LOW POWER MIRROR SEQUENCE MARCH TEST ALGORITHM FOR STATIC RANDOM ACCESS MEMORIES

G RAJESH KUMAR, K BABULU

1,2JNTUK, Kakinada, India

Abstract—Low power design has emerged as the most essential field of work in VLSI system design. At the same time power dissipation in testing became more critical while designing a testable integrated circuit. During testing the design will consume much more power compared with normal functional mode. A novel Static Random Access Memory Testing scheme is presented in this paper, which can test memories using March test algorithms with a great reduction in test power. Testing in March algorithms is done marching from top to bottom and from bottom to top in a sequential manner. Like this each byte in the memory is tested multiple number of times. Sequencing in these testing algorithms is of binary sequencing. Because of the binary sequencing multiple number of bits toggle in each address sequence. If the binary sequencing is replaced with Single bit change address sequencing, number of transitions in the sequence reduces. There by power consumption in the testing reduces enormously. March Y based Built in Self-Test (BIST) algorithm is described using Verilog HDL language and is simulated using Xilinx ISE. Spartan-3E FPGA is used to implement the architecture.

Index Terms—Built in Self-Test (BIST), March Test algorithm, Memory Testing, Static Random Access Memory (SRAM)

I. INTRODUCTION

A circuit defect may lead to a fault, a fault can cause a circuit error, and a circuit error can result in a system failure. Manufacturing defects are physical (circuit) defects introduced during manufacturing that cause the design to fail to function properly in the device, on the printed circuit board (PCB), or in the system or field. These manufacturing defects can result in static faults (such as stuck-at faults) or timing faults (such as delay faults). There is general consensus with the rule of ten, which says that the cost of detecting a faulty device increases by an order of magnitude as we move through each stage of manufacturing, from device level, to board level, to system level, and finally to system operation in the field. Fault diagnosis and location in semiconductor Random Access Memories (RAM) are of prime importance in connection with the increasing density and dominating portion of embedded memories in system-on-chips (SOC). Manufacturing defects should be detected, diagnosed and located for further repair in order to improve the product quality, reliability and yield. Built-in self-test (BIST) is a design technique in which, parts of a circuit are used to test the circuit itself. Built-in self-test is the capability of a circuit (chip, board, or system) to test itself. BIST represents a merger of the concepts of built-in test (BIT) and self-test, and has come to be synonymous with these terms. The related term built-in-test equipment (BITE) refers to the hardware and/or software incorporated into a unit to provide DFT or BIST capability. In present day scenario low power digital circuit design has emerged as a principal theme in electronics industry. Previously design engineers more concentrate on area, cost and performance. In present scenario designers concentrating more on portable devices. Because of the portability, area is one of the major important factor to be considered while designing digital circuits.

Fig 1: Block diagram representing Memory BIST
This paper reviews some power modeling techniques, issues arise because of excessive power. This paper also describes March Y based architecture and the low power technique adopted to March Y architecture. Power consumption can be reduced by considering any one or combination of three factors called voltage, physical capacitance and data activity. Mainly data activity is considered and made proper changes in data activity to reduce power consumption during testing. Switching activity mainly responsible for power consumption. In order to reduce power consumption switching activity should be reduced. The data activity is measured by the switching activity. Switching in the sense a signal carrying data, changing from logic one to logic and vice versa. A logic method is proposed to reduce the switching activity. This paper concludes with the analysis of the low power consumption technique proposed and also suggests the need of developing low power and high performance test architectures.

This paper is organized as follows: Section 2 presents a review of power modeling in integrated circuits; Section 3 discusses issues arise due to high power consumption in integrated circuits, while Section 4 presents the March algorithms related to Memory testing. Section 5 gives a clear idea on the proposed method of low power testing; Section 6 presents the HDL implementation and Section 7 shows logic simulation results and the comparison statements related to power consumption during testing. Section 8 concludes this manuscript.

II. POWER MODELING

In the process of changing the output (from 0 to 1), a charge \( Q = C_L \cdot V_{dd} \) is delivered to the load. The power supply must supply this charge at voltage \( V_{dd} \) so the energy supplied is \( Q \cdot V_{dd} = C_L \cdot V_{dd}^2 \). However, the energy stored on a capacitance \( C_L \) charged to \( V_{dd} \) is only half of this (i.e., \( \frac{1}{2} \cdot C_L \cdot V_{dd}^2 \)). The other half must be dissipated by the PMOS transistors in the pull-up network. There are three components to the power consumed by the logic gate: (1) the dynamic power, because of the charge of capacitance \( C_L \); (2) the short-circuit power, because of the short circuit between power and ground during switching; and (3) the leakage power. The main component is the dynamic power, which still represents a significant fraction of the total power consumption despite the proportional increase of the other two components with technology improvements. This dynamic power consumption occurs during the transition from 0 to 1 on the gate output as current flows between power and ground. The dynamic power consumption of the logic gate during the time interval \([0,T]\) can finally be expressed as follows:

\[
P_{\text{dyn}} = \frac{1}{2} \cdot C_L \cdot V_{dd}^2 \cdot N \cdot 1/T
\]

