Diagnosing Single Faults for Interconnects in SRAM Based FPGAs [1]

Yinlei Yu, Jian Xu and Wei Kang Huang
ASIC & System State Key Laboratory
Fudan University
Shanghai, 200433, China

Fabrizio Lombardi
Electrical & Computer Engineering
Northeastern University
Boston, MA 02115, USA

Abstract: *This paper presents a method to diagnose faults in FPGA interconnection resources. A single fault model is given. Under the given model, a diagnosing method is proposed. At most five programming steps in the proposed method is required if adaptive testing scheme is used. For non-adaptive test, eight programming steps is required to diagnose all the possible faults under the given single fault model. The accuracy of the fault diagnosing is one segment for a segment stuck-at or stuck-open fault, a segment pair for a bridge fault, a switch for switch stuck-on or stuck-off fault.*

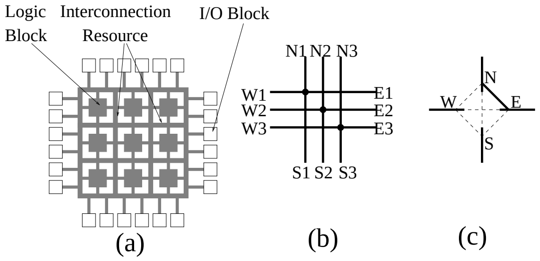
Keywords: *FPGA, testing, fault model, fault diagnosis*

1. Introduction

With the wide application of FPGAs, its testing technology has been developing rapidly. There are many recent articles on the FPGA testing [1]-[5]. Some researchers proposed methods for testing the logic block, while some others deal with the interconnection resources. In [2] and [3], only the verification test is discussed. The method proposed in [4] requires too many programming steps, while only fault detection is discussed in [5].

This paper proposes a method that needs at most five programming steps to diagnose faults in FPGA interconnection resources. A single fault model is used in the discussion. The configurations in the first three programming step is similar to the configurations in [5]. With the proposed method, the accuracy of fault location is a single segment for a segment stuck-at or stuck-open fault, a segment pair for a bridge fault and a switch for switch stuck-on(off) fault under the single fault assumption.

The following parts of this paper are organized as follows: The second section introduces the simplified model of the interconnection resource and its single fault model. Section 3 is an intensive description and analysis of the proposed programming method. Application to XILINX FPGAs of the proposed method is given in Section 4 and the last section summarizes the paper.



(a) (b) (c)

Figure 1. The Basic Structure of an FPGA

2. Structure Model and Fault Modeling for FPGAs

The basic structure of an FPGA consists of four parts: Logic Blocks, Interconnection Resources, I/O Blocks and Programming Circuits. The basic structure without programming circuits is shown in Figure 1(a). The Interconnection Resource consists of tracks and switch matrices. There are horizontal and vertical tracks. Every track is composed of all the segments and switchboxes arranged in a straight line, while several switches form a switch matrix. The area between two switch matrices is referred to as **segment block**. There are n_t segments in each segment block. A switch matrix has n_t terminals in each direction (side). We assume that the switchboxes only exist on the diagonal of a switch matrix. Figure 1(b) shows a switch matrix with three switchboxes and three terminals in each side. A switchbox is represented by a small dot in the figure. The model of a switchbox is shown in Figure 1(c). There are six switches in a switchbox, which can connect W and N, and S, N and E, S and E, W and E and N and S respectively when they are turned on. A switch that is used to connect W and N (N and E, . . ., N and S) is referred to as switch NW (WS, NE, SE, WE or NS). A switch is on a track if it can connect two segments in the track. A thin line is used to denote a switch that is OFF and a thick line to represent a switch that is ON, as shown in Figure 1(c). In a single switchbox, more than one switch is permitted to be on.

