DRAFT: A Scanning Test Methodology for Dynamic and Partially Reconfigurable FPGAs

Manuel G. Gericota, Gustavo R. Alves
ISEP – Rua Dr. António Bernardino de Almeida
4200-072 Porto - PORTUGAL

Miguel L. Silva, José M. Ferreira
FEUP – Rua Dr. Roberto Frias
4200-465 Porto - PORTUGAL

Abstract

A new class of FPGAs that enable partial and dynamic reconfiguration without disturbing system operation, raised a new test challenge: how to assure a continuously fault free operation, independently of the circuit present after many reconfiguration processes.

A new on-line test method for those FPGAs is proposed, based on a scanning methodology and in the reuse of the IEEE 1149.1 Boundary Scan test infrastructure, already widely employed for In-System Programming.

1 Introduction

Reconfigurable logic devices, namely Field Programmable Gate Arrays (FPGAs), experienced a considerable expansion in the last few years, due in part to an increase in its size and complexity. The new dynamic and partially reconfigurable SRAM-based FPGAs (e. g. the Virtex family from Xilinx) have reinforced the advantages of these devices, by enabling partial and concurrent device reconfiguration without disturbing its operation. However, as in the rest of the semiconductor industry, the trend to smaller submicron scales increases the threat of electromigration, due to higher electronic current density in metal traces, causing FPGAs to be less reliable. Larger FPGA die sizes is another factor that increases the probability of failure. Certain defects related to manufacturing imperfections are not large enough to influence manufacturing tests, but after a large period of operation they become exposed, emerging as either stuck-at or transient faults [1]. Therefore a higher reliability level can only be achieved through the continuous test of the FPGA throughout system lifetime and through the introduction of fault tolerance mechanisms.

Using the dynamic and partially reconfigurable features and reusing the well established IEEE 1149.1 Boundary Scan (BS) test infrastructure [2], a new structural concurrent test approach based on a scanning strategy, with a very low test overhead at chip and board level, is proposed to test the FPGA Configurable Logic Blocks (CLBs).

In our approach, configuration memory is considered fault free and will not be tested. Nevertheless, the same test infrastructure could be used to perform a readback of the configuration data loaded into the FPGA, helping to detect faults in the configuration elements.

We start by a general description of the structural concurrent test solution envisaged for the CLBs of an FPGA, followed by two sections dedicated to the two components of the proposed solution: the mechanism used to free the CLBs to be tested; and the strategy used to test them. Concluding remarks are presented in the last section.

2 The DRAFT method

An FPGA comprises an array of independent CLBs, surrounded by a periphery of Input/Output Blocks (IOBs), which are interconnectable by configurable routing resources. In a given application, 100% usage of its resources is hardly ever achieved, even when independent hardware blocks dynamically share the same FPGA device (and hence a few blocks will always be free). Therefore, it is possible to consider a strategy to test temporarily unused blocks, without disturbing system operation, taking advantage of the dynamic and partially reconfigurable features offered by those FPGAs.

After being tested, unused defect-free CLBs remain available as spare parts, ready to replace others eventually found defective. Through a dynamic rotation mechanism, illustrated in figure 1, the CLBs currently being used are released for test, after their current functionality is replicated in previously tested blocks. Both CLBs (the original and its copy) remain active with the same state, inputs, outputs, and functionality, for at least one clock cycle in order to avoid output glitches.

Depending on the structure of the CLB to be tested and on the readback mechanism available, as well as on the functional specification currently implemented in the FPGA, the replication process may not be an easy task. When purely combinational functions are implemented on the CLB, there are no register values to be copied, but in the case of a sequential function, the register values must

♦ This work is supported by the Portuguese Foundation for Science and Technology (FCT), under contract POCTI/33842/ESE/2000

also be copied during the replication process. A temporary transfer path should be established between the registers in the two CLBs, to allow state information to be copied between them, and at least one clock pulse applied to both.