Where \( C_L \) is load capacitance. \( V_{dd} \) is the supply voltage, \( N \) represents number of rising or falling transitions and \( T \) represents the time period. A circuit composed of \( N \) nodes and a test sequence of length \( L \) is applied than the energy consumed at a node during time interval \( t \) is given by:

\[
E = \frac{1}{2} \cdot S \cdot F \cdot C_o \cdot V_{dd}^2
\]

Where \( C_o \) is the minimum output capacitance of the circuit and \( S \) is the average number of transitions during the time interval \( t \).

III. ISSUES ARISE BECAUSE OF HIGH POWER CONSUMPTION

Now a days it is very much required that the chip performance and the number of functions that an IC can perform should be increased. For such a high computing integrated circuits it is very much expensive to provide packaging and cooling arrangements. As chip power consumption increases, core power should be dissipated through the packaging. So expensive packaging and cooling arrangements must be provided. In addition to cost, there is the issue of reliability. High power consuming integrated circuits dissipate more heat resulting many chip failures such as silicon failure. Every 10°C increase in operating temperature roughly doubles a component’s failure rate. Unless the power consumption of integrated circuits is reduced the heat dissipated may affect the system behavior as well as the reliability.

IV. MARCH ALGORITHMS

Embedded memories are the important design elements in many System on Chip (SoC). Almost all SoC’s have memories in different formats. Embedded memories greatly affects chip area. Physical faults may be observed in memories. These memories they should be tested. In normal working mode only one address can be accessed at a time. Some of the locations may be accessed rarely. Built in Self-Test structure for memories is highly desirable. Memory test algorithm is implemented on a BIST controller is called Memory BIST (MBIST). March test algorithms are used to test memories for high speed and highly efficient testing with good fault detection. Like most of the algorithms, begins by writing a background of zeroes. Then it reads the data at the first location and writes a 1 to that address. It continues this read/write procedure sequentially with each address in memory. When the end of memory is reached, each cell is read and changed back to zero in reverse order. The test is then repeated using complemented data. A finite state machine is designed for March test algorithm. This is the BIST controller block that is testing the Circuit Under Test (CUT). A typical march algorithm (such
VI. MIRROR ADDRESS SEQUENCING

Major research work with respect to testing is related with combinational and sequential circuits. Many low power test techniques have been proposed for testing of digital circuits. Very little work is going for memory testing. Power consumption in test mode is influenced by the voltage levels, physical capacitance as well as the switching activity. Switching activity is the main cause for dynamic power. In test mode it is very high because the switching activity is more in test mode. In order to reduce power consumption in memory testing the switching activity must be reduced. In March like algorithms there is a walking in all the cells in one direction the process is repeated in opposite direction. This sequencing is just like marching from top to bottom and from bottom to top. In almost all the march algorithms address sequencing is in binary increment/decrement fashion.

In single bit change address sequences adjacent sequence is differed by only one bit. Suppose for example in binary sequence if there is a change between address seven and address eight, all the bits in the sequence must be changed. The switching activity is maximum in this case, while switching from 0111 to 1000. This can be eliminated in single bit change sequence. In single bit change sequence while address changing from seven to eight or eight to seven there is only one bit change. i.e. 0111 to 1000.

VI. HDL IMPLEMENTATION

If the memory under test is of 2^m X n then each address in the memory is verified multiple number of times. A typical example of March Y algorithm is shown here. March Y algorithm consists of four testing blocks. March Y algorithm is represented in a notation ↓ (W0); ↑(R0,W1,R1); ↓(R1,W0,R0); ↓ (R0).

In this March Y algorithm the notation ↓ represents the address sequencing either way. Once current address is checked then the next address to be checked is generated either incrementor or decrementor logic. Either way the address sequencing is allowed. The notation ↑ represents the address sequencing is upward direction. Incrementor is used to generate next address. 0 address to n-1 address all the locations are checked. The notation ↓ represents the address sequencing is downward direction. Decrementor is used to generate next address. n-1 address to 0 address all the locations are checked. The notation W0 represents a write operation at an address specified by address decoder with the input data all 0’s. R1 represents a read operation from an address specified by address decoder and the retrieved information is checked.