In our discussion, a single fault model is used. It is assumed that there is at most one fault in a FPGA's inter-

[1] This work is a part of Project 699773011 supported by National Natural Science Foundation of China.

connection resources. The switch programming faults need special test and are not discussed here. We assume that the programming circuit had already undergone the test and no fault had been found, so they will not be considered in this paper. It is also assumed that floating output is reported as ’0’. The following four type of single faults are considered. (1) Single segment stuck-at-1(0) fault. (2) Single segment stuck-open fault. (3) Single switch stuck-on(off) fault. (4) Single bridge fault(connects two segments). One of the following two kinds of bridge faults can occur: (i) Parallel Bridge (PB) fault: A bridge fault between two adjacent parallel segments in a segment block. (ii)Orthogonal Bridge (OB) fault: A bridge fault between a vertical segment and a horizontal segment surrounding a switch matrix. Line-OR output of the bridge fault is assumed.

3. Fault Detection and Diagnosis

In our proposed method, the first three programming steps are same as in [5], which can detect all single faults. However, we focus our discussion on the diagnosis issue. At most other two steps are required for locating a detected fault, and the configurations of these two steps are dependent on the test results of the first three steps.

Without loss of generality, we take for example the FPGA interconnection resource with three horizontal and vertical channels, each channel containing three tracks, as shown in Figure 2(a). There are nine ports on the each side of the FPGA. The ports on the top (bottom) are named as tv_1 to tv_9 (bv_1 to bv_9) and the ports on the left (right) side are named as lh_1 to lh_9 (rh_1 to rh_9).

The followings are the configurations, test patterns and diagnosis analysis of the programming steps. The configurations are shown in Figure 2 and the test patterns are shown in Table 1.

Program Step 1: All the NS and WE switches are programmed to ON. The following single faults can be detected and diagnosed: (1) Any segment stuck-at-1(0) and stuck-open fault can be located to one track. (2) Any PB faults can be located to the two tracks where the bridged segments are in. (3) Any OB faults can be located to the crossing point of the two bridged segments. (4) Any stuck-on fault of the NE, NW, SE or SW switches can be located to a switchbox. (5) Any stuck-off fault of the NS or WE switches can be located to a net.

Program Step 2: All the NE and WS switches are programmed to ON. The following faults can be detected and the faults diagnosis result of the first two steps are also listed as follows: (1) The segment stuck-at and stuck-open fault can be located to one net. Since every net in Step 1 have at most *one* segment shared with a net in Step 2, the segment stuck-at fault can be located to one segment in this step. (2) The PB fault can be located to the pair of segment where the bridge fault occurs for the same reason. (3) The stuck-on fault of a NW, SE, NS or WE switch can be located to the switchboxes two nets meet. (4) The stuck-off fault of an NE and SW switch can be located to the switches in a net.