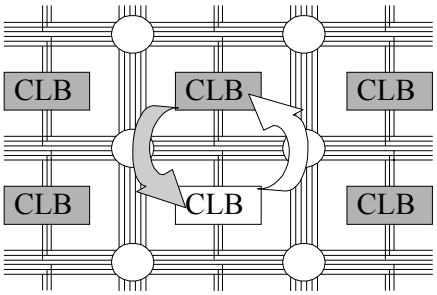

Figure 1. CLB replication and rotation of free resources

This solution guarantees that the whole FPGA can be tested, without disturbing the system operation, if at least one unused CLB is available in the current implementation. The introduction of fault tolerance features will however require more than one unused CLB, since a pool of spare resources has to be continuously available to replace those eventually found defective.

Our proposed DRAFT method (Dynamically Rotate And Free for Test) is controlled through the BS test infrastructure, including test application and response capturing operations.

3 Dynamic rotation strategies

The rotation method used in order to free CLBs for test should have a minimum influence (preferably none) in the system operation, as well as a reduced overhead in terms of reconfiguration cost. This cost depends on the number of reconfiguration frames needed to replicate and free each CLB, since a great number of frames would imply a longer test time. The impact of this process in the overall system operation is mainly related to the delays imposed by re-routed paths, since the rotation process might imply a longer path, reducing the maximum frequency of operation (in an FPGA the longest path delay determines the maximum frequency of operation).

Simulations performed for horizontal and vertical rotation scheme of the free CLB, using Virtex Xilinx FPGAs, whose results were presented by the authors in [3], have shown that the vertical rotation strategy, illustrated in figure 2, achieves lower costs. While the size of the reconfiguration files obtained by the application of both strategies to the same circuit implementation was very close, the influence over the maximum frequency of operation was substantially different, mainly due to a pair of dedicated paths that propagate carry signals vertically between adjacent CLBs. When the rotation process breaks a dedicated carry path, due to the insertion of the free CLB, the propagation of this carry signal between the nearest adjacent CLBs (above and below the free one) is re-established through generic routing resources, increasing the path delay.

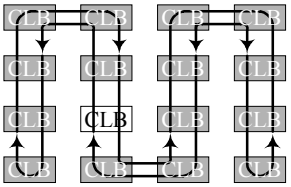

Figure 2. Dynamic rotation of the free CLB

This back and forth dynamic free CLB rotation across the chip implies a variable test latency. The time to again reach a given CLB alternates, according to the rotation direction, between a maximum and a minimum value, depending on the device size (number of CLB columns and rows).

The maximum fault detection latency is given by

$$\tau_{scan_{MAX}} = ((\#CLB_{rows} \times \#CLB_{columns}) - 2) \times 2 \times (t_{reconf} + t_{test})$$

The minimum fault detection latency is in turn given by

$$\tau_{scan_{min}} = 2 \times (t_{reconf} + t_{test})$$

where:

t_{reconf}: time needed to complete a CLB replication

t_{test}: time needed to test a free CLB

After a complete rotation, the initial routing is restored.

4 The test methodology

A Virtex CLB comprises two equal slices, each of them as shown in figure 3. The CLB test model (comprising the two slices) has 13 inputs (test vectors are applied to both slices of each CLB simultaneously) and 12 outputs (6 from each slice).

The BS test infrastructure is used to apply test vectors and to capture test responses, with the outputs of the CLB under test being routed to unused BS register cells associated to the IOBs. A User Test Register (UserTR) must be implemented in order to apply test vectors to the CLB under test, since its application through the BS register would affect the values present at the FPGA inputs. This UserTR comprises 13 cells, corresponding to the number of our CLB test model inputs. The number of CLBs occupied by this register (7), and the CLB needed to perform the rotation, represent the minimum hardware

overhead implied by our test methodology. Figure 4 illustrates the implementation of our test procedure.

