International Conference on Information and Communication Technologies (ICICT 2014)

Design and Implementation of an On-Chip Test Generation Scheme Based on Reconfigurable Run-Time Programmable and Multiple Twisted-Ring Counters

Aida S Tharakan\textsuperscript{a}\textsuperscript{*}, Binu K Mathew\textsuperscript{b}

\textsuperscript{a,b}Dept. of Electronics and Communication, Saintgits College of Engineering, Kottayam-686532, India

Abstract

Built-in-self-test (BIST) has emerged as a very effective solution to VLSI testing problems. Related work based on single fixed-order twisted-ring-counter design requires longer testing time to achieve high fault coverage and large storage space to store the seeds and the control data. By using multiple programmable twisted-ring-counters (PTRC), a considerable reduction in test application cycles were achieved. In this paper, an on-chip test generation scheme based on reconfigurable run-time programmable multiple twisted-ring-counters is proposed to generate more number of different test patterns based on the requirements. The design was modeled in VHDL and simulated and synthesized using Xilinx ISE 14.2.

Keywords: Fault Coverage; Logic BIST; Reconfiguration; Twisted-ring-counters

1. Introduction

Built-in-self-test (BIST) is becoming an attractive solution in Integrated Circuit (IC) testing. For design and test development, BIST considerably reduces the costs of automatic test-pattern generation (ATPG), and reduces the product introduction delays as a fully-designed system cannot be tested. Without BIST, maintenance test requires

\textsuperscript{*} Aida S Tharakan. Tel.: +91 474 2454233.
aidastharakan@gmail.com
the presence of expensive automatic test equipment (ATE) at the site of the failing system, and this is the major cost. 
With BIST, there is no need for an ATE, so this reduces testing cost of the system. For boards and systems, BIST 
significantly reduces the diagnosis and repair cost, by rapidly determining and indicating which sub-assembly or 
component is faulty, without the extensive labor and equipment normally required. This great reduction in diagnosis 
and repair time naturally leads to a major shortening in the service interruption, particularly at the system level. 

Earlier BIST methods can be classified into test-per-scan\(^2,4\) and test-per-clock\(^5,14\) according to their application 
schemes. The test-per-scan method serially loads one test pattern into the scan chains bit-by-bit. Therefore large test 
application time on pattern loading is necessary, especially when the scan chain length is long. The test-per-clock 
method applies one test pattern for each test cycle and all output responses corresponding to one test pattern are 
captured and loaded to a response monitor simultaneously. 

Though the area overhead required for the response monitor may be large, the test-per-clock do have the benefit 
of much shorter test time as one test pattern can be applied for each clock cycle. Therefore in the applications where 
the area overhead is not the major concern while test time is, test-per-clock can become a good choice for testing. In 
this paper, it focuses on the test-per-clock BIST scheme.

Most of the BIST techniques use pseudorandom patterns, which can be generated on-chip by means of linear 
feedback shift registers (LFSR). However, problems with pseudorandom testing include long test times and the need 
for large LFSRs to avoid linear dependencies in the generated test sequence, so that it is difficult to provide 
satisfactory fault coverage, especially for circuits containing hard-to-detect (random pattern resistant) faults.

In this paper, twisted–ring counters are used for test generation. A reconfigurable, programmable, multiple 
twisted-ring-counter based on-chip test generation is proposed. All input scan cells together with the primary inputs 
are configured into multiple equal-length scan segments, each of which is converted to a programmable TRC 
(PTRC) design. A mode switching logic unit and some distributed reversion logic units are developed which control 
various TRC operations to produce the required test patterns based on the seeds. By providing the suitable control 
data, the state of each PTRC design can be individually programmed so that the probability to generate required test 
paterns can be greatly enhanced. Here the PTRC can be reconfigured by adding some additional features. The 
design was modeled in VHDL. It is simulated and synthesized using Xilinx ISE 14.2. The design is implemented on 
various FPGA devices to obtain improved device utilization.