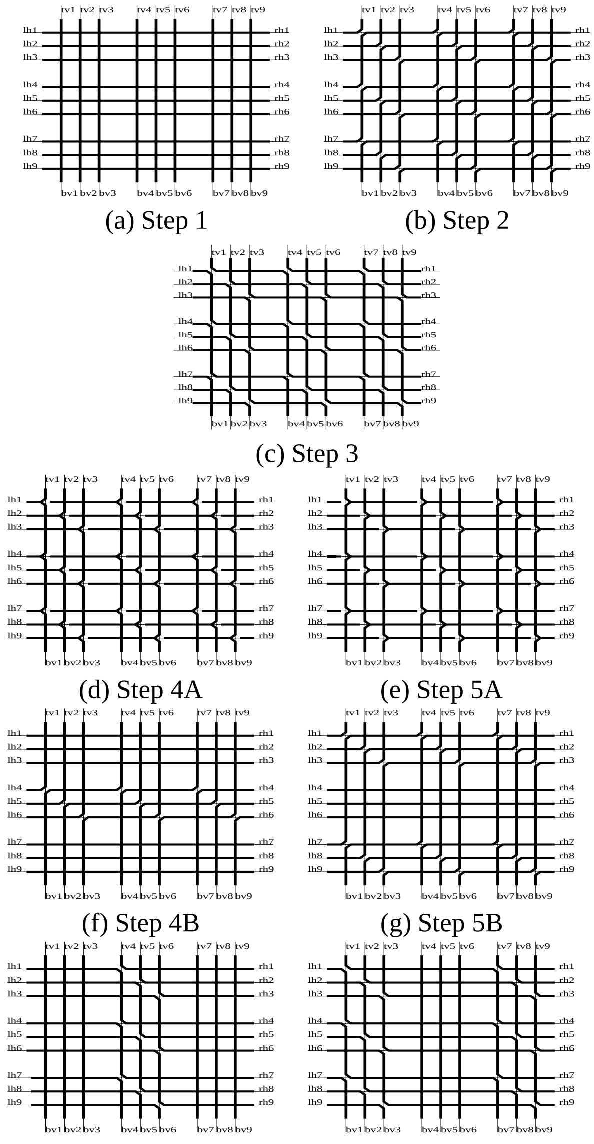

tv1 tv2 tv3 tv4 tv5 tv6 tv7 tv8 tv9
lh1 lh2 lh3 lh4 lh5 lh6 lh7 lh8 lh9
rh1 rh2 rh3 rh4 rh5 rh6 rh7 rh8 rh9
bv1 bv2 bv3 bv4 bv5 bv6 bv7 bv8 bv9
(a) Step 1
(b) Step 2
(c) Step 3
(d) Step 4A
(e) Step 5A
(f) Step 4B
(g) Step 5B
(h) Step 4C
(i) Step 5C

Figure 2. The Configurations for the Programming Steps

Program Step 3: All the NW and SE switches are programmed to ON. The following single faults can be detected and located further: (1) The stuck-on fault of any NE, SW, NS or WE switch can be located to all the switchboxes two nets meet. (2) The stuck-off fault of any NW or SE switch can be located to all the switches in a net.

Discussion 1: After the above steps, the fault diagno-

Step 1	lh_1	lh_2	lh_3	lh_4	lh_5	lh_6	lh_7	lh_8	lh_9	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9
Step 2	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9	rh_1	rh_2	rh_3	rh_4	rh_5	rh_6	rh_7	rh_8	rh_9
Step 3	lh_1	lh_2	lh_3	lh_4	lh_5	lh_6	lh_7	lh_8	lh_9	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9
	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0
	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0
	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1

Step 4A	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9
Step 5A	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9
	1	0	0	0	0	0	1	0	0
	0	1	0	0	0	0	0	1	0
	0	0	1	0	0	0	0	0	1
	0	0	0	1	0	0	0	0	0
	0	0	0	0	1	0	0	0	0
	0	0	0	0	0	1	0	0	0

Step 4B	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9	rh_1	rh_2	rh_3	rh_4	rh_5	rh_6	rh_7	rh_8	rh_9
Step 5B	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9	rh_1	rh_2	rh_3	rh_4	rh_5	rh_6	rh_7	rh_8	rh_9
Step 4C	lh_1	lh_2	lh_3	lh_4	lh_5	lh_6	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	lh_7	lh_8	lh_9	tv_7	tv_8	tv_9
Step 5C	lh_1	lh_2	lh_3	lh_4	lh_5	lh_6	lh_7	lh_8	lh_9	tv_1	tv_2	tv_3	tv_4	tv_5	tv_6	tv_7	tv_8	tv_9
	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0
	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0
	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1
	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0

Table 1. Test Patterns for Programming Steps

sis result can be categorized into following **Seven** cases: **Case(1)** Any segment stuck-at and stuck-open fault can be located to one segment. **Case(2)** Any PB fault can be located to the pair of bridge segments. **Case(3)** Any OB fault can be located to the crosspoint of the two bridged segments. **Case(4)** Any single switch stuck-on fault can be detected twice in the three programming steps. For a switch stuck-on fault, the possible faulty switches detected in two different types can only have a pair of switch in a common switchbox. So this kind of fault can be located to a pair of switches (NE and SW, NW and SE or NS and WE) in a switchbox. **Case(5)** Any stuck-off fault of the NS or WE switch can be located to all the NS(WE) switches in a horizontal (vertical) track. **Case(6)** Any stuck-off fault of the NE or WS switch can be located to all the NE or WS switches in a top-right to bottom-left net as in Step 2. **Case(7)** Any stuck-off faults of the NW or SE switch can be located to all the NW or SE switches in a top-right to bottom-left net as in Step 3.

