12th IEEE European Test Symposium (ETS'07), pp. 131-136, May, 2007.

Delay Testing for Application-Specific Interconnects of FPGAs based on Inphase Structure

Satoshi Ohtake Kousuke Yabuki Hideo Fujiwara
Graduate School of Information Science, Nara Institute of Science and Technology
Kansai Science City, Nara 630-0192, Japan
E-mail: {*ohtake, fujiwara*}@*is.naist.jp*

Abstract

This paper presents a method of path delay fault testing for application-specific interconnects in field-programmable gate arrays (FPGAs). The paper shows that if the circuit structure of a configuration corresponding to an application is inphase structure, all the paths in the circuit can be robustly tested by using two configurations with test application time $d+2$ *for each configuration where* d *is the maximum sequential depth of the circuit. The scheme for inphase structure is extended for acyclic structure. The proposed method reduces over-testing by excluding paths of outside the configured area in the FPGA.*

1. Introduction

Recent years, field-programmable gate arrays (FPGAs) which are reconfigurable device are widely used. Since the most of the area in a FPGA is occupied by interconnects, testing methodologies for interconnects are important issue[1][2]. With increasing the operational speed and integration density of FPGAs, delay testing is getting important to capture the newly elicited defects[3][4]. Moreover, in these several years, in order to reduce the cost of ASIC (Application Specific Integrated Circuit) development, the business model as using FPGAs instead of ASICs have been proposed [5]. In this case, the testing of a whole FPGA can be simplified down to only the testing its application-dependent functionality and it facilitate speed and quality of testing[6]. Testing only a part, which is used for configuration of some specific application, in a FPGA is called application-specific testing. Elimination of testing the unused portion of the FPGA can reduce over-testing and then manufacturing yield can be improved[7].

Testing for an FPGA is generally performed by loading a test structure which is called a *test configuration*, applying test vectors and observing responses. If it is necessary for testing interconnects and configurable logic blocks (CLBs) in the FPGA, this flow is repeated several times. For application-specific testing of interconnects in an FPGA, interconnects are configured as the same routing as its application and CLBs are only changed to appropriate functions for testing. Similarly, for application-specific testing of CLBs in the FPGA, CLBs are configured as the same functions as its application and interconnects are appropriately configured for testing.

In this paper, we target delay fault testing of application-specific interconnects of FPGAs and propose a method of path delay fault testing for the interconnects. We first define inphase structure as a class of easily testable sequential circuits. Then we show that if an application of a sequential circuit to be loaded into an FPGA is inphase structure, all the path delay faults can be robustly tested with $d+2$ cycles where d is the sequential depth of the circuit by using two test configurations that are its networks of AND function and of OR function [6] and square waves, whose value is turned over for each cycle, for every primary input as a test sequence.

Furthermore, we show that, for any acyclic sequential circuit, by making all the outputs of flip-flops (FFs) inverted and each gate are replaced appropriately with AND or OR, all the paths are tested under either robust condition or functional sensitization condition with $d+2$ cycles. In this case, the number of test configurations is also two. In the experimental results, we show how many paths can be robustly tested for acyclic-transformed ISCAS'89 benchmark circuits.

This paper is organized as follows. Section 2 introduces related works and discusses their problems. In Section 3, we define inphase structure and then we propose methods of path delay fault testing for application-specific interconnects for sequential circuits with inphase structure and for that with acyclic structure. Then we show experimental results in Section 4 and conclude this paper in Section 5.

2. Prior Work

2.1 Testing of FPGAs

In this paper, an FPGA is assumed to be composed of an interconnect network and CLBs. The interconnect network is composed of signal lines and switch matrices (SMs). A CLB is composed of a look-up table (LUT) and an FF. Wiring between CLBs is performed by programming configuration memories of SMs. A CLB can realize any logic function by programming its configuration memory.

