KHARKOV NATIONAL UNIVERSITY OF RADIOELECTRONICS

Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2012)

Technically Co-Sponsored by

Kharkov, Ukraine, September 14 – 17, 2012

IEEE EAST-WEST DESIGN AND TEST SYMPOSIUM 2012 COMMITTEE

General Chairs
V. Hahanov – Ukraine
Y. Zorian – USA

General Vice-Chairs
R. Ubar - Estonia
P. Prinetto - Italy

Program Chairs
S. Shoukourian – Armenia
D. Speranskiy – Russia

Program Vice-Chairs
Z. Navabi – Iran
M. Renovell – France

Publicity Chair's
G. Markosyan - Armenia
S. Mosin - Russia

Public Relation Chair
V. Djigan - Russia

Program Committee
E. J. Aas - Norway
J. Abraham - USA
M. Adamski - Poland
A.E.Mohamed Mohamed - Egypt
A . Barkalov - Poland
R. Bazylevych - Ukraine
A. Chaterjee - USA
V. Djigan - Russia
A. Drozd - Ukraine
E. Evdokimov - Ukraine
E. Gramatova - Slovakia
A. Ivanov - Canada
M. Karavay - Russia
V. Kharchenko - Ukraine
K. Kuchukjan - Armenia
W. Kuzmicz - Poland
A. Matrosova - Russia
V. Melikyan - Armenia
L. Miklea - Romania
O. Novak - Czech Republic
Z. Peng - Sweden
A. Petrenko - Ukraine
J. Raik - Estonia
A. Romankevich - Ukraine
A. Ryjov - Russia
R. Seinauskas - Lithuania
S. Sharshunov - Russia
A. Singh - USA
J. Skobtsov - Ukraine
V. Tverdokhlebov - Russia
V. Vardanian - Armenia
V. Yarmolik - Byelorussia

Steering Committee
M. Bondarenko - Ukraine
V. Hahanov - Ukraine
R. Ubar - Estonia
Y. Zorian - USA

Organizing Committee
S. Chumachenko - Ukraine
E. Litvinova – Ukraine
V. Kharchenko – Ukraine

EWDTS 2012 CONTACT INFORMATION

Prof. Vladimir Hahanov
Design Automation Department
Kharkov National University of Radio Electronics,
14 Lenin ave,
Kharkov, 61166, Ukraine.

Tel.: +380 (57)-702-13-26
E-mail: hahanov@kture.kharkov.ua
Web: www.ewdtest.com/conf/

10th IEEE EAST-WEST DESIGN & TEST SYMPOSIUM (EWDTS 2012)

Kharkov, Ukraine, September 14-17, 2012

The main target of the **IEEE East-West Design & Test Symposium** (EWDTS) is to exchange experiences between scientists and technologies of Eastern and Western Europe, as well as North America and other parts of the world, in the field of design, design automation and test of electronic circuits and systems. The symposium is typically held in countries around the Black Sea, the Baltic Sea and Central Asia region. We cordially invite you to participate and submit your contributions to EWDTS'12 which covers (but is not limited to) the following topics:

- Analog, Mixed-Signal and RF Test
- Analysis and Optimization
- ATPG and High-Level Test
- Built-In Self Test
- Debug and Diagnosis
- Defect/Fault Tolerance and Reliability
- Design for Testability
- Design Verification and Validation
- EDA Tools for Design and Test
- Embedded Software Performance
- Failure Analysis, Defect and Fault
- FPGA Test
- HDL in test and test languages
- High-level Synthesis
- High-Performance Networks and Systems on a Chip
- Low-power Design
- Memory and Processor Test
- Modeling & Fault Simulation
- Network-on-Chip Design & Test
- Modeling and Synthesis of Embedded Systems
- Object-Oriented System Specification and Design
- On-Line Testing
- Power Issues in Design & Test
- Real Time Embedded Systems
- Reliability of Digital Systems
- Scan-Based Techniques
- Self-Repair and Reconfigurable Architectures
- Signal and Information Processing in Radio and Communication Engineering
- System Level Modeling, Simulation & Test Generation
- System-in-Package and 3D Design & Test
- Using UML for Embedded System Specification
- CAD and EDA Tools, Methods and Algorithms
- Design and Process Engineering
- Logic, Schematic and System Synthesis
- Place and Route
- Thermal, Timing and Electrostatic Analysis of SoCs and Systems on Board
- Wireless and RFID Systems Synthesis
- Digital Satellite Television

The Symposium will take place in Kharkov, Ukraine, one of the biggest scientific and industrial center. Venue of EWDTS 2012 is Kharkov National University of Radioelectronics was founded 81 years ago. It was one of the best University of Soviet Union during 60th - 90th in the field of Radioelectronics. Today University is the leader among technical universities in Ukraine.

