Implementation of Testing Methods for VLSI Circuits

A. Ooha\(^1\) and V. Leela Rani\(^2\)

\(^1\)M.Tech. Student, Department of Electronics and Communication Engineering, G.V.P. College of Engineering (A), Visakhapatnam, AP-India, Email: a.ooha9@gmail.com

\(^2\)Associate professor, Department of Electronics and Communication Engineering, G.V.P college of Engineering (A), Visakhapatnam, AP-India, Email: lee_rani@gvpce.ac.in

ABSTRACT:

An ASIC chip designed may not meet the functionality requirements. There may be lot of conditions which may cause damage to the designed circuit and effect its functionality. Those conditions include processing faults (such as missing contact windows, oxide breakdown), material defects (such as cracks, crystal imperfections), packaging failures etc. These problems may cause faults in the digital logic circuits and effects circuit functionality. Therefore, the circuit must be tested in order to know whether it is working properly or not.

Several testing methods and algorithms have been proposed such as D-algorithm, PODEM algorithm, Built-in-self-test etc in literature. In this paper, Built-in-logic-block-observer (BILBO) method of testing is implemented. A test pattern compaction method, STAR-EDT is combined with BILBO to derive minimum number of test data for detecting all faults that are injected into the circuit. The combined method called BILBO with STAR-EDT methodology is proposed. Existing and proposed methods are implemented and applied to various test circuits. After that, a Scheduling method is applied to BILBO with STAR-EDT method for all test circuits at a time. The proposed method requires less number of test patterns than the number of patterns required in existing BILBO methodology. Scheduling concept applied to proposed method results in reduced timing requirements for test procedures.

KEYWORDS: BILBO, STAR-EDT, Scheduling, Test data, Faults.

*Corresponding author

Ms. A. Ooha

M.Tech. student,
G.V.P. College of Engineering (A)
Visakhapatnam, AP, India.
Email: a.ooha9@gmail.com.
INTRODUCTION:

An ASIC chip designed may not meet the functionality requirements. Several testing methods and algorithms have been proposed such as D-algorithm, PODEM algorithm, Built-in-self-test etc to test the designs in order to define whether it is faulty or fault-free. The D-algorithm and PODEM (Path-Oriented-Decision-Making) algorithm are external methods of test data application. Whereas, BIST (Built-In-Self-Test) architecture is designed on-chip and the entire testing process is done on chip. BIST is categorized into several types namely Logic BIST (LBIST), Memory BIST (MBIST), Scan-based BIST, Programmable BIST, BILBO etc\textsuperscript{3,4,5,8}. In this paper, BILBO (Built-In-Logic-Block-Observer) method of testing is considered. BILBO needs more number of patterns to detect faults. A test data compression methodology STAR-EDT is used which derives minimum number of test patterns to detect more number of faults\textsuperscript{1,6}. Testing multiple test circuits requires more time. Scheduling methodology is applied to test reduce the time consumption and hence, test circuits can be tested in a parallel way.\textsuperscript{2}

BILBO METHODOLOGY:

BILBO (Built-In-Logic-Block-Observer) is one of the methodologies of BIST. It uses a register that operates in different modes based on its control inputs. The structure of BILBO register consists of flip-flops and a combinational logic comprising of XOR, NAND and NOR gates between every two flip-flops. Two control inputs are present in order to decide the mode of operation. The structure of BILBO register is shown in Figure 1.

![Figure 1: Architecture of BILBO register](image-url)
MODES OF BILBO REGISTER:

The inputs B1 and B2 are control inputs to BILBO register which decides the modes of operation of BILBO register.

- If B1=B2=0, then BILBO register will be in RESET mode.
- If B1=0, B2=1, then BILBO register will be in Scan mode.
- If B1=1, B2=0, then BILBO register will be in PRPG or MISR mode.
- If B1=B2=1, then BILBO register will be in Register mode.

The BILBO register in PRPG mode generates test patterns in order to give them as inputs to the test circuit. The response of the test circuit is given to BILBO register in MISR mode as input to generate a signature. This signature of test circuit is compared with reference signature. If both signatures match with each other, then the circuit is said to be fault-free.