For an application of an FPGA, its configuration uses only a part of the FPGA to implement the application and the other part of the FPGA is not used. Application-independent testing must guarantee that any configuration can work correctly. Application-specific testing must guarantee that only the configuration for the application can work correctly. For application-specific testing of an FPGA with respect to a configuration, faults are assumed to be presented only in a part used by the configuration. The part of the configuration in which target faults are presented is not changed during testing while the other part of the FPGA can be programmed with any configuration for testing purpose. For example, if faults in interconnects are targeted, configurations used for testing, called *test configurations*, preserve the original configuration for the interconnects and do not need to preserve the original configuration for the CLBs and the unused part. In contrast, if faults in CLBs are targeted, test configurations preserve the original configuration for CLBs and do not need to preserve the original configuration for interconnects and the unused part.

Application-specific testing targets faults in interconnects and CLBs which are actually used for the application. Therefore, unused parts and functions are eliminated from targets of testing and then over-testing can be reduced. It induces reduction of test application time and virtual improvement of production yield of FPGAs. Notice that, the number of test configurations for an FPGA is very dependent on its test application time because time required for programming a configuration dominates the whole test application time.

Furthermore, if defects which induce delay are strictly tested, path delay fault (PDF) model must be considered. Since paths can not be defined without considering some application, it is intractable to test them under the application-independent strategy. A method for testing of interconnect PDFs have been proposed by Tahoori et al. [7]. In this method, only two test configurations are used for an acyclic sequential circuit and four test configurations are used for an general sequential circuit. In the following subsection 2.2 and 2.3, we introduce application of the method for acyclic sequential circuits and that for general sequential circuits, respectively. Then we discuss problems of the method in subsection 2.4.

2.2 A Network of AND (OR) Function

In the PDF model, each path can have two delay faults, slow-to-rise and slow-to fall at its ending point. For an AND gate, when rising transitions are arrived at all the inputs of the AND gate, its output also has a rising transition. If arrival of a rising transition is delayed among them, the delay can be propagated through the output robustly. In contrast, for an OR gate, when falling transitions are arrived at all the inputs of the OR gate, its output has falling transition. If arrival of a falling transition is delayed among them, the delay can be propagated through the output robustly. Tahoori et al. utilize these properties, a network of AND function and that of OR function are used as test configurations. In this paper, we define an AND (OR) transformation for a sequential circuit as follows.

Definition 1: *(AND (OR) transformation)* For a sequential circuit S, a transformation that replace each gate in S with AND (resp. OR) and the transformed circuit are called *an AND (resp. OR) transformation* of S and the *AND (resp. OR) network* of S, respectively. □

Notice that each inverter or buffer is assumed to be replaced with one input AND (resp. OR) gate in the AND (OR) transformation.

In Tahoori's method, a given configuration which realizes an acyclic sequential circuit for an FPGA is transformed into its AND (resp. OR) network by replacing each LUT function used in the configuration with AND (resp. OR) function. The transformed configurations of the AND network and the OR network are used as test configurations. For each configuration, a test sequence to launch transition at primary inputs is calculated so that all the inputs of each LUT can have the same appropriate transition simultaneously. Figure 1 shows an example of a programmed part that realizes an acyclic AND network on an FPGA. Consider making rising transitions for all the inputs of the LUT G1. Since the sequential depth (the number of FFs on the interested path) between the primary input X2 and G1 is smaller than that between the primary input X1 and G1 and the difference is one, a rising transition is launched on X2 at one cycle after when a rising transition is launched on X1 and then rising transition can be arrived at the inputs of G1 simultaneously. And then the output can have the rising transition. In the same way as this calculation, rising transitions must be launched on the primary inputs X3, X4 and X5 at the same time as X1 and then rising transitions can be launched on all the inputs of each LUT simultaneously. If any path has a rising PDF, the controlling value of AND gate as the error induced by the PDF appeared at the ending point of the path. It is obvious that the error can be propagated to some primary output. Similarly, all the falling PDFs on the application-specific interconnects can be tested using the OR network of the configuration. Thus, PDFs on all the paths composed of application-specific interconnects can be tested.