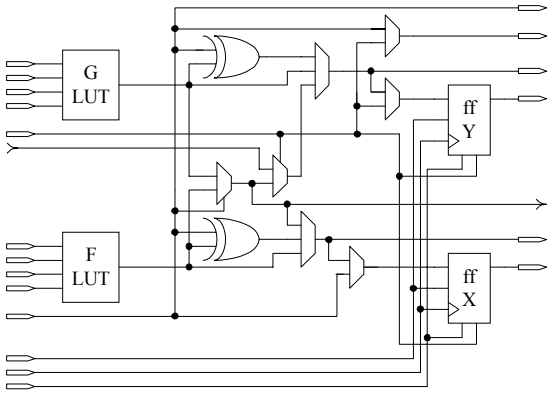

Figure 3. Test model of one Virtex slice structure

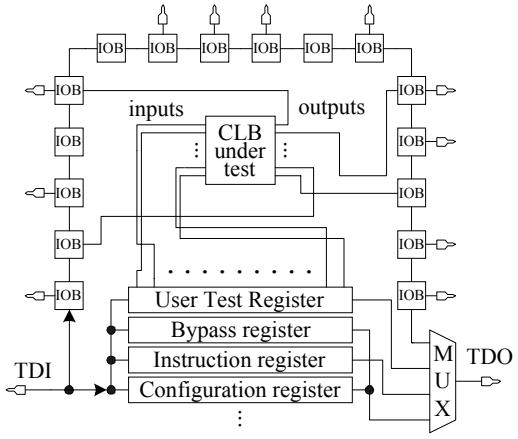

Figure 4. Test of a CLB

Shifting a test vector through the UserTR is very fast, in view of its reduced length. The time needed to shift the response vector depends on the length of the BS register (device size). Since the UserTR is part of the CLB array, the CLBs where it is implemented are also tested through the same process. This means that all hardware resources used to implement the test procedure are self-tested.

As the result of our analysis on the Virtex CLB test model structure, we concluded that four test phases are sufficient to exercise all possible configurations in the CLB. Since the implementation structure of the CLBs multiplexers and flip-flops was not known, a hybrid fault model was considered. To test the SRAM elements of the LUT, the value of each bit was toggled. By programming the LUTs (four in each CLB) to implement XOR and XNOR functions, which requires at least two test phases, it is easy to propagate any excited fault to a primary CLB output. All LUT input stuck-at faults are also detected, together with their corresponding addressing faults. For test purposes, Virtex CLB multiplexers have to be divided in two types: conventional and programmable multiplexers. At least three test configurations are needed to test programmable multiplexers, so a total number of four test configurations are needed to completely test the combinational part of the CLB. Flip-flops are all tested during these four phases. This procedure accounts for 100% fault coverage under the considered fault model. Since the reconfiguration process is slow, the small number of test phases is a good measure of our reduced test time. Table 1 summarises our experimental results.

Test session	
1st test phase	18 test applications
2nd test phase	3 test applications
3rd test phase	2 test applications
4th test phase	16 test applications

Table 1. Experimental test results

5 Conclusion

The solution proposed enables the implementation of a concurrent test method that reuses the standard BS test infrastructure and the partial dynamic reconfiguration features of recent FPGA devices, in order to improve the reliability of reconfigurable hardware systems, with minimal test overhead, in a way that is completely transparent to the system operation.

Emphasis is being placed on the development of computational tools to introduce a higher degree of automation in the whole process.

References

[1] Shnidman, N. R., Mangione-Smith, H., Potkonjak, M., "On-Line Fault Detection for Bus-Based Field Programmable Gate Arrays", *IEEE Trans. on VLSI Systems*, Vol. 6, Nº 4, pp. 656-666, December 1998.

[2] *IEEE Standard Test Access Port and Boundary Scan Architecture* (IEEE Std 1149.1), IEEE Standards Board, October 1993.

[3] Gericota, M. G., Alves, G. R., Ferreira, J. M., "Dynamically Rotate And Free for Test: The Path for FPGA Concurrent Test", *2nd IEEE Latin-American Test Work.*, February 2001.