FAULTS INJECTED INTO TEST CIRCUITS:

The test circuits considered in this paper are 1-bit full-adder, 32-bit ALU, 32-bit ripple carry adder and 32-bit magnitude comparator. In 1-bit full-adder, 2 faults (stuck-at-0 and stuck-at-1) are injected. In 32-bit ALU, 10 faults (8 s-a-1 and 2 s-a-0) are injected and in 32-bit magnitude comparator, 7 faults (4 s-a-1 and 3 s-a-0) are injected. In 32-bit ripple carry adder, 5 faults (2 s-a-1 and 3 s-a-0) are injected during test process.

PROPOSED METHOD:

BILBO with STAR-EDT methodology is proposed in this paper. STAR-EDT is a test data compression scheme which derives minimum number of test data that is able to detect all faults in the test circuit. STAR-EDT consists of a ring generator that produces test patterns that are applied to the test circuit. As BILBO register has an advantage of high fault coverage, ring generator is replaced with BILBO register. The responses of the test circuit helps to recognize parent pattern (a pattern that is able to detect at least one fault in the fault-list). By applying that parent pattern to a phase shifter, children patterns can be derived, which together forms a test cluster. The derived test clusters are again fault simulated in order to get minimum number of test clusters. The flow of proposed method implemented is shown in Figure 2.
SCHEDULING METHODOLOGY:

Scheduling is applied to above test procedure in order to test four test circuits at a time. The scheduling method considered in this paper consists of three sessions. In first session, ripple carry adder and 1-bit full adder are tested. In second session, magnitude comparator and in third session, 32-bit ALU are tested respectively. The structure of scheduling methodology is shown in Figure 3.

RESULTS AND DISCUSSIONS:

The simulation results of above methodologies implemented on all test circuits are shown below. The proposed method derived less number of test patterns for all test circuits compared with number of patterns required in BILBO methodology.

BILBO METHODOLOGY:

Figure 4 presents simulation result of BILBO methodology for 1-bit full adder. In the simulation, the signals “sig_faulty” and “sig_nofault” represents the signatures of both faulty and
fault-free test-circuits respectively. The signal “fault” compares both signatures and it is enabled if both signatures differ with each other indicating that the circuit is faulty.

![Simulation result of BILBO methodology for 1-bit full adder](image)

Figure 4: Simulation result of BILBO methodology for 1-bit full adder

```
Console

ISim P-40xd (signature 0x7708f090)
This is a Full version of ISim.

# run 1000 ns
Simulator is doing circuit initialization process.
Finished circuit initialization process.
circuit is faulty!
circuit is faulty!
ISim>
```

Figure 5: Console window of BILBO methodology for 1-bit full adder

Figure 6 to Figure 11 shows the simulation results of BILBO methodology for test circuits 32-bit ALU, 32-bit Ripple carry adder and 32-bit Magnitude comparator respectively.
Figure 6: Simulation result of BILBO methodology for 32-bit ALU

Figure 7: Console window of BILBO methodology for 32-bit ALU
Figure 8: Simulation result of BILBO methodology for 32-bit Ripple carry adder

```
Console
ISim P-40xd (signature 0x7708f090)
This is a Full version of ISim.

# run 1000 ns
Simulator is doing circuit initialization process.
Finished circuit initialization process.
circuit is faulty!
circuit is faulty!
ISim>
```

Figure 9: Console window of BILBO methodology for 32-bit Ripple carry adder
Figure 10: Simulation result of BILBO methodology for 32-bit magnitude comparator

PROPOSED METHOD:

Figure 12 to 19 presents the simulation results of all test circuits when BILBO with STAR-EDT methodology is applied. In the simulation result, signals ‘I1’, ‘I2’, ‘I3’ and ‘I4’ represents the derived cluster of patterns. The signal ‘numberoffaultsdetected’ indicates the number of faults detected by that particular test cluster.
Figure 12: Simulation result of proposed method for 1-bit full adder

```plaintext
80 The number of faults detected by this cluster is: 1
96 The number of faults detected by this cluster is: 2
104 The number of faults detected by this cluster is: 1
112 The number of faults detected by this cluster is: 0
120 The number of faults detected by this cluster is: 2
136 The number of faults detected by this cluster is: 0
144 The number of faults detected by this cluster is: 2
152 The number of faults detected by this cluster is: 1
160 The number of faults detected by this cluster is: 0
168 The number of faults detected by this cluster is: 2
184 The number of faults detected by this cluster is: 1
```