2.3 Partitioning of a Cyclic Sequential Circuit

For sequential circuits with cycles, the configuration is partitioned into two configurations so that each configuration has no sequential loop. An algorithm for partitioning has been proposed in [7]. For example, a cyclic sequential circuit shown in Fig. 2 is partitioned into to sub-circuits: one has only forward signal lines and the other has back-

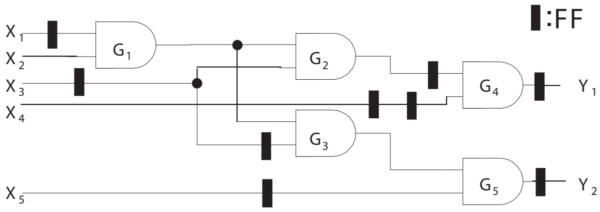

Figure 1. The AND network of an acyclic sequential circuit.

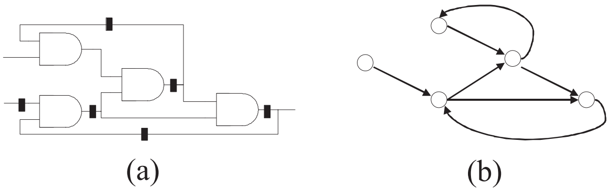

Figure 2. A sequential circuit with cycles.

ward signal lines as shown in Fig. 3. By using this partitioning, the method described in subsection 2.2 can be applied to each partitioned configurations. Therefore, the number of test configurations for testing cyclic sequential circuit is four.

2.4 Problems of the Prior Work

The prior work [7] has the following two problems.

First, a pair of segments that forms reconvergence structure may have problem. If the difference between sequential depths of these segments is odd number, all the inputs of the gate that is the ending point of the reconvergence structure can not have the same direction of transition at the same time. For the gate, it is possible to justify non-robust off-input values however no transition can be launched on the inputs of gates in the following sub-circuit. An example of this situation is shown in Fig. 4. One way to solve this problem is to partition the reconvergence structures into several configurations. However, in this case, the number of test configuration is increased and it induces overhead of test application time.

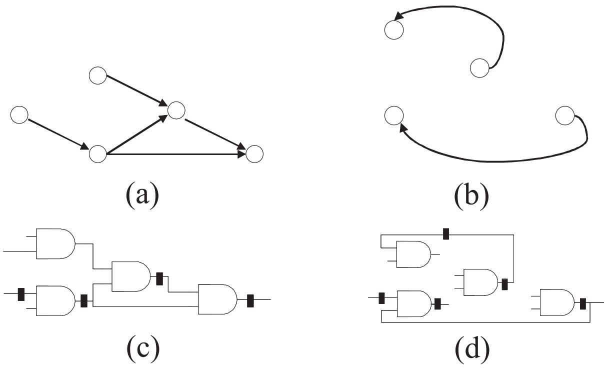

Figure 3. Partitioned acyclic sequential circuits.

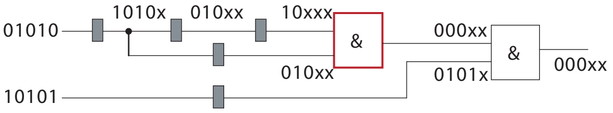

Figure 4. An example of a reconvergence structure.

Second, interconnects for accessing the partitioned configurations may have problem. If the routing for the inputs and the outputs of the partitioned configuration uses the unused part of the original configuration, it induces over-testing by testing the unused part. Furthermore, if the original routing is concentrated around some input or output of the partitioned configuration or the input and output ports of the FPGA are occupied by the original configuration, it may be very difficult to access some input or output of the partitioned configuration.

In this paper, we solve the first problem in the next section.

3. Proposed Method

In this section, we classify circuits which are realized by the given configurations into inphase structure and acyclic structure. For each structure, we propose methods for generation of test configurations, generation of test sequences, and application of these test configurations and test sequences.

3.1 Inphase Structure

We first define a circuit model.

Definition 2: *(Circuit graph)* [8] A *circuit graph* for a sequential circuit S is a directed graph $G = (V, A, w)$.