The symposium is organized by Kharkov National University of Radio Electronics and Science Academy of Applied Radio Electronics http://anpre.org.ua/ in cooperation with Tallinn University of Technology. It is technically co-sponsored by the IEEE Computer Society Test Technology Technical Council (TTTC) and financially supported by Trades Committee of Kharkov National University of Radioelectronics and Trades Committee of Students, Aldec, Synopsys, Kaspersky Lab, DataArt Lab, Tallinn Technical University.

CONTENTS

Models for Embedded Repairing Logic Blocks

Hahanov V.I., Litvinova E.I., Frolov A., Tiecoura Yves (Cote d'Ivoire)
Computer Engineering Faculty, Kharkov National University of Radioelectronics, Kharkov, Ukraine, hahanov@kture.kharkov.ua

Abstract

The models of combinational circuits, focused on solving practical problems of embedded repairing components of the logic units are proposed. The logical circuit is complemented by operational and control automata for modeling digital devices, which increases processing time and hardware costs for creating a wrap of addressable elements. The structures can also be used for hardware modeling functionalities of digital projects by using PLD, which allows improving the performance of software model verification. The proposed solution of embedded gate repair for combinational circuits makes it possible to comprehensively solve the problem of autonomous repairing digital systems on chips due to the time and hardware project redundancy [1-12].

1. Introduction

If executing with failures within a single chip, the strategy of increasing the reliability of the system based on reconfiguring the functionality can be used, but it has two disadvantages: 1) considerable time of this operation is incompatible with the functioning of critical systems; 2) significant complication of reconfiguring related to the existence of faulty areas on the chips.

The second way of improving the reliability of digital systems-on-chips can be the addition of redundant elements to the basic functionality, which are designed to compensate for negative effects of faulty components by their readdressing in the testing loop and fault detection. It requires smaller time cycle in comparison with reconfiguration of functional areas of the chip. Implementation of the second way, proposed in [9-12], lies in adding (modifying) three components (spare functional elements and ports for reading/writing data, advanced multiplexers for switching the first two ones) in SoC functionality, which enables the replacement of faulty elements in the cycle BIST/BISR. The reliability of a discrete area on the chip that implements the complete functionality of a digital system is characterized by the criterion with an exponential distribution depending on the intensity λ of uncorrelated failures, which are constant over time: $R_A(t) = e^{-\lambda t}$. To consider the reliability of the original structure as a whole, the following estimation can be used [9]: $R_O(t) = R_A(t)^{A_O}, A_O = M \times A_{FU}$. At that the system is unable to compensate for the failures, resulting it in inoperable state. When considering the complexity of the hardware (Area) of the system with redundancy A_O, containing M main components and N spares, to repair functional modules with faults the following conversions of expressions for determining the reliability evaluation R_{BISR} can be used:

$$R_{BISR} = R_{SW} \times R_{MN};$$
$$R_{MN} = \sum_{i=M}^{N} \binom{N}{i} R_{FU}^{i} \times [1 - R_{FU}]^{N-i};$$
$$R_{SW} = R_O^{(A_{SW})/A_O} = R_O^{(A_{SW})/(M \times A_{FU})};$$
$$R_{FU} = R_O^{(A_{FU})/A_O} = R_O^{1/(M)} = R_O^{-M};$$
$$R_{BISR} = R_O^{(A_{SW})/(M \times A_{FU})} \times$$
$$\times \sum_{i=M}^{N} \binom{N}{i} R_O^{i-M} \times [1 - R_O^{-M}]^{N-i}.$$

Here A_{FU}, A_{SW} – a number of discrete elements (hardware complexity) in functional and spare component of the structure A_O; R_{MN}, R_{SW}, R_{FU} – reliability indexes of the system with redundant components, auxiliary hardware for switching, and functional component respectively. The best value of reliability index of the system is fixed when the parameter of switching (addressed) hardware A_{SW} tends to zero. Thus, the reliability of the system with redundancy will always be higher than the reliability of the structure without spare components, if the volume of the switching hardware is less than the functional one. The exception is the case when M=1, N=0, and reliability of the systems with redundancy and without it are equal.

