International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST - 2015)

Window based Input Vector Monitoring Concurrent BIST using SRAM Cells with Diagnostic Data Compression

Kavya R.\textsuperscript{a},*, Dr. Santhosh Kumar S.\textsuperscript{b}
\textsuperscript{a}PG Scholar, Department Of ECE, Government Engineering College Idukki, Idukki, Kerala 685606, India
\textsuperscript{b}Associate Professor, Department Of ECE, Government Engineering College Idukki, Idukki, Kerala 685606, India

Abstract

Window based Input vector monitoring concurrent built-in self test (BIST) schemes perform testing during the normal operation of the circuit without the requirement to set the circuit offline to perform the test. It is evaluated based on the hardware overhead and the concurrent test latency (CTL). This thesis makes an attempt using novel input vector monitoring concurrent BIST scheme, which is based on the idea of monitoring a set of vectors called windows, and the use of a static-RAM like structure. To detect Static Faults in Random Access Memories, proposed a BIST architecture with the capability of hamming syndrome compression. To reduce the diagnostic data volume, a new idea of March element based (MEB) compression is proposed in this BIST Architecture. The data to be diagnosed in a RAM tested with a March test can be efficiently compressed by the MEB compression scheme. The scheme with SRAM cells is shown to perform the detection of error, error location and the error bit position in that location successfully. The software simulation is carried out in ModelSim SE PLUS 6.2b and Xilinx 14.2i design suite. FPGA implementation is done in Xilinx Spartan 3E kit and the output is verified.

Keywords: BIST (Built-in self-test); March test; RAM (Random Access Memory); CTL (concurrent test latency).

1. Introduction

Built-in self test (BIST) techniques reduces the dependency on expensive external testing equipment which includes a set of schemes that has the capability to carry out at-speed testing high fault coverage. Hence, they provide an attractive solution to the VLSI circuits testing problem\cite{1}. BIST techniques are of two types, namely, offline and online. Test mode and normal mode are the two offline architectures operating modes, during normal mode the BIST circuitry is idle. During test mode, a test generator module generates the inputs and are applied to the inputs of the CUT (circuit under test) and the RV (response verifier) captures the responses. So the normal operation of the CUT is stalled to perform the test, and due to this the performance of the system is degraded in which the circuit is included.

\*Kavya R. Tel.: +91-828-995-1007.
E-mail address: kavya.achu20@gmail.com
Various Input vector monitoring concurrent BIST techniques [2] – [7] have been proposed to eliminate this degradation in performance. By exploiting the CUT input vectors, these architecture performs the test of the CUT and its normal operation concurrently. The RV is enabled to capture the CUT response, if the incoming vector belongs to a set called active test set. The block diagram of an input vector monitoring concurrent BIST architecture is shown in Fig. 1.

The test and diagnosis of embedded RAMs(random access memories) are performed widely with the help of Built-in self-test (BIST) technique. In BIST the exportation of diagnostic data is time consuming because here the diagnostic data exportation is serial and hence exported bit-by-bit. Various techniques for compression of diagnostic data have been proposed in [8]–[12] to reduce the exportation time of data to be diagnosed. In [2] and [6], at-speed compression schemes for diagnostic data are presented. But diagnostic data in those compression techniques are compressed in a lossy form. The CTL(concurrent test latency) of an input vector monitoring scheme is the mean time required to complete the test, counted either in time units or number of clock cycles, while the CUT operates in normal mode.

A March element in an applied March test requires multiple Read operations to detect faults[10]. If a March test is performed by a BIST circuit for a RAM under test, the data to be diagnosed for a static fault is exported by the PAE method multiple times. As a result, exportation of many redundant diagnostic data occurs. Here, therefore, a March-element-based (MEB) compression scheme is introduced, where the data to be diagnosed is compressed in a PAE way only after the complete operations of a March element are carried out or finished to detect static fault(SF) for a word in a particular address. As a result, minimization of the diagnostic data which is redundant for a fault, detect SFs, error location and error bit position in that particular location.

This brief is organized as follows: In Section 2, the existing scheme is discussed and in Section 3, introduces the proposed scheme. Section 4, discusses Diagnostic Data Compression and in Section 5, the simulation results. Finally, Section 7 wind up with the conclusion of this brief.

2. Existing Scheme

Consider a combinational CUT, shown in Fig. 2 with n bit input lines; hence this DUT has a set of possible input vectors of $2^n$. The idea behind the proposed scheme is monitoring a window of vectors, with size $W$, where $W = 2^w$, in which $w$ is an integer number $w < n$. At each point, this scheme monitors the test vectors belonging to the window, and if a hit is performed by a vector, the RV is enabled. Examination of the next window will start only after all the vectors that belong to the current window under examination have reached the CUT inputs[12].