- V is the set of vertexes representing primary inputs, primary outputs and gates in S.
- $A \subset V \times V$ is the set of arcs representing FFs and wires in S.
- $w : A \mapsto \{0\} \cup \mathcal{N}$ (natural numbers) defines the weights of the arcs where a weight is the number of FFs between the corresponding gates. □

Then inphase structure is defined by using this circuit model as follows.

Definition 3: *(Inphase structure)* Let $G = (V, A, w)$ be the circuit graph of an acyclic sequential circuit S. S is said to be *inphase structure* if $b(v) \in \{0, 1\}$ that satisfies the following condition can be assigned to each node $v \in V$.

$$b(v_j) = ((b(v_i) + w(a)) \; mod \, 2 \quad \forall a(v_i, v_j),$$

where $b(v)$ is called *direction of phase* and $w(a)$ denotes a weight assigned to $a \in A$. □

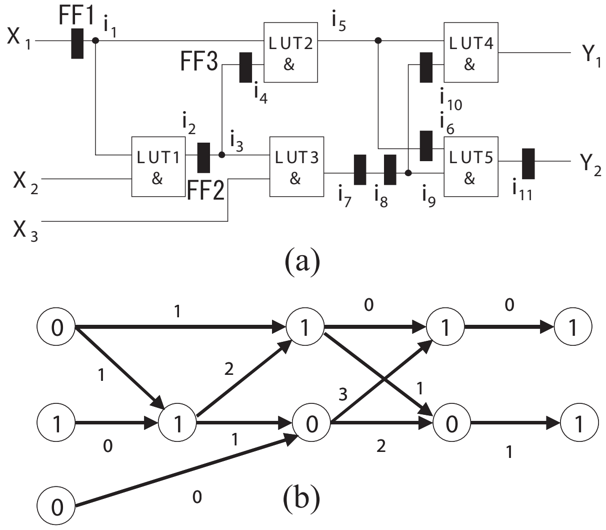

Figure 5. A sequential circuit with inphase structure

For example, a sequential circuit with inphase structure and its circuit graph is shown in Fig. 5. In the circuit graph, the numbers over each arc a and in each vertex v denote $w(a)$ and $b(v)$, respectively.

For an FPGA that has an inphase sequential circuit as a configuration, its AND network and its OR network are used as test configurations. For each configuration, square waves of length $SD(S)+2$ are applied to as a test sequence, where $SD(S)$ and square waves denote the sequential depth of S (the maximum number of flip-flops on paths in S) and a bit stream composed of reciprocally assigned 0s and 1s, respectively.

Theorem 1 (Testability of inphase AND network) *Let $G=(V,A,w)$ be the circuit graph of an inphase sequential circuit S. Let S^{AND} and v^d be the AND network of S and the starting vertex of a path that has $SD(S)$, respectively. Let T be an input sequence, composed of square waves of length $SD(S)+2$, of S such that, the square wave for each primary input corresponding to v_i with $b(v_i)=b(v^d)$ starts from 0 and the other square waves start from 1 if direction of phase $b(v)\in\{0,1\}$ is assigned for each vertex $v\in V$. Then T is an robust test for the rising PDF on any path in S^{AND}.*

Proof: Let $D_{MAX}(v)$ be the maximum sum of weights on paths between sources and v in G. From Definition 3 and the property of an AND gate, when T is applied to S^{AND}, square waves whose initial value is 0 are applied to all the inputs of the gate corresponding to each $v\in V$ during $SD(S)-D_{MAX}(v)+1\geq 2$ cycles after the $(D_{MAX}(v)+1)$-th cycle if $b(v)=b(v^d)$. Otherwise, square waves whose initial value is 1 are applied during $SD(S)-D_{MAX}(v)+1\geq 3$ cycles after the $(D_{MAX}(v)+1)$-th cycle. Therefore, at least one rising transition can be propagated through each path in S^{AND}.

Here, let v^e be the ending vertex of a path that has $SD(S)$. Let $O=(o_1,o_2,\ldots,o_n)$ be a primary output vector of S^{AND} such that o_i is the value of the i-th primary output at time $SD(S)+2$ if its corresponding vertex v_{o_i} has the same value as $b(v^e)$, where n is the number of primary outputs of S. Otherwise, o_i is the value at time $SD(S)+1$. The primary output vector is called the *output observation vector* of S^{AND}. For fault free S^{AND}, the observation vector must be $O=(1,1,\ldots,1)$.