The ability to create circuits with built-in self-testing and self-repairing is considered in [10-12]. Now

these problems are well solved for regular structures, such as memory. At the same time for the circuits that have combinational organization of logic elements, this problem is much more difficult. In paper [10] it is proposed to divide the circuit on separate blocks and add one spare for every three basic blocks or two spares for 6 basic blocks, Fig. 1. If a fault is detected a spare can replace one of the base functional modules. To manage the repairing it is necessary the input and output switching circuits, decoder of block selection, and two bits of memory to store the system state.

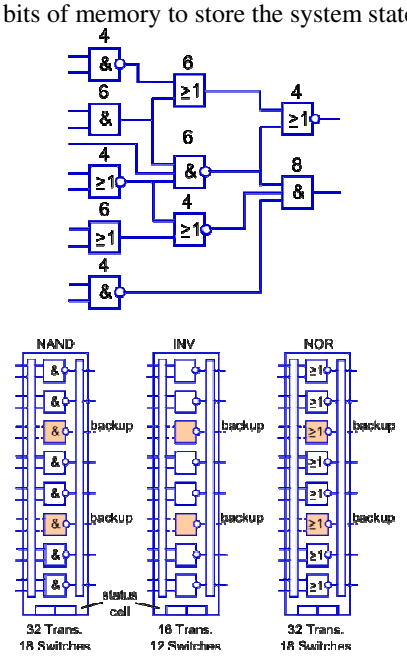

Fig. 1. Reconfigurable logic and mapping circuits

It have found that the higher dimension of system blocks the smaller the percentage of redundant hardware that is necessary for the implementation of self-testing and repairing circuits. For 32-bit ALU, a number of redundant elements is equal to 38%, while for the AND-NOT gate – 200%. To reduce redundancy and simplify the self-test circuit spares are controlled by the same circuit. At that the model of errors is based on the assumption that only one block of a subsystem can be faulty at the current time. Self-test and self-repair circuits are initiated by turn-on power, and they perform test functions when configuring FPGA. In [9] a strategy for creating self-testable logic circuits without description of realization is proposed. It is shown the abstract architecture of reconfigurable block with unlimited number of sub-blocks and without taking into account the redundancy for creating it.

Disadvantages and comments: 1. There is mentioned FPGA, but the results don't focused on real chip architecture with configurable CLB, which are basic elements for implementation of combinational circuits. Gates, standard elements are considered but not blocks CLB. 2. Replacing a single logic element in CLB is hard to implement, because the function on the blocks is presented in tabular form. The minimum block for the construction of self-testable circuits can be only CLB. 3. The proposed self-testing method does not apply to specific parts of the adder and devices based on them. Adders are realized on through carry lines, directed along the CLB columns of FPGA. Impossible to exclude a block from the adder without violating the integrity of the device. Therefore, redundancy can be realized only for whole component or it is necessary to change FPGA circuit, but at microelectronics level. 4. Creating self-test circuit at the level of logic elements leads to a high redundancy of hardware. The topical problem is generating register or system-level descriptions of digital devices to obtain a uniform circuit, which can be transformed in self-repairable structure. A method for self-repairing based on the use of regular FPGA structure and exclusion CLB column or row with faulty element is proposed, Fig. 2.

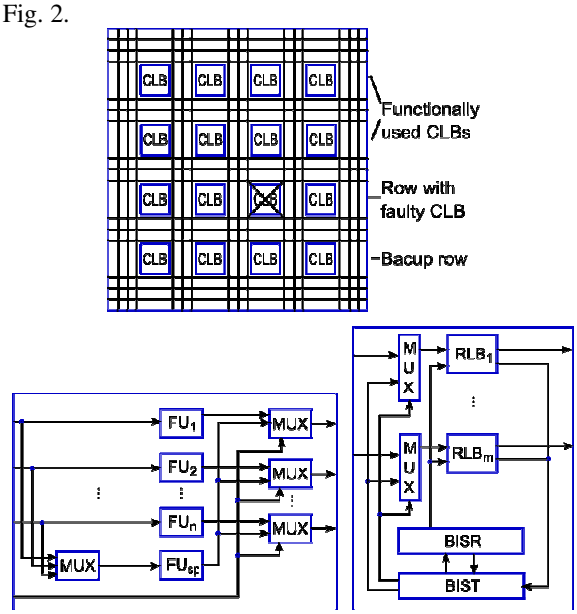

Fig. 2. Built-in-self repair for RLBs

The authors argue that the method needs to be improved. In fact, it seems more realistic and technologically advanced, because it is permissible to add redundant CLB columns, introducing an additional shadow configuration.

