Czech Technical University in Prague

WORKSHOP 2011

Project carried out within the framework of the CTU Student grant competition in 2010

This research has been supported by SGS grant No SGS10/118/OHK3/1T/18.

Design of Dependable Systems Based on Programmable Circuits

Kubátová Hana, Balcárek Jiří, Chloupek Martin

kubatova@fit.cvut.cz, balcaji2@fit.cvut.cz, martin.chloupek@fit.cvut.cz
Faculty of Information Technology, Dept. of Digital Design

Abstract: Project will be focused on dependable system design implemented in programmable gate arrays (FPGA) and their testing. Increasing of dependability of circuits will be achieved by effective usage of dynamic reconfiguration. For improving the testability of the circuits methods for decomposition of the circuit are used. The compressed test generation methodology based on overlapping of the test vectors will be improved using their implicit representation to maximize the test effectiveness. We are also focused on circuit's area with goal to minimize the hardware overhead. Developed architecture and methods are tested on experimental designs in FPGA.

During the last decade, systems realized by programmable hardware like Field Programmable Gate Arrays (FPGAs) are more and more popular and widely used in more and more applications due to their capability of implementing complex circuitry within a very short development time, together with the potential for easy reconfiguration or for other actual changes of the implemented circuit. Different types of faults (Single Event Upset, Single Event Latchup, Delay fault etc.) can arise in these FPGAs and if we would use these FPGAs in critical application, the design implemented inside have to be dependable at the requested level according the concrete application.

Some redundancy has to be incorporated into the circuit design to improve dependability parameters. We have performed experiments with online testing to obtain Concurrent Error Detection (CED) [1]. Basic dependability criteria are: fault security (FS), self-testing (ST), totally self-checking (TSC), availability, reliability, testability, mean time between failures (MTBF), mean time to repair (MTTR), etc.

To determine whether the circuit satisfies the TSC property, the possible faults are classified and separated into four classes, A, B, C and D [2] according to their impact on the tested circuit design in the FPGA. The measurement of the behavior of a real designed system or a benchmark circuit (Design Under Test - DUT) affected by fault is required for dependability parameters calculations. The international safety standard IEC-61508 highly recommends fault injection techniques in all steps of the development process in order to analyze the reaction of the system in a faulty environment. Simulated fault injection enables an early dependability assessment that reduces the risk of late discovery of safety related design pitfalls and enables the analysis of fault tolerance mechanisms at each design refinement step [3].

Faults can be injected into design or FPGA chip directly. Many papers show experimental results in the area of the dependable design, but there are problems with the comparison of obtained results. Many authors use standard ISCAS benchmarks that are provided in a bench

format. The bench format is close to a netlist representation, but the benchmarks are available also in VHDL or Verilog formats. There are no rules about processing these benchmarks. Published results are difficult not only to compare but also to verify. Important processing options influencing the concrete system design are listed below [4]:

- Synthesis tool
- Synthesis options and optimization
- Target architecture
- Place and route options

Formal dependability models are necessary to calculate the level of reliability of a modeled system. Markov chains are well-known and widely used for reliability calculations, but there are more models available. Markov chains were originally proposed by the Russian mathematician Markov in 1907. Over the many decades since, they have been extensively applied to problems in social science, economics and finance, Computer science, computer-generated music, and other fields. The availability parameter of the modeled system can be calculated as the steady-state distribution of the probabilities of Markov chain. This distribution can be obtained as the solution of the system of linear equations. Stochastic Petri nets (SPNs) can be created as easy as Markov chains and allow the same calculations. Mathematical properties, that are available for Petri nets, can be used to analyze SPNs. Moreover, SPNs are not only a dependability model but they are able to represent the structure of the design. Generalized Stochastic Petri nets (GSPNs) have two different classes of transitions: immediate transitions and timed transitions. Once enabled, immediate transitions fire in zero time. Timed transitions fire after a random, exponentially distributed enabling time as in the case of SPNs. Our results with more details about models relations and comparison can be found in [5].