Let f_p be a rising PDF on any path p in S^{AND}. No rising transition can occur at the ending FF or the primary output of p. In other words, the controlling value 0 is propagated during the application of T to the ending point and it is propagated to primary outputs that are the transitive fanouts of the point. Thus, corresponding coordinates in the output observation vector has 0s and it is different from that for the fault free network. Therefore f_p is observed. Furthermore, since all the off inputs of gates on p satisfy the robust off-input condition [9], T is a robust test for any rising PDF in S^{AND}. □

Theorem 2 (Testability of inphase OR network) *Let $G=(V,A,w)$ be the circuit graph of an inphase sequential circuit S. Let S^{OR} and v^d be the OR network of S and the starting vertex of a path that has $SD(S)$, respectively. Let T be an input sequence, composed of square waves of length $SD(S)+2$, of S such that, the square wave for each primary input corresponding to v_i with $b(v_i)=b(v^d)$ starts from 1 and the other square waves start from 0 if direction of phase $b(v)\in\{0,1\}$ is assigned for each vertex $v\in V$. Then T is an robust test for the falling PDF on any path in S^{OR}.*

Proof: It is obvious from the duality of the logical algebra and the proof of Theorem 1. □

For example, the maximum sequential depth of the inphase AND network shown Fig. 5 is 5 and the primary input corresponding to the starting point of the sequential path with depth 5 is X_1. From Theorem 1, square waves of length 7 with initial value 0 are applied to X_1 and X_3, that have the same direction of phase as $b(X_1)$, respectively, and a square wave of length 7 with initial value 1 is applied to X_2 that has the different direction of phase from $b(X_1)$. For the fault free network, at least one rising transition is propagated through each path in the network. Suppose that the path between FF2 and FF3 in Fig. 1 has a rising PDF. In this case, value 1 cannot be propagated to FF3 and the output value of FF3 is fix to 0. The value 0 is the controlling value for LUT2 with AND function and then the error can be propagated through subcircuit followed by LUT2. Consequently, the errors induced by the PDF are observed at the primary outputs Y_1 and Y_2.

3.2 Acyclic Structure

To extend the property of inphase sequential circuits into acyclic sequential circuits, we introduce a transformation of rising (falling) sensitization. Given an FPGA and a configuration that realizes an acyclic sequential circuit S, PDFs corresponding to the given configuration on the FPGA are tested by using square waves of length $SD(S)+2$ and configurations obtained by the transformations from the given configuration.

Definition 4: *(Transformation for path sensitization)* For an acyclic sequential circuit S, let $l(i)$ be the parity of the minimum sequential depth among sequential paths in the output cone of i. The following procedure is called the *transformation of rising (resp. falling) sensitization* for S.

1. For each gate g, g is replaced with an AND (resp. OR) if $l(i)=0$ where i is a signal line driven by g. Otherwise, g is replaced with an OR (resp. AND).
2. For each FF, an inverter is added to its output.

The circuit S^R (resp. S^L) obtained from S by this procedure is called the *rising (resp. falling) sensitized circuit.* □

Corollary 1 (Property of gate inputs in S^R) *For a rising sensitized circuit S^R, when inphase square waves with the parity of $SD(S^R)$ as their initial values of length $SD(S^R)+2$ are applied to all the primary inputs of S^R, inphase square waves with the parity of $SD(S^R)-SD(IC(g))$ as its initial value can be applied to all the inputs of each gate g in S^R after $SD(IC(g))+1$ cycles, where $IC(g)$ denotes the input cone of g.*