During the normal mode, $T/N = 0$, the inputs to the CUT are the normal input vectors. The CBU is also driven with the inputs of the CUT as follows: $k$-stage comparator inputs are driven with the high order $k$ bits; the other inputs of the comparator are driven by the $k$ bit MSB outputs of the decoder. A modified decoder shown as $m_{dec}$ in Fig. 2 is used in the scheme proposed and a SRAM(static RAM) like cell based logic module.
The design of the particular scheme is for the case of \( n = 8, w = 3, \) and \( k = 2. \) The logic module consists of \( W_s \) Logic Cells. Corresponding to a vector that belongs to the active window, there is a LogicCell[i], \( 1 \leq i \leq W_s, \) and that can be either empty or full.

Depending upon the number of operations carried out in a operation sequence for sensitization, Functional faults of RAMs is divided into SFs(Static Faults) and DFs. SFs requires maximum of one sensitizing operation. DFs needs more than one sensitizing operation. Functional faults of RAMs are widely detected with the help of March Tests. A March test consists of multiple March elements. An example of March C test is \( \uparrow\downarrow(w_0); \uparrow(r_0,w_1); \uparrow(r_1,w_0); \downarrow(r_0,w_1); \downarrow(r_1,w_0); \uparrow(r_0) \) and is used to detect SFs in a RAM, where \( w_d \) is a Write operation with input vector \( d \) and \( r_d \) shows a Read operation with expected data \( d. \) This paper focus on Static Fault detection. As a future scope of this paper, DFs can be detected. Detection of DFs comprises of March element of multiple Read operations. Example of March RAW test is \( \uparrow\downarrow(w_0); \uparrow(r_0,w_0, r_0, w_1, r_1); \uparrow(r_1,w_1, r_1, r_1, w_0, r_0); \downarrow(r_0,w_0, r_0, r_0, w_1, r_1); \downarrow(r_1,w_1, r_1, r_1, w_0, r_0); \uparrow(r_0) \) and is used to detect both SFs and DFs [15].

### 3. Proposed Architecture

The design of proposed architecture and the \( m\_dec \) module with \( w = 3 \) is shown in Fig. 3(a), Fig. 3(b) respectively and operates as follows. All outputs of the decoder are equal to one when test generator enable (tge) is enabled. All outputs are disabled when comparator (cmp) is disabled and tge is not enabled. The architecture acts as a basic decoding structure when tge is disabled and cmp is enabled. Here the cmp signal is generated by the comparator by comparing the k bit MSB of the decoder output and the k bit MSB of the mux output. The tge signal is generated within the logic circuitry by ORing the delayed rve signal and the reset signal.

Using this modified decoder, the input to the SRAM cell logic is derived. The written value in the addressed SRAM cell is read and this value is compared within a comparator with another modified decoder driven with the same input value in order to verify that the written and the read values are same. If error is present then the following scheme as shown in Fig. 4(c) is used to detect the error location and the particular bit error position in the data in addressed location.

The data to be diagnosed, comprising the session number, faulty address, and Hamming syndrome (HS) are exported by a BIST circuit to ATE(Automatic Test Equipment) to support the memory diagnosis of a detected fault. The session number shows in which operation within the march test detects the fault. The HS is obtained by computing bitwise XOR operation of the expected data and the read data[5]. Hence, the value of HS can show the faulty bit in a faulty word. In BIST the testing and exportation of diagnostic data is carried out in a PAE way. If a fault is detected by BIST, the test application is stalled and the data to be diagnosed are exported. Also, if the March element within
applied March test composed of more than one Read operation, the BIST with PAE way for each Read operation to export diagnostic data may result in exportation of redundant diagnostic data.

4. Diagnostic Data Compression

The diagnostic data is exported by a BIST circuit comprises of the faulty address, session number, and Hamming syndrome (HS), for the memory diagnosis of a detected fault. The bitwise XOR operation on the expected data and the read data gives the HS[9]. The test operation which detects the fault within a March element is denoted as Session Number. The faulty bit of a faulty word is indicated by the information provided by HS. Typically, the exportation of diagnostic data and the testing of a BIST is performed in a PAE way.

The MEB compression scheme proposed here exports one faulty address independent of March element Read operations. The faulty addresses can be compressed by the BIST design with MEB method for the diagnostic data to be exported. Here, only the HS is compressed in the proposed MEB compressor using PAE efficiently. Hence, only the HS compression function is included.