Test application time is also vital part of manufacturing process and has a significant influence on final cost of the chip. That is the reason, why compression techniques of test patterns have been developed. There are several methods used for a test patterns compression. Many of these compression methods are based on using some encoding, such as statistical codes, run-length codes, and Golomb codes, others are based on XOR networks, hybrid patterns, folding counters, EDT (Embedded Deterministic Test) and reuse of scan chains. Other approaches are based on test pattern compaction and overlapping.

Further research has been focused on compression techniques based on test patterns overlapping because they proved to have a high compression rate and also the decompression hardware is very simple. However, compression rate depends on the number of don't care bits in pre-generated test patterns which should be overlapped. In our new approach we try to eliminate dependence on pre-generated test patterns and reach best possible overlap (compression rate). The main idea is not to overlap test patterns pre-generated by an ATPG (Automatic Test Patterns Generation), but to generate most suitable test patterns in the process, to reach the (locally) best overlap. Each fault has its set of test patterns by which it is detected. If we were able to pick the right pattern for each fault in the right order, we could have reached the best possible compression of the test patterns. Because explicit computation and storing of all these test patterns is inefficient (and mostly even infeasible), we were forced to find another, more efficient way of test pattern set representation. We have researched possibilities of implicit representations of test patterns. We have found that we can take advantage of principles of SAT-based ATPGs and efficiently represent all test patterns for one fault by one SAT problem instance in a CNF (Conjunctive Normal Form). Than the best test pattern (locally) to be overlapped is find by applying constraints. Our results with more

details about test patterns compression method based on implicit representation of the test patterns in CNF and detail comparison with competitive approaches can be found in [6].

One method of testing FPGA based designs is using scan-chain. Scan-based DUT testing is widely used in today's application-specific integrated circuits (ASICs) and it can be adopted in FPGA, however it imposes more area overhead. Applying test vector to the DUT is much time consuming, it involves shifting in and shifting out the test vector, so the number of scan cells in the scan chain is major issue here. We have developed a new scan-based test vector de-compressor architecture that utilizes technique called "broadcasting". We split single long scan-chain into many short parallel chains, thus we efficiently reduce the test application time. Shifting-in the same data into many parallel chains brings up a constraint, that some test vectors are not applicable; we say these vectors have dependences. Our decompression architecture contains a certain mechanism, that overcomes these constraints, but it increases the test application time. Our goal is to minimize the count of test vectors with dependences in the test vector set, so we have developed a new technique for splitting single long scan-chain into shorter parallel chains. Our results with more details about the decompression architecture and the scan-chain partition technique can be found in [7].

This research has been partially supported by GA102/09/1668 and MSM6840770014.

References:

[1] PRADHAN, D.K.: *Fault-Tolerant Computer System Design* Prentice-Hall, Inc., New Jersey, 1996.

[2] KUBALÍK, P. - FIŠER, P - KUBÁTOVÁ, H.: *Fault Tolerant System Design Method Based on Self-Checking Circuits* Proc. 12th International On-Line Testing Symposium 2006 (IOLTS'06), 2006.

[3] PEREZ, J. - AZKARATE-ASKASUA - M. AND PEREZ, A.: *Codesign and Simulated Fault In-jection of Safety-Critical Embedded Systems Using SystemC* In Proceedings of the IEEE European Dependable Computing Conference (EDCC), 2010, p. 221-229

[4] BORECKÝ, J. - KOHLÍK, M. – KUBÁTOVÁ, H.: *How to Measure Dependability Parameters of Programmable Digital Circuits - A Survey.* In 6th Doctoral Workshop on Mathematical and Engineering Methods in Computer Science. Brno: NOVPRESS, 2010, p. 28-35

[5] KOHLÍK, M.: *Dependability models based on Petri nets and Markov chains.* Počítačové architektury & diagnostika. Soláň, Universita Tomáše Bati ve Zlíně, 2009, p. 95-103

[6] BALCÁREK, J. - FIŠER, P. - SCHMIDT, J.: *Test Patterns Compression Technique Based on a Dedicated SAT-Based.* Proceedings of the 13th Euromicro Conference on Digital System Design. Los Alamitos: IEEE Computer Society Press, 2010, p. 805-808

[7] CHLOUPEK, M.: *Simple Heuristic for Broadcast Scan Chain Configuration*. In Počítačové architektury & diagnostika. Brno: Vysoké učení technické v Brně, 2010, p. 39-44