Theorem 3 (Testability of PDFs in S^R) *Let T be an input sequence of S^R composed of inphase square waves with the parity of $SD(S^R)$ as an initial value of length $SD(S^R)+2$. Let p and k be a combinational path which starts at a primary input or an FF and ends at an FF or a primary output in S^R and the signal line just before the ending point of p, respectively. Let $l(k)$ and Q be the parity of the minimum sequential depth among sequential paths in the output cone of k and a sequential path from k to a primary output, respectively. Let q and $PSD(Q,p,q)$ be a combinational path which starts at a primary input or an FF and ends at an FF or a primary output on Q and the parity of the sequential depth of a sequential path between p and q on Q, respectively.*

Case $l(k)=0$:

- *If all the gates on p are AND gates, T can be a robust test of the rising PDF on p.*
- *For path p on which all the gates are OR gates, if there exists Q such that all the gate on path q that satisfies $PSD(Q,p,q)=0$ are OR gates and all the gate on path q that satisfies $PSD(Q,p,q)=1$ are AND gates, T is a robust test of the falling PDF on p. If such Q does not exist, T is a functional sensitizable test[9] of the rising PDF on p.*
- *If both AND and OR gates are mixed on p, T is a functional sensitizable test of the rising PDF on p.*

Case $l(k)=1$:

- *If all the gates on p are OR gates, T is a robust test of the falling PDF on p.*
- *If all the gates on p are all AND gates, if there exists Q such that all the gate on path q that satisfies $PSD(Q,p,q)=0$ are AND gates and all the gate on path q that satisfies $PSD(Q,p,q)=1$ are OR gates, T is a robust test of the rising PDF on p. If such Q does not exist, T is a functional sensitizable test of the falling PDF on p.*
- *If both AND and OR gates are mixed on p, T is a functional sensitizable test of the falling PDF on p.*

Proof: From Definition 4, there exists a sequential path that has the minimum sequential depth from k to a primary output to propagate the error of the rising PDF on P such that $l(k)=0$. For the falling PDF on p such that $l(k)=1$, there exists a sequential path that has the minimum sequential depth from k to a primary output to propagate its error.

We show that the error of the rising PDF on p appeared at k can be propagated to a primary output if there exist the sequential path Q that satisfies the condition described in the theorem from k to a primary output. If p is fault free, from Corollary 1, the ending point of Q can have a rising or a falling transition. If there exists a rising PDF on p, no rising transition cannot be appeared at k or the value of k is fixed to 0. From the condition of Q and the property of controlling value of a gate, 1 is fixed to inputs of OR gates on q in Q such that $PSD(Q,p,q)=1$ and 0 is fixed to inputs of AND gates on q in Q such that $PSD(Q,p,q)=0$.

If all the gates on p are AND gates, therefore the rising PDF on p can be robustly activated and the error can be propagated to the primary output. On the other hand, if both AND and OR gates are mixed on p, the rising PDF on p can be functionally sensitized.

Similarly, if all the gates on p are OR gates, the falling PDF on p can be robustly activated and the error can be propagated to the primary output. On the other hand, if both AND and OR gates are mixed on p, the falling PDF on p can be functionally sensitized. □

Testability of S^F can be shown by the similar way as Theorem 3. The circuit S^F can be obtained from S^R by replacing AND into OR and OR into AND. If we apply inverted T to S^F, the relation of controlling and non controlling values for each gate is reversed. Therefore, it is obvious that the PDF with a transition direction on a path p' in S^F can be tested under the same condition of the corresponding PDF with its opposite direction on p in S^R.

For example, since the maximum sequential depth of the rising sensitized circuit shown in Fig. 6 is two, we apply square waves of length 4 with initial value 0 to all the primary inputs. There exists a reconvergence structure con-

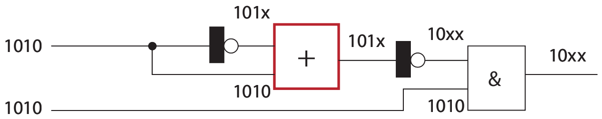

Figure 6. An example of a rising sensitized circuit.

Table 1. The ratio of robust testable paths for acyclic-transformed ISCAS'89 benchmarks.