The concept of addressed performing logical operations, implemented in the memory elements LUT of programmable logic devices (PLD), provides the potential possibility to create on-chip address space only for embedded repair of all components involved

in functionality [1-8]. The tendency of increasing the memory leads to the possibility of embedded repair of faulty cells by providing additional spare logic cells. The problem of autonomous removal of logic element faults (self-repair) is associated with the lack of addresses. But it can be solved if flexible connections between logic elements will be formed by using structure description program located in memory and connected the logical components in the circuit. In addition to the structure of element interaction the memory must contain a procedure for their processing. In the event of a fault in one of the addressable logical elements, the BISR system will repair it by readdressing to a known-good spare.

The aim of this paper is to improve the quality and reliability of digital systems-on-chips by creating an infrastructure for embedded testing, diagnosis, optimization and repairing through hardware redundancy and reduction of the speed of functional operations [1-8].

Problems and references: 1) Analysis of modern repair technologies for SoC components [4-11]. 2) Development of mathematical model for embedded repairing of logic elements, involved in combinational structure of the functionality, implemented in SoC [4-8]. 3) Creation of operational and control automata to emulate the functionality of combinational circuit in a chip PLD [1-3].

2. Model of combinational structure

The few works devoted to repair of logic circuits [6, 9-12] describe the two ideas. The first one is the reconfiguration of the structure of logical components off-line, which provides the possibility of replacing each of the faulty primitives. The second one creates the conditions for replacement of faulty components through the use of spare components and extension of multiplexers for readdressing failed primitives. An example for the considering the theory and practice of addressing primitives of combinational circuits for built-in repairing functional failures of the logic elements by using the simplest circuit structure is presented below (Fig. 3).

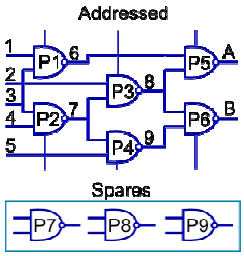

Fig. 3. Addressable logic elements

It contains six logic elements of the same type, which can be represented in the address space by the following list (two-dimensional array):

S =										
	No =	1	2	3	4	5	6	7	8	9
	P =	1	1	1	1	1	1	1	1	1
	L_1 =	1	3	2	7	6	8	X	X	X
	L_2 =	3	4	7	5	8	9	X	X	X
	L_3 =	6	7	8	9	A	B	Y	Y	Y

No =	1	2	3	4	5	6	7	8	9	A	B
M =	1	1	1	1	1	0	0	1	1	1	1

F(1) =

X_1	X_2	Y
0	0	1
0	1	1
1	0	1
1	1	0

Each column corresponds to a logical element of the circuit, and the primitives with the numbers 7, 8, 9 are the spares, which are used to replace any three of the six elements when diagnosing in them any functional failures. In the line P the primitive types are indicated, lower – the number of input and output variables; the simulation vector M contains the result of analysis for the input word 11111 of the circuit structure (see Fig. 3). The process model for generating the output values of the circuit, depending on the concatenated state of inputs, which form the address of the cell of output state, can be represented in abstract form or hardware-focused on the use of the state vector M, as follows:

$$\begin{bmatrix} Y_6 = P_1(X_1 * X_3); \\ Y_7 = P_2(X_3 * X_4); \\ Y_8 = P_3(X_2 * X_7); \\ Y_9 = P_4(X_7 * X_5); \\ Y_A = P_5(X_6 * X_8); \\ Y_B = P_6(X_8 * X_9). \end{bmatrix} \rightarrow \begin{bmatrix} M_6 = P_1(M_1 * M_3); \\ M_7 = P_2(M_3 * M_4); \\ M_8 = P_3(M_2 * M_7); \\ M_9 = P_4(M_7 * M_5); \\ M_A = P_5(M_6 * M_8); \\ M_B = P_6(M_8 * M_9). \end{bmatrix}$$