Conclusion 1: After the above three steps, all the faults in the given fault model can be detected. Concerning diagnosis, for the case (1) to (3) in **Discussion 1**, no further diagnosis is required. For the case (4) to (7), adaptive method should be used. The diagnostic steps of the adaptive method are discussed as follows.

The Diagnostic Steps for Case (4), (6),(7): In Case (4), a switch stuck-on fault is located to a pair of switch in a switchbox. An additional programming step is required to determine which switch is faulty. According to the preliminary information from the first three steps, the switchbox where the faulty switch is in. **Programming Step 4A** is required. In its configuration, all the NW and SW switches are programmed to ON. So any stuck-on fault of the NE, SE or WE switches and any stuck-off fault of the NW or SW switches can be located to a horizontal track.

Discussion 4A: For **Case (4)**, in Step 1 to 3, the stuck-on fault of the switches is located to a pair of switches, *i.e.* {NS, WE}, {NE, WS} or {NW, SE} of a switchbox. In Step 4A, the stuck-on fault of NE, SE or WE can be detected, while other stuck-on faults cannot be detected. Since NE, SE and WE belong to different switch pairs, the stuck-on fault can be distinguished and located to a single switch in a switchbox.

For **Case (6) and (7)**, **Programming Step 5A** is necessary besides Step 4A. In this step, all the NE and SE switches are programmed to ON. After it, any stuck-on fault of the NW, SW or WE switch and any stuck-off fault of the NE or SE switch can be located to the horizontal track of the faulty switch.

Discussion 5A: By Step 2 and 3, the stuck-off fault of the NW, NW SE or SW switch can be located to all switches in a diagonal net as shown in Figure 2(b) and (c). In step 4A and 5A, the fault can be located to a horizontal track. The switch NW and SE, NE and SW have the same syndrome in Step 2 and 3, while in

step 4A and 5A, NE and SE, NW and SW have the same syndrome. Combining the results from step 1,2 and 3 and from 4A and 5A, we can determine which type of switch (NE, NW, SE or SW) the faulty switch is. Because there are only two different type switches at the place where a diagonal net meets a horizontal net, the fault can be located to a single switch.

The Diagnostic Steps for Case (5): The stuck-off fault of the NS or WE switch must be discussed separately. From the Step 1, we know the type and the track of the faulty switch. Two additional steps are required to test the odd numbered switches and even numbered switches separately. For NS switch stuck-off fault, programming step 4B and 5B should be applied. In **Programming Step 4B**, all the NS and WE switches in odd numbered channels are programmed to ON, while all the NW and SE switches are programmed to ON in even numbered tracks. In **Programming Step 5B**, all the NS and WE switches in even numbered horizontal channels are programmed to ON, while all the NW and SE switches are programmed to ON in odd numbered horizontal channels.

After Step 4B and 5B, every NS switch is shared by two different nets, so the location of its stuck-off fault is determined.

For stuck-off fault of WE switches, the fault detection is similar to that of NS switch. **Programming step 4C and 5C** should be applied. In step 4C, all the NS and WE switches in odd-numbered vertical channels are programmed to ON, while all the NW and SE switches are ON in the rest switch matrices. In step 5C, all the NS and WE switches in even numbered vertical channels are programmed to ON, while all the NW and SE switches are ON in the rest switch matrices.

Fault diagnosis in non-adaptive scheme

Non-adaptive scheme can be also used for fault diagnosis to achieve the same accuracy. All the programming steps should be used except Step 1, which is covered by the combination of configuration 4B, 4C, 5B and 5C. So *eight* programming steps are needed for non-adaptive method.