Circuit	#paths	Robust ratio (%)
s27	28	60.71
s208.1	142	38.03
s298	231	25.97
s344	355	10.14
s349	365	9.86
s382	400	49.25
s386	207	43.96
s400	448	48.21
s420.1	474	35.86
s444	535	50.47
s510	369	37.67
s526	410	33.66
s526n	408	33.82
s641	1744	32.05
s713	21812	25.77
s820	492	36.38
s832	506	35.38
s838.1	1714	34.66
s1423	44726	14.66
s1488	962	53.85
s1494	976	53.59
s5378	13542	60.57
s9234	244854	36.04
s9234.1	244854	36.14
s38417	1391579	3.05
s38584	1080723	20.95
s38584.1	1080723	20.95

cerned with the OR gate in the figure. By inverting the output of the FF just before the OR, the direction of the transition can be inverted and then all the inputs can have inphase square waves.

For sequential circuits with cycles, by partitioning introduced in Section 2, our proposed method can be applied to each partitioned circuit.

In the next section, for acyclic-transformed ISCAS'89 benchmark circuits, we will show how many PDFs can be tested robustly for its rising (falling) sensitized circuits.

4. Experimental Results

In this section, we evaluate the number of robustly testable paths by the proposed method for acyclic structure. In this experiments, we transformed ISCAS'89 benchmark circuits into acyclic ones and the transformed circuits were used. The total number of paths and the ratio of the number of robustly testable paths to the total number of paths are reported in Table 1. From the table, we can see that the ratios were from 3% to 60% and their average was around 17% unfortunately. Improvement of the ratio will be our future work.

5. Conclusions

In this paper, we proposed a method of testing for path delay faults for application-specific interconnects in FPGAs. Concretely, we define inphase structure as a class of sequential circuits and a method for testing for an FPGA with an inphase sequential circuit as a configuration. We showed that two test configurations of its AND and OR networks are sufficient to test all the path delay faults in the circuit robustly and the test length for each test configuration is linear to the sequential depth of the circuit. For testing an FPGA with an acyclic sequential circuit as a configuration, we define the transformation of rising (falling) sensitization and the transformed circuits are used as test configurations. We also showed that two test configurations are sufficient to test all the path delay faults in the circuit under robust or functional sensitization condition and the test length for each test configuration is linear to the sequential depth of the circuit. Through the experiments, we evaluated the ratio of the number of robust testable paths to the total number of paths. To increase the ratio is one of the our future work. The proposed method can be applied to sequential circuits with cycles using the method of configuration partitioning.

References

[1] M. Renovell, P. Faure, J. M. Protal, J. Figueras and Y. Zorian: "Testing the interconnect of ram-based FPGAs," *IEEE Design and Test of Computers*, pp. 45–50, 1998.

[2] I. Harris and R. Tessier: "Diagnosis of interconnect faults in cluster-based FPGA architectures," in *Proc. of Design Automation Conference*, pp. 49–54, 2000.

[3] I. Harris, P. Menon and R. Tessier: "BIST-based delay path testing in FPGA architectures," in *Proc. of Int'l Test Conf.*, pp. 932–938, 2001.

[4] A. Krasnierski: "Testing FPGA delay-faults in the system environment is very different from ordinary delay-fault testing," in *Proc. of Int'l On-Line Test Workshop*, pp. 37–40, 2001.

[5] Xilix: The Xilinx EasyPath Solution, http://www.xilinx.com/products/silicon_solutions/fpgas/easypath/, 2006.

[6] M. B. Tahoori: "Application-dependent diagnosis of FPGAs," in *Proc. of Int'l Test Conf.*, pp. 645–654, 2004.

[7] M. B. Tahoori and S. Mitra: "Interconnect delay testing of design on programmable logic devices," in *Proc. of Int'l Test Conf.*, pp. 635–644, 2004.

[8] S. Ohtake, S. Miwa and H. Fujiwara: "A method of test generation for path delay faults in balanced sequential circuits," in *Proc. of VLSI Test Symp.*, pp. 321–327, 2002.

[9] A. Krstić and K.-T. T. Cheng: *Delay fault testing for VLSI circuits*, Kluwer Academic Publishers, 1998.