Table I: Binary and Single bit change sequences for four bit address

<table>
<thead>
<tr>
<th>Decima</th>
<th>Binary Sequence</th>
<th>Single Bit Change Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000</td>
<td>0000</td>
</tr>
<tr>
<td>1</td>
<td>0001</td>
<td>0001</td>
</tr>
<tr>
<td>2</td>
<td>0010</td>
<td>0011</td>
</tr>
<tr>
<td>3</td>
<td>0011</td>
<td>0010</td>
</tr>
<tr>
<td>4</td>
<td>0100</td>
<td>0110</td>
</tr>
<tr>
<td>5</td>
<td>0101</td>
<td>0111</td>
</tr>
<tr>
<td>6</td>
<td>0110</td>
<td>0101</td>
</tr>
<tr>
<td>7</td>
<td>0111</td>
<td>0100</td>
</tr>
<tr>
<td>8</td>
<td>1000</td>
<td>1100</td>
</tr>
<tr>
<td>9</td>
<td>1001</td>
<td>1101</td>
</tr>
<tr>
<td>10</td>
<td>1010</td>
<td>1111</td>
</tr>
<tr>
<td>11</td>
<td>1011</td>
<td>1110</td>
</tr>
<tr>
<td>12</td>
<td>1100</td>
<td>1010</td>
</tr>
<tr>
<td>13</td>
<td>1101</td>
<td>1011</td>
</tr>
<tr>
<td>14</td>
<td>1110</td>
<td>1001</td>
</tr>
<tr>
<td>15</td>
<td>1111</td>
<td>1000</td>
</tr>
</tbody>
</table>
with or it is all 1’s or not. In these March algorithms if the address decoder is of Binary address sequencer then the switching activity is more. In order to reduce the switching activity Single bit change address decoder is used. Fig 2 shows the hierarchical representation of the testable SRAM. BIST controller is the block implemented with March algorithm. BIST controller generates test patterns according to the March algorithm. Address decoder generates addresses where read and write operations should take place. SRAM here shows a Static Random Access Memory of size $2^n \times m$. This SRAM is tested by the BIST controller logic by using March algorithm. Address decoder here used is the single bit change address decoder.

![Fig 2: Hierarchical representation of SRAM BIST structure](image)

Fig 3 shows the flow graph for March Y algorithm that was implemented on finite state machine of the Built in Self-Test circuit. First of all it writes all 0’s from 0 address to n-1 address. After writing all the locations in the memory then start reading from zero address. Read at the address 0 and check whether the retrieved information is zero or not. If it matched then fill the same address with all 1’s. Again read from the same address and check whether the retrieved information is all 1’s or not. After that increment address. Again read for zeroes then write all 1’s and then read for 1’s. Repeat the same until address reaches maximum. Then Read for 1’s and write 0’s and read for 0’s from top to bottom. When the address reaches zero then read for zeroes either from top to bottom or bottom to top. These ways check all the locations for proper read and write operations. In all the read operations if the retrieved information is correct then it is fault free chip. If there is any mismatch is observed at any instance then the system stops checking further and asserts error signal and finish signal.

### VII. SIMULATION RESULTS

Testable Static Random Access Memory is described using Verilog HDL. And the description is simulated using Xilinx ISE simulator. Fig 4 shows the simulation results when the system working in normal SRAM mode. In this mode read and write operations at different addresses shown in Fig 4.

![Fig 4: Simulation results showing normal working mode](image)

When the test control signal is asserted then the memory test process starts. Built in self-test controller checks each location in memory. Fig 5 shows the test process while in ‘W0’ operation. When the algorithm is at W0 it writes all 0’s in the memory from top to bottom or bottom to top.

![Fig 5: Simulation results showing test mode during ‘W0’ operation](image)

Finite State Machine handling March algorithm checks all the locations in the memory multiple number of times for different faults that occur in memories. Fig 6 shows the simulation results when the BIST controller is in test mode. Here the signal ‘error’ is asserted, indicating the memory is faulty memory. Fig 7 shows the simulation results of the circuit in test mode. Here signal ‘error’ zero indicates the circuit under test is fault free and signal finish indicates that the test is completed.

![Fig 6: Simulation results showing the circuit under test is faulty](image)
In order to reduce power consumption during testing instead of binary address sequencing single bit change address sequencing is used. Fig 8 shows the simulation results when single bit change address sequencing is used in March algorithm. Because of the single bit change address sequencing switching activity also reduced.

**CONCLUSION**

Built in self-test structure implemented using March algorithm can perform well for all known faults that occur in memories. The switching activity is reduced with the help of single bit change address sequencer. Without affecting the fault coverage test power can be reduced enormously. An exponential increase of power reduction can be observed with a linear increase in the memory size. This great reduction in power is because of the switching activity while marching from top to bottom and bottom to top in the memory. Complete system is simulated and is implemented on Spartan 3E FPGA.

**REFERENCES**


---

***