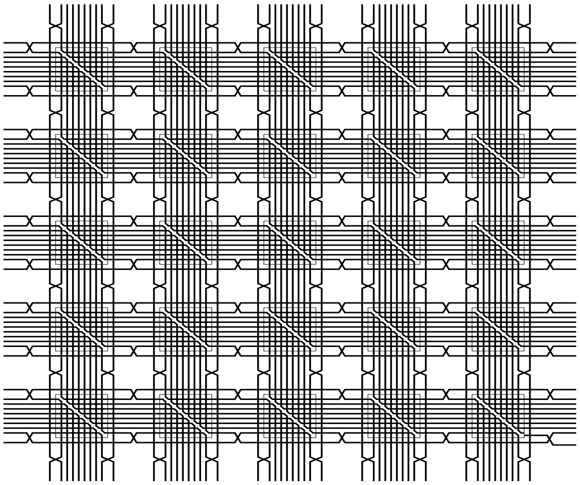


Figure 3. The configuration for step 2 in Xilinx 4000

4. Application to XILINX FPGAs

The method discussed in Section 3 can be applied to XILINX FPGAs. Let's take XILINX 4000 family as an example. In an FPGA of the XILINX 4000 family, all switches exist in the diagonal position of the connection matrix, which is the same as the structure given in section 2. However, there are double-length segments and quadruple-length segments besides the single-length ones, which is different from the structure model above. Therefore, some modest modification is required to apply the proposed diagnosis method on FPGAs in XILINX 4000 family.

To apply the proposed diagnosis method, the configurations in all the steps except step 1 have to be modified. In the structure of 4000 family, different type of the segments are separate. So they can be diagnosed separately except for bridge faults of different types. For OB faults, all the faults can be located in Step 1. For the PB faults, the process is also similar: The switches in a switch matrix to be turned on in all the steps are same as the proposed configurations in step 1, 2, 3, 4A and 5A. For Step 4B, 5B, 4C and 5C, since we can know the type of segment (single, double or quadruple) the stuck-off switch connects, we can just program the segment-switches of the same type in the similar way. The modified Step 2 is shown in Figure 3.

In practical FPGAs, there are not enough pads. This can be alleviated by configuring the nets on which same test patterns are applied connected together to share a pad. Certainly this will reduce the accuracy of fault diagnosis, but the problem can be solved by using additional programming steps. The test patterns used for each configuration are also similar.

The given method can also be applied to XILINX 3000 with some modification but it will not be discussed here.

5. Summary

In this paper, we have presented a single fault model and proposed a general diagnosis method for SRAM-based FPGAs. The method is based on a simplified model of the FPGA interconnect structure. The method consists of four to five programming steps for adaptive scheme. The number of the steps depends on the nature of the fault detected. Based on the method proposed in [5], first three steps can detect any single fault. The other programming steps are designed to locate the fault(s) precisely. The proposed diagnosis method can be easily applied to real FPGAs after certain modification. The XILINX 4000 family has been used as an application example to the proposed method.

References

[1] T. Inoue, et. al., "Universal Test Complexity of Field-Programmable Gate Arrays" *Proc. Asian Test Symp.*, pp. 259-265 Bangalora, India, November 1995.

[2] W. H. Kautz, "Testing for Fault in Wiring Networks", *IEEE Trans. on Computers*, Vol.C-23, No.4, pp. 358-363, 1974.

[3] M. Renovell, J. Figueras and Y. Zorian. "Testing the Interconnect Structure of Unconfigurated FPGA", *IEEE European Test Workshop*, pp. 125-129, Sete, France, June 1996.

[4] W. K. Huang, X. T. Chen and F.Lombardi, "On the Diagnosis of Programmable Interconnect Systems: Theory and Application" *Proc. 14th IEEE VLSI Test Symposium*, pp. 204-209, Princeton, NJ, USA, May 1996.

[5] M. Renovell, J. M. Portal, J. Figueras, Y. Zorian, "Testing the Interconnect of RAM-Based FPGAs", *IEEE Design & Test*, pp. 45-50, January-March, 1998.

[6] H. Michinishi, et. al., "Testing for the programming circuits of LUT based FPGAs", *Proc. Asia Test Symp.*, pp. 242-247 , Japan, November, 1997.