Here M is a state vector for circuit lines; P_i is logic function AND-NOT that has two inputs and realized as the memory element LUT. Since all of six primitive elements implement a single logic function AND-NOT then the previous expression can be simplified by replacing all the structural elements by the function primitive F with subsequent use for concatenation of two-dimensional array of lines (L) between the inputs and outputs of logic gates:

$$\begin{bmatrix} M_6 = F(M_1 * M_3); \\ M_7 = F(M_3 * M_4); \\ M_8 = F(M_2 * M_7); \\ M_9 = F(M_7 * M_5); \\ M_A = F(M_6 * M_8); \\ M_B = F(M_8 * M_9). \end{bmatrix} \rightarrow \begin{bmatrix} M_6 = F[M(L_{11}) * M(L_{12})]; \\ M_7 = F[M(L_{21}) * M(L_{22})]; \\ M_8 = F[M(L_{31}) * M(L_{32})]; \\ M_9 = F[M(L_{41}) * M(L_{42})]; \\ M_A = F[M(L_{51}) * M(L_{52})]; \\ M_B = F[M(L_{61}) * M(L_{62})]. \end{bmatrix}$$

Thus, a structure for implementing a process model of the circuit, which has two-input functional primitives, can be represented as follows:

$$M(L_{is_p}) = F[M(L_{ij}) * M(L_{ir})] = F[M(L)] .$$

Given the fact that all the calculations in the circuit are focused on the structural elements which have identifier of logical operation, the previous formula can be transformed to the view:

$$M(L_{is_p}) = P_i[M(L_{ij}) * M(L_{ir})] = P[M(L)].$$

In general, the structure of the model for the functionality, focused on PLD implementation, has five components:

$$S =< P, F, M, L, T >;\ P = (P_1, P_2, ..., P_i, ..., P_n);$$
$$F = (F_1, F_2, ..., F_j, ..., F_m);\ M = (M_1, M_2, ..., M_r, ..., M_k);$$
$$L = [L_{pq}]; p = \overline{1,n};\ q = \overline{1,s_p};\ T = [T_{te}]; t = \overline{1,\eta};\ e = \overline{1,\mu};$$
$$M(L) = P[M(L)].$$

Here are: 1) the primitives of circuit structure P, defined by the identifiers of functionality type (number or code of instruction); 2) the types of functional elements F – a set of memory elements LUT, from which the primitives are implemented, as well as redundant components for functionality repair; 3) the simulation vector M (binary), which defines the status of all lines (input, internal, output); 4) the matrix of the equipotential lines L for combining n logic elements in the structure; 5) the matrix of input test patterns T. The processing circuit on a chip is reduced to the determination of address compiled from binary bits of the simulation vector, which determines a logical function. Each primitive has processing loop that contains three procedures: 1) the address reading the numbers of input variables from the corresponding column of the matrix L to form the address of the state of the input variable for the simulation vector: $A = L_{ij}$, $i = \overline{1,n}$; $j = \overline{1,s_p - 1}$; 2) Generating the address (binary code) to compute the logic function by concatenating the corresponding states of the input variables in the modeling vector $A = M(L_{ij}) * M(L_{ir})$; 3) Saving the result of the logical functions (the status of the output) to the corresponding bit of the modeling vector $M(L_{is_p}) = F[M(L_{ij}) * M(L_{ir})]$.

3. Operation device for simulating the combination structure

The processing of all the primitives of the circuit in this case is strongly consistent that result in considerable slowdown of procedure for forming state of output variables. However, the decrease in performance can be considered as payment for embedded and autonomous repairing the functionality of digital structure, which is one of the stages of SoC infrastructure IP, shown in Fig. 4. Combinational circuit is operational device, where there are operational and control automata. Replaceable components of operational automaton are the types of primitives – functional elements, Fig. 4.