Thus if there is any error in the result, first the faulty location is detected for the addressed word. Then the proposed method extracts the HS and outputs the particular error bit position in that Hamming Syndrome for the particular faulty address location.

5. Results and Discussion

The modeling and simulation is done in ModelSim SE PLUS 6.2b and the design and implementation is done in Xilinx ISE 14.2i.

The output waveforms of the conventional and modified window based concurrent BIST with SRAM cells is shown below. Fig. 4. shows the output waveform of the input vector monitoring concurrent BIST using SRAM cells. The decoder input shows the relative location of the incoming vector in the current window and the output gives the value to be written. Fig. 5. shows the output waveform of the input vector monitoring concurrent BIST using SRAM cells with diagnosis of data in a March Element Based error detection. As shown in the waveform, on the basis of march elements in a march test for static faults, the error is detected and the corresponding address location is obtained. Then the each error bit position in that particular location is also detected. Here the diagnostic data compression is based on the hamming syndrome. The pause and export compression scheme is used.
The main aim of this paper is to find out the error, error or the faulty location and the error bit position in the diagnostic data in that faulty location. Apart from this the main consideration in CBIST unit is the area and delay. While comparing the area and the delay of the proposed scheme with the conventional CBIST scheme using SRAM
cells, the area as well as the delay is reduced to a large extend. This is shown in the following Table 1. Here the conventional BIST scheme referred is one with ALU as Circuit Under Test and the output for a particular address is read using RAM and is compared with the values stored in the ROM with the same address input.

Table 1. Comparison of Area and Delay.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>No.of Slices</th>
<th>Minimum period(nS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional BIST</td>
<td>16</td>
<td>3.492</td>
</tr>
<tr>
<td>Conventional BIST with SRAM cells</td>
<td>32</td>
<td>14.982</td>
</tr>
<tr>
<td>Conventional scheme with proposed data compression</td>
<td>41</td>
<td>14.492</td>
</tr>
<tr>
<td>Proposed scheme with error detection</td>
<td>11</td>
<td>3.150</td>
</tr>
<tr>
<td>Proposed scheme with data compression</td>
<td>8</td>
<td>3.139</td>
</tr>
</tbody>
</table>

Table 2. Comparison of Delay and Area in conventional and proposed scheme.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Minimum Period (nS)</th>
<th>Minimum input arrival time(nS)</th>
<th>Maximum output required time(nS)</th>
<th>No.of Slices</th>
<th>No.of FFs</th>
<th>No.of 4-input LUTs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional BIST with SRAM cells</td>
<td>14.982</td>
<td>8.801</td>
<td>4.285</td>
<td>32</td>
<td>14</td>
<td>61</td>
</tr>
<tr>
<td>Proposed scheme with data compression</td>
<td>3.139</td>
<td>4.155</td>
<td>4.252</td>
<td>8</td>
<td>6</td>
<td>16</td>
</tr>
</tbody>
</table>

The reduction in minimum period is 11.843 nS. Comparing the code length this is a considerable reduction. Number of slices is reduced in terms of 24 slices. The main delay and area comparison between the existing and the proposed scheme is performed by comparing the terms as shown in Table 2 and the graphical representation is shown in Fig. 6. Analysis shows that there is 75% reduction in area and 81.72% reduction in delay.

The FPGA implementation is done in FPGA device Xilinx Spartan3E(Family) XC3S100E(Device) and CP132 (Package).

6. Conclusion

Input vector monitoring concurrent BIST scheme using SRAM cells is one of the efficient input vector monitoring concurrent BIST with respect to the hardware overhead and CTL tradeoff available today, which is based on the idea of monitoring a window of vectors and the use of a static-RAM. To detect Static Faults in RAM, a March element of an applied March test consists of a single sensitizing operation. If a BIST circuit performs such type of March test for a RAM under test to detect DFs, then the pause-and-export method exports the diagnostic data of a static fault multiple times. This causes that many redundant diagnostic data are exported. Thus this thesis proposes a March-element-based (MEB) error detection and compression technique which compresses the diagnostic data in a PAE form.
when all the operations of a March element are completed for an addressed word. Consequently, the error location and the particular error bit position in that location is detected successfully. The FPGA implementation shows that the output obtained is as expected from the software simulation results. The Future scope is in the field of Dynamic Fault detection. This Scheme can be modified to detect the Dynamic Fault in the proposed system and the large volume of diagnostic data exportation during multiple read operation to detect DFs can be reduced to a large extend as in this case of Static Faults. For this purpose March RAW test for DFs will be used in place of March C test for SFs.

References