM using March C-

The written input values into BRAM are read with expected values except at address "aa". Here the contents of BRAM are read with fault. Hence March test C- is Successful and BRAM is faulty.

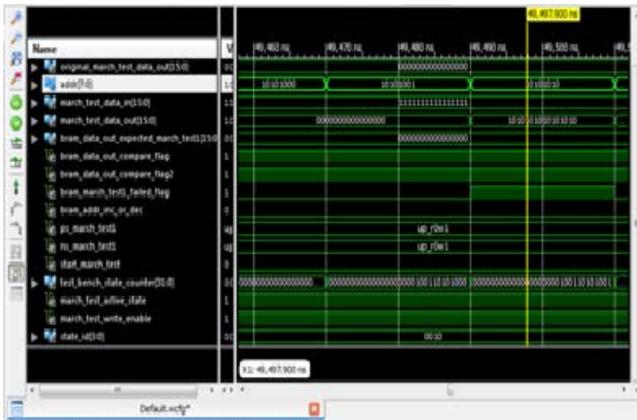


Figure 5.1: Data output of BRAM using March C with Fault insertion

The written input values into BRAM are read with expected values. Here the contents of BRAM are read without any failure. Hence March B test is Successful and BRAM is fault Free.

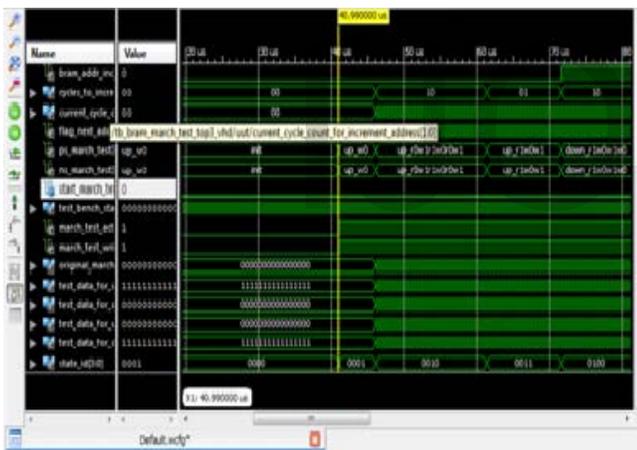


Figure 5.2: Data output of BRAM using March B

The written input values into BRAM are read with expected values except at address “aa”. Here the contents of BRAM are read with fault. Hence March B test is Successful and BRAM is faulty.

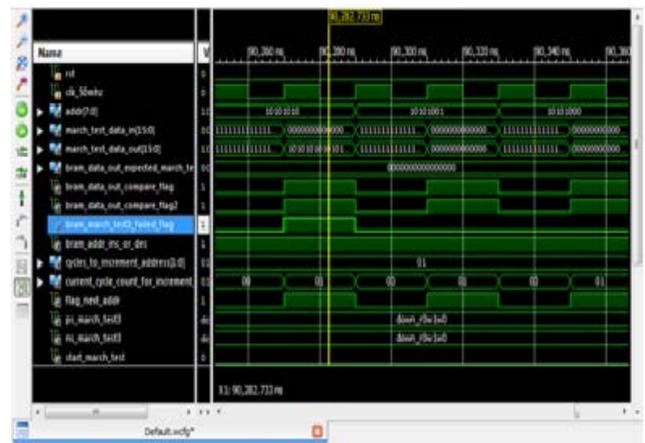


Figure 5.3: Data output of BRAM using March B with fault insertion

VII CONCLUSION:

With technology scaling, process variations result in functional failures in memory systems. In this work, physical failure mechanisms in BRAM on FPGA boards are analyzed and classified into the established logic fault models. March tests sequences are compared and optimized to target these emerging failure mechanisms. March C - test sequence, the memory test time is reduced.

VIII ACKNOWLEDGEMENT:

I would like to articulate my profound gratitude and indebtedness to Prof. Dr. V. CHANDRA MOULI, Principal, Spoorthy Engineering College and Dr. SHAIK ABDUL MUZEER, Head of the Department, Spoorthy Engineering College , for guiding and encouraging me in all aspects.

I wish to extend my sincere thanks to all the faculty members of Spoorthy Engineering College.

REFERENCES:

[1] Versatile march test generator for hands-on memory testing laboratory, Galliere, J.; Dilillo, L.; Polytech. Montpellier, Univ. of Montpellier, Montpellier, France Microelectronic Systems Education (MSE), 2011 IEEE International.

[2] Efficient Testing of SRAM With Optimized March Sequences and a Novel DFT Technique for Emerging Failures Due to Process Variations, Qikai Chen, Hamid Mahmoodi, Swarup Bhunia, and

Kaushik Roy, IEEE Transactions on Very Large Scale integration(VLSI) Systems , VOL. 13, NO. 11, NOVEMBER 2005.

[3] A.J. Van de Goor, "Testing Semiconductor Memories: theory and practice," John Wiley & sons, ISBN 0 - 471 -92586 -1, 1991.

[4] A.J. Van De Goor, 1993, "Using March Test to test SRAMs", IEEE design of computers, USA.

Authors Profile:



IJOART

S. Charitha Pal presently

pursuing Master degree in ECE at Sphoorthy Engineering college, Hyderabad, India. She has obtained her bachelor's degree in ECE in the year 2010 from Sri Sai Jyothi Engineering college, Hyderabad, India .

J. Shailaja is an assistant professor of Electronics and communications engineering at Sphoorthy Engineering college, Hyderabad. She had done her masters ME in MVSR Engineering College, and bachelor's in Shadhan Engineering College.