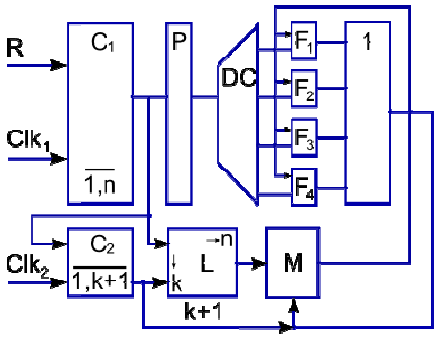

Fig. 4. Operational structure of combinational circuit

Operational device for implementation of element-addressable combinational circuits contains the following components: counter for processing the current primitive C1; memory for storing the types of primitives, corresponding to the structural elements P; counter for reading the numbers of input and output variables of the current primitive C2; decoder of primitive types DC; memory for storing the modeling vector M; matrix memory for storing the numbers of inputs-outputs of the structural primitives L; memory line, realizing the functional primitives F; register for generating the input address word for the processed primitive RG; logic element Or for switching results of processing of functional primitives.

Flow-chart for simulating the structure of combinational circuit is shown in Fig. 5.

1. Initialization (formation) of all components of circuit structure (numbers and types of elements, connection lines for inputs and outputs of logic gates):

$$P = (P_1, P_2, ..., P_1, ..., P_n);\ F = (F_1, F_2, ..., F_j, ..., F_m);$$
$$L = [L_{pq}]; p = \overline{1,n};\ q = \overline{1,s_p}.$$

2. Initialization of parameter of the processed primitive and the number of input pattern $i = 0, t = 0$ for its simulating in the binary alphabet $M_r = \{0,1\}$.

3. Incrementing the index of the primitive, the numbers of test and initialization of the input test pattern: $i = i + 1, t = t + 1, M(X) = T_t(X), |T_t(X)| = \eta$.

4. Concatenation (#) of word bits for generating of the input stimulus $\#_{j=1}^{k} M(L_{ij})$ for logic element P_i; performing the procedure for determining the state of its output and subsequent writing into the correspondent coordinate of the modeling vector: $M(L_{k+1}): M(L_{k+1}) = P_i[\#_{j=1}^{k} M(L_{ij})]$.

5. Iteration of items 3 and 4 in order to obtain the states of the outputs of all the logic elements to satisfy the condition: $i = n$.

6. Iteration of items 2–4 for simulating all input test patterns to the equality: $t = \eta$, where η is length of the test.

7. The end of the simulation of digital device.

Verification of models for embedded repairing are implemented on 10 test case studies (26 – 1024 logic elements), which confirmed the fact that the cost characteristics of the hardware redundancy (x1,5–x2,5) and the reliability of combinational circuits is not worse than in the reference publications [9-12].

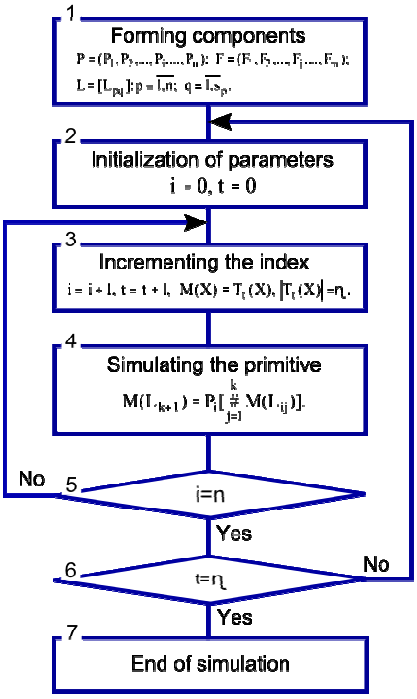

Fig. 5. Flow-chart for simulating the structure of combinational circuit

4. Conclusion

The scientific novelty lies in the following. The proposed operational and control automata for simulating digital combinational circuits are focused on two real-world problems: 1) Embedded repairing components of combinational logic circuits by increasing the processing time of digital device and hardware redundancy for creating the infrastructure for simulating addressable elements. 2) Hardware simulating the functionalities of digital projects by using PLD, which makes it possible to substantially improve the performance of verification of software models. The practical value of the research lies in obtaining the solution of the problem of embedded repairing logic elements of the combinational circuits that makes it possible to solve the problem of autonomous repair of digital systems-on-chips due to the time and hardware redundancy of the project, which has a positive impact on the reliability of the system and its life cycle.