The rest of the paper is organized as follows. Section 2 gives a brief description of related works. Section 3 
describes the programmable multiple twisted ring counter on-chip test generation scheme and the proposed 
reconfigurable programmable multiple twisted ring counter. Section 4 shows the simulation and synthesis results. 
Finally the concluding remark and future work is presented in section 5.

2. Related Works

A number of pattern-generation methods have been proposed for both test-per-scan\(^2,4\) and test-per-clock\(^5,14\) 
BIST. Most of these methods use pseudorandom patterns, which can be generated on-chip using LFSRs. The 
problem associated with pseudo random testing includes longer testing times, so it is difficult to achieve sufficient 
fault coverage.

To avoid the problem of achieving high fault coverage with reasonable application times, several methods have 
been proposed. The most common approach is mixed mode BIST\(^2,7\), where deterministic patterns that target hard-to-
detect faults specifically are added to one or more pseudorandom sequences. The deterministic patterns are produced 
via LFSR reseeding\(^2,3\) or additional specific logic\(^4,7\). For the test process, a complicated control is necessary to 
switch between different modes of testing and to load required seeds and to reconfigure the logic. These methods 
also require longer test application time, large storage data volume and additional logic on the paths which is 
critical.

An alternative test-per-clock method based on a single twisted ring counter and reseeding logic is proposed\(^9,13\). 
The main benefits of these methods are simple test control, no need of adding additional logic in the critical paths of 
Circuit under Test (CUT) and also the hardware is independent of the circuit under test structure.

A built-in pattern generation based on reseeding of twisted-ring-counter was proposed\(^9,10\). To load seed patterns 
into the input scan registers and to perform TRC operations, a control unit is used so as to generate more patterns. 
The seeds can be stored in on-chip ROM and also described a scheme for seed selection for the TRC from the
patterns of precomputed test set.

Then the control unit is modified to reduce the number of required seeds such that more deterministic patterns in the test set can be embedded into the seed\(^{11,12}\). The benefit of this method is reduced tester memory. The seeds can be viewed as an encoded version of the test set which is stored in the tester memory. Here the number of seeds is reduced, but the length of the test sequence for each seed will be longer. Thus a test sequence reduction technique is proposed\(^{13}\), which enables the control unit to skip those redundant patterns for each seed. A single TRC is used in all of these methods and the patterns that are generated from each seed are applied in a fixed order. Long test application time is still required to attain complete fault coverage.

Also a novel programmable, multiple-TRC-based on-chip test generation scheme is proposed\(^{14}\). In this all input scan cells together with the primary inputs are configured into multiple equal-length scan segments, each of which is converted to a programmable TRC (PTRC) design. A mode switching logic unit and some reversion logic units are developed which controls various TRC operations. The state of each PTRC design can be individually programmed by appropriate control data so it can generate more number of test patterns. An efficient procedure to determine a minimal set of seeds and their corresponding control signals is also proposed, thus test sequence length and the storage test data volume is minimized.

3. Proposed Scheme

Twisted-ring-counters are used as built-in test pattern generators due to their simple control circuitry and small area overhead. A k-bit ring counter (RC) is a group of k flip-flops connected as a shift register, with a feedback from the output of last flip-flop to the input of the first flip-flop. A k-bit twisted-ring counter (TRC) is similar to a k-bit ring counter except that an inverter is inserted on the feedback path. With an initial state, a k-bit RC and a TRC generates ‘k’ and ‘2k’ different test patterns respectively.

Fig. 1 (a) and (b) show a k-bit ring counter and a k-bit twisted ring counter and (c) shows a combination of an RC and a TRC. It can switch between the RC and TRC modes by setting the control signal ‘ctrl’ and can generate ‘3k’ patterns.

3.1. Programmable Multiple Twisted-ring-counter based Test Generation

The Programmable Twisted-Ring Counter (PTRC) is based on the design shown in Fig.1(c), which can generate maximum number of test patterns, maximally ‘3k’ patterns by a k-bit PTRC. Also PTRC can be programmed to perform different orders of operations so that it can attain more pattern generation flexibility. It can also decrease the
test application time by concurrently generating the test patterns for all segments with different control signals.