Figure 13: Console window of proposed method for 1-bit full adder
Figure 14: Simulation result of proposed method for 32-bit ALU

![Simulation result of proposed method for 32-bit ALU](image1.png)

Figure 15: Console window of proposed method for 32-bit ALU

![Console window of proposed method for 32-bit ALU](image2.png)
Figure 16: Simulation result of proposed method for 32-bit Ripple carry adder

Figure 17: Console window of proposed method for 32-bit Ripple carry adder
SCHEDULING METHODOLOGY:

The test procedure considered in this paper is scheduled to test different test circuits at a time. The simulation results of scheduling methodology are presented from Figure 20 to Figure 23. The simulation result of all sessions is shown in Figure 20. Figure 21 to Figure 23 represents the testing process in session-1, session-2 and session-3 respectively. In the first session, 1-bit full-adder and 32-bit ripple carry adder are tested. In session-2, 32-bit magnitude comparator and in session-3, 32-
bit ALU are tested respectively.

Figure 20: Simulation result of Scheduling methodology for all sessions

Figure 21: Simulation result of Scheduling methodology for session-1
Table 1 presents the comparison of existing methodology BILBO and proposed method called BILBO with STAR-EDT. BILBO requires more number of test patterns to detect all faults that
are injected into the test-circuits. Whereas, BILBO with STAR-EDT derived less number of test patterns compared to BILBO methodology to detect all faults that are injected into test-circuits.

### Table 1: Comparison between BILBO and proposed methodologies

<table>
<thead>
<tr>
<th>TEST CIRCUIT</th>
<th>NUMBER OF FAULTS INJECTED</th>
<th>NUMBER OF PATTERNS REQUIRED IN BILBO METHODOLOGY</th>
<th>NUMBER OF TEST PATTERNS DERIVED BY BILBO WITH STAR-EDT METHODOLOGY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-BIT FULL ADDER</td>
<td>2</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>32-BIT ALU</td>
<td>10</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>32-BIT MAGNITUDE COMPARATOR</td>
<td>7</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>32-BIT RIPPLE CARRY ADDER</td>
<td>3</td>
<td>9</td>
<td>1</td>
</tr>
</tbody>
</table>

The proposed method derived minimum number of test pattern required to detect all faults that are injected into the test-circuits. The number of compressed test patterns are given in Table 1.

### SCHEDULING METHODOLOGY:

Table 2: Scheduling applied to proposed methodology

<table>
<thead>
<tr>
<th>TEST-CIRCUITS</th>
<th>SESSIONS</th>
<th>TIME REQUIRED TO TEST ALL TEST-CIRCUITS IN SCHEDULING</th>
</tr>
</thead>
<tbody>
<tr>
<td>32-BIT RIPPLE CARRY ADDER</td>
<td>SESSION-1</td>
<td></td>
</tr>
<tr>
<td>1-BIT FULL-ADDER</td>
<td>SESSION-1</td>
<td></td>
</tr>
<tr>
<td>32-BIT MAGNITUDE COMPARATOR</td>
<td>SESSION-2</td>
<td>2.213 ns</td>
</tr>
<tr>
<td>32-BIT ALU</td>
<td>SESSION-3</td>
<td></td>
</tr>
</tbody>
</table>

The scheduling methodology reduces the time required to test all test-circuits individually. Total time required to test all circuits is 2.213 ns only with three sessions. The process of scheduling and its sessions are given in Table 2.

### CONCLUSION:

In this paper, BILBO methodology applied to four test-circuits is presented. In order to increase the fault coverage and to compress test data, STAR-EDT methodology is applied to BILBO method. Later, Scheduling methodology is applied in order to reduce the time required to test the circuits independently. Existed and proposed methodologies are verified on various test-circuits. The simulation results concludes that the proposed BILBO with STAR-EDT methodology requires minimum number of test patterns to detect all faults that are injected into the test-circuits and
scheduling methodology applied to BILBO with STAR-EDT reduces the time consumption by testing them parallely.

REFERENCES:

4. Dong Xiang, Xiaqing Wen and Laung-Terng Wang, “Low-Power Scan-Based Built-In Self-Test Based on Weighted Pseudorandom Test Pattern Generation and Reseeding”, IEEE transactions on Very Large Scale Integration (VLSI) systems, March 2017; 25: 3.