Areas for further research are associated with the following problems: 1) Models for switching failed primitives. 2) Parallelization of computations by levels of combinational circuit elements. 3) Replace the types and primitives of the structure. 4) Creating the operational device or infrastructure for simulating sequential elements and structures. 5) Simulating circuits consisting of functional complex primitives. 6) Development of infrastructure for embedded testing, diagnosis and repairing elements of combinational and sequential devices. 7) Testing and repairing infrastructure IT for combinational circuit is solution of the problem «watching the Watchmen». 8) Creating the analytical models for estimating the efficiency of the infrastructure IP for embedded repairing the combinational digital systems with different complexity levels of the entities and structures.

10. References

[1] *Hahanov V. I., Litvinova E. I., Chumachenko S. V., Guz O.A.* Logic associative computer. Electronnoe modelirovanie. № 1. 2011. P. 73-90.

[2] *Hahanov V., Wajeb Gharibi, Litvinova E., Chumachenko S.* Information analysis infrastructure for diagnosis. Information. An international interdisciplinary journal. 2011. Japan. Vol.14. No 7. P. 2419-2433.

[3] *Hahanov V.I.* and others. Infrastructure of intellectual property for SoC simulation and diagnosis service. Springer, Germany, 2011. P. 289-330.

[4] *Chung J., Park J., Abraham J. A.* Built-In Repair Analyzer With Optimal Repair Rate for Word-Oriented Memories // IEEE Transaction on Very Large Scale Integration (VLSI) Systems, 2012. Iss. 99. P. 1–11.

[5] *Mincent Lee.* A Memory Built-In Self-Repair Scheme Based on Configurable Spares / Lee Mincent, Denq Li-Ming, Wu Cheng-Wen // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. Vol. 30, Iss. 6.– 2011. P. 919 – 929.

[6] *Koal T.* A comprehensive scheme for logic self repair / T. Koal, D. Scheit, H.T. Vierhaus // Conference Proc. on Signal Processing Algorithms, Architectures, Arrangements, and Applications. 2009. P. 13 – 18.

[7] *Rab M.T.* Improving Memory Repair by Selective Row Partitioning / M.T. Rab, A.A. Bawa, N.A. Touba // IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems. 2009. P. 211 – 219.

[8] *Zorian Y.* Test and reliability concerns for 3D-ICs / Y. Zorian // IEEE 16th Internationa On-Line Testing Symposium (IOLTS). 2010. P. 219 – 219.

[9] *Koal T., Vierhaus H.T.* Optimal Spare Utilization for Reliability and Mean Lifetime Improvement of Logic Built-In Self-Repair / T. Koal, H.T. Vierhaus // 14th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). 2011. P. 219 – 224.

[10] *Koal T., Scheit D., Vierhaus H.T.* A Concept for Logic Self Repair / T. Koal, D. Scheit, H.T. Vierhaus // 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools, DSD'2009. P. 621-624.

[11] *Koal T., Vierhaus H.T.* Basic Architecture for Logic Self Repair / T. Koal, H.T. Vierhaus // 14th IEEE International On-Line Testing Symposium, IOLTS '08. 2008. p. 177-178.

[12] *Koal T., Vierhaus H.T.* A software-based self-test and hardware reconfiguration solution for VLIW processors / T. Koal, H.T. Vierhaus // IEEE Design and Diagnostics of Electronic Circuits and Systems. Los Alamitos, CA, USA.– 2010. P. 40–43.

Camera-ready was prepared in Kharkov National University of Radio Electronics
Lenin Ave, 14, KNURE, Kharkov, 61166, Ukraine

Approved for publication: 27.08.2012. Format 60×841/8.
Relative printer's sheets: 49. Circulation: 200 copies.
Published by SPD FL Stepanov V.V.
Ukraine, 61168, Kharkov, Ak. Pavlova st., 311

Матеріали симпозіуму «Схід-Захід Проектування та Діагностування – 2012»
Макет підготовлено у Харківському національному університеті радіоелектроніки
Редактори: Володимир Хаханов, Світлана Чумаченко, Євгенія Литвинова
Пр. Леніна, 14, ХНУРЕ, Харків, 61166, Україна

Підписано до публікації: 27.08.2012. Формат 60×84$^{1}/_{8}$.
Умов. друк. Арк. 49. Тираж: 200 прим.
Видано: СПД ФЛ Степанов В.В.
Вул. Ак. Павлова, 311, Харків, 61168, Україна