Fig. 2 shows the programmable multiple TRC which consists of a mode switching logic unit and a set of PTRC units with a reversion logic unit associated with each PTRC unit. Fig. 2 contains two scan segments, scan segment 1 and 2. A PTRC unit consists of a 2-to-1 multiplexer and a 3-input XOR gate in front of the scan input of the scan segment. For each PTRC unit, one reversion logic unit with a one-bit control pin is employed for each PTRC unit. Here area overhead is small because of simple hardware units and also no performance degradation.

The overall test generation process is controlled by the mode switching logic unit and the reversion logic unit. With the concurrent control of these units, all PTRC designs can jointly produce the required test patterns at the same time. By appropriately controlling the control signal for each reversion logic units, the TRCs can be separately programmed. Therefore more effective patterns can be generated so it leads to significant reduction in the total storage data volume. The response monitor is used to capture the test responses. This method is a test-per-clock one.

The test generation modes in the PTRC design namely shift-in, rotate, and twist. The shift-in mode is to load the seed, i.e. initial patterns into the PTRCs. Each seed input can be stored in an on-chip ROM. The rotate mode rotates the seed by one bit per test cycle. The twist mode also performs rotate operation but flips the value of the last bit to be shifted to the first bit. The various test patterns can be generated based on the seed by appropriately switching between the twist and rotate modes.

In Fig. 2, in the mode switching logic unit, an m-bit binary counter is used to keep track of how many cycles have been executed in the PTRC mode, where \( m = \left\lfloor \log_2 k \right\rfloor \). When the count value equals ‘k’, the control signal ‘CE’ is activated to increment the 2-bit binary counter so as to activate a mode transition. The control signal ‘ctrl’ is used to control the reversion logic units. The control signal and the 2-bit counter in the mode switching logic unit decide the test modes. Fig. 3 (a) shows the control signals for various PTRC modes and (b) shows the test pattern generation using the multiple-PTRC-based BIST. Each mode will be executed for ‘k’ cycles before transferring to the next mode. There are two possible orders to generate 3k patterns for each seed depending on whether ctrl[1] is 0 or 1. The execution order is (shift-in → rotate → twist → twist) when ctrl[1] = 0 while the execution order when ctrl[1] = 1 is (shift-in → twist → twist → rotate). Now the twist mode is executed twice in both orders.

![Fig. 2. Multiple Programmable twisted ring counter](image-url)
Fig. 3. (a) Control signals for various PTRC modes. (b) Test pattern generation using the multiple-PTRC-based BIST with seed = (101001) and \( \text{ctrl} = (10) \)

Fig. 3(b) shows an example where two scan segments are used, each with three scan cells \((k = 3)\). Assume that the seed pattern is \((101001)\) and the control vector \(\text{ctrl} = (10)\). For scan segment 1, the seed is \((101)\) and the \(\text{ctrl}[0] = 1\), two twist modes are executed followed by the rotate mode. Thus nine different patterns are obtained from segment 1. Similarly for the scan segment 2, the rotate mode is executed before the two twist modes. In this, the test sequence length is \((3k+k)\), where ‘3k’ patterns generated in three test generation modes and ‘k’ cycles for the shift-in mode. As a result by using multiple numbers of segments, the test sequence length can be reduced.

3.2. Reconfigurable Multiple Programmable Twisted-ring-counters based Test Generation Scheme

Fig. 4. Reconfigurable Multiple Programmable twisted ring counters
In the proposed design, the existing PTRC can be reconfigured by adding some additional features such as a block select module and an ‘and’ gate. The ‘and’ gate is placed in between the scan register and the multiplexer. The PTRC unit can be replicated depending on the requirements. The outputs of the block select module and the multiplexer are given to the inputs of the ‘and’ gate. The outputs of the block select module vary depending upon the control signal, ‘Control’. The PTRC works only if the output of the decoder connected to it is active.

In the design in Fig. 4, since the output of the block select module is 8 bits, the PTRC can be replicated 8 times and each output of the block select module is connected to the corresponding PTRC. For example, when the Control = (011), the first four segments will work, thus it can generate different test patterns based on the seeds. In this scheme, seeds can be applied simultaneously. Since it is reconfigurable, it can activate any number of PTRC units based on the needs. This significantly reduces the pattern generation time.

3.3. Reconfigurable Run-Time Programmable and Multiple Twisted-Ring-Counters Based Test Generation Scheme

Fig. 5 shows the Reconfigurable Run-Time Programmable Multiple twisted-ring-counters based test architecture. Here it can activate any bit PTRC based on the requirements by changing the input of the block select module and the output will be the output of currently running PTRC by using the signal ‘Select’.

4. Result and Analysis

To evaluate the feasibility of the proposed scheme, it was modeled using VHDL. Then it is simulated and synthesized using Xilinx ISE 14.2 device.
The simulated waveform of the programmable multiple twisted-ring-counters with two segments is shown in Fig. 6 and the simulated waveform of the reconfigurable, programmable multiple twisted-ring-counters with eight segments is shown in Fig. 7. After obtaining the test patterns, find the fault coverage using an algorithm by applying it to the ISCAS benchmark circuits. Fig. 8 shows the simulated waveform of the reconfigurable run-time programmable multiple twisted-ring-counters.

Fig. 7. Simulated waveform of the reconfigurable, programmable multiple twisted-ring-counters with eight segments

Fig. 8. Simulated waveform of the reconfigurable, run-time programmable multiple twisted-ring-counters

Table 1 shows the device utilization summary of the proposed reconfigurable PTRC with eight segments for different FPGA devices. It is clear that the introduction of the proposed scheme do not result in considerable increase in the total area and delay. The proposed scheme utilizes less number of slices, LUTs and IOBs when it is implemented on Virtex 5. But, when it is implemented on Virtex 4, the device utilization is more. It is clear that the proposed scheme has less delay in Virtex 5.

Table 1: Device utilization summary of the proposed reconfigurable PTRC with eight segments
Table 1. Device utilization and delay for Reconfigurable PTRC and Reconfigurable Run-Time PTRC.

<table>
<thead>
<tr>
<th>Device</th>
<th>Architecture</th>
<th>No. of occupied Slices</th>
<th>No. of LUTs</th>
<th>No. of IOBs</th>
<th>Delay (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtex-4</td>
<td>Reconfigurable PTRC</td>
<td>51</td>
<td>56</td>
<td>111</td>
<td>5.028</td>
</tr>
<tr>
<td></td>
<td>Reconfigurable Run-Time PTRC</td>
<td>61</td>
<td>91</td>
<td>70</td>
<td>3.879</td>
</tr>
<tr>
<td>Virtex-5</td>
<td>Reconfigurable PTRC</td>
<td>34</td>
<td>34</td>
<td>28</td>
<td>2.825</td>
</tr>
<tr>
<td></td>
<td>Reconfigurable Run-Time PTRC</td>
<td>43</td>
<td>58</td>
<td>70</td>
<td>3.04</td>
</tr>
</tbody>
</table>

5. Conclusions and Future Work

In this paper, the design of a reconfigurable multiple PTRC based built-in-self-test scheme is presented that can generate more number of different patterns in less time based on the needs. Also a reconfigurable multiple run-time PTRC is proposed, which can activate any bit PTRC based on the requirements. A considerable reduction in test application cycles were achieved by using multiple PTRC compared to the single fixed-order TRC design. It is clear from the analysis that the introduction of the proposed scheme do not result in considerable increase in the total delay and area. When implemented on different FPGA devices, it is found that Virtex 5 has less delay and area. The future effort is to implement an efficient algorithm to reduce the test sequence length to obtain high fault coverage for VLSI applications.

Acknowledgements

The author would like to acknowledge the Department of Electronics and Communication, Saintgits College of Engineering, Kottayam for their technical support in simulations.

References