KHARKOV NATIONAL UNIVERSITY OF RADIOELECTRONICS

Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2013)

Technically Co-Sponsored by

Rostov-on-Don, Russia, September 27 – 30, 2013

IEEE EAST-WEST DESIGN AND TEST SYMPOSIUM 2013 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

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

V. Hahanov – Ukraine
R. Ubar – Estonia
Y. Zorian – USA

Organizing Committee

S. Chumachenko – Ukraine
M. Karyakin – Russia
S. Krutchinskiy – Russia
E. Litvinova – Ukraine
N. Prokopenko – Russia
B. Steinberg – Russia

EWDTS 2013 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/

11th IEEE EAST-WEST DESIGN & TEST SYMPOSIUM (EWDTS 2013)

Rostov-on-Don, Russia, September 27-30, 2013

The main target of the **East-West Design & Test Symposium** (EWDTS) is to exchange experiences between the scientists and technologies of the 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 systems. The symposium aims at attracting scientists especially from countries around the Black Sea, the Baltic states and Central Asia. We cordially invite you to participate and submit your contribution(s) to EWDTS'13 which covers (but is not limited to) the following topics:

- Analog, Mixed-Signal and RF Test
- Analysis and Optimization
- ATPG and High-Level TPG
- 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 Test
- Power Issues in Testing
- 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
- Using UML for Embedded System Specification

CAD Session:

- 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 Systems Synthesis
- Digital Satellite Television

The Symposium will take place in Rostov-on-Don, Russia, one of the biggest scientific and industrial center. Venue of EWDTS 2013 is Don State Technical University – the biggest dynamically developing centre of science, education and culture.

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 Don State Technical University and Tallinn University of Technology. It is technically co-sponsored by the IEEE Computer Society Test Technology Technical Council (TTTC) and financially supported by Aldec, Synopsys, DataArt Lab, Tallinn Technical University, Aldec Inc.

.

CONTENTS

Quantum Modeling and Repairing Digital Systems

Baghdadi Ammar Awni Abbas (Baghdad University)[1], Hahanov V.I.[2],
Palanichamy Manikandan[3], Litvinova E.I.[2], Dementiev S.[2]
[1]Baghdad University, Iraq
[2]Computer Engineering Faculty, Kharkov National University of Radioelectronics,
Kharkov, Ukraine, hahanov@kture.kharkov.ua
[3]Norwegian University of Science and Technology
manikandan.palanichamy@iet.ntnu.no

Abstract

The results of studies concerning the models and methods of quantum diagnosis of digital systems, qubit fault simulation and analysis of fault-free behavior, as well as repair of faulty primitives, are presented.

1. Qubit fault models

A fault is defined as each individual discrepancy of a product to specification, but fault model should never lead out the product beyond the functionality limits [1,2,8]. Therefore fault (fault model – failure) is time fixed part of the functionality that is tied to a physical component. The constant line fault is fixed transition 0-0 at two adjacent cycles. It makes no sense to consider it as a further extension to other cycles, because according to the automaton model they are all described by means of two adjacent time frames. By extending this two-frame concept to automaton variables we can introduce the full set of fault transitions: 00, 01, 10, 11. Indeed, if we consider the automatic variables, for instance for the register, it is necessary to generate test patterns for verifying the above transitions. Based on the concept of the fault, it follows that the total number of states of functionality also forms a complete set of faults, with the only difference being that the specific fault is always a complement to test signal that detects a fault [2]:

$$\begin{cases} F \oplus T \oplus L = \varnothing, \\ T \oplus F = L; \\ T \oplus L = F; \\ F \oplus L = T. \end{cases}$$

Here it is shown that the interaction of the functionality test and faults is always convoluted to empty or null space, which allows determining any fault by non-empty interaction between the test and functionality. But because the test is a derivative of the function, then it follows that a fault (its model) is a derivative of the function or its spatio-temporal frame. In order to such triad interaction was possible it is necessary to lead the models of three components to a single type or view (matrix or table, Boolean equations, structure). One of the variant of replacing the traditional Roth symbols [8], aimed on searching active paths in the circuit, can be vector representation of spatio-temporal frames of a model, test, and fault in Boolean discrete measurement.

Qubit data model carries out "transformation" of the information quantum from the set-theoretic symbol in the logical bit-vector with unitary encoding primitives. This practically means the replacement of set-theoretic operations by vector-logic ones, which has a positive impact on performance, but there is a subjective loss in the visualization of information for a human who is not able to read the letters (texts), presented in the form of binary vectors. Thus, any closed multi-valued set-theoretic alphabet is written by a set of binary vectors, where the transformation is determined by the relation: $n = \log_2(B = 2^n)$, where B is the Boolean (power set), n – the number of bits or primitives (power of the universe), which form the power set. Otherwise, qubit is a vector-logical equivalent of any closed set-theoretic alphabet. This means that qubit is a vector with the number of bits (at least two) equal to the number of primitive symbols, which is capable to represent the power set of states by means of its binary digits. Single-bit qubit does not make sense, since it has no conformance in the set-theoretic space.

Example of a four-digit qubit is presented below. Suppose we have a two-stroke alphabet for description of automatic variables, which is the power set on the universe of the four primitives:
B*(Y)={Q=(1000), E=(0100), H=(0010), J=(0001), O={Q,H}=(1010), I={E,J}=(0101), A={Q,E}=(1100), B={H,J}=(0011), S={Q,J}=(1001, P={E,H}=(0110),

C={E,H,J}=(1110), F={Q,H,J}=(1011), L={Q,E,J}=(1101), V={Q,E,H}=(1110), Y={Q,E,H,J} =(1111), U=(0000)}.

Operations on the symbols of the set-theoretic alphabet are reduced to logical instructions and, or, not, xor, which form a functionally complete basis according to Post's theorem [3]. For example, the following logical transformations of some set-theoretic operations are presented below:

$$Q \cup E = 1000 \vee 0100 = 1100 = A;$$
$$S \cap V = 1001 \wedge 1110 = 1000 = Q;$$
$$\tilde{B} = \overline{0011} = 1100 = A;$$
$$F \Delta P = 1011 \oplus 0110 = 1101 = Y;$$
$$H \Delta J = 0010 \oplus 0001 = 0011 = B;$$
$$F \Delta Y = 1011 \oplus 1111 = 0100 = Y;$$
$$F \Delta F = 1011 \oplus 1011 = 0000 = U(\varnothing);$$

The quantum interpretation of the power set of four primitives (binary positional codes: 00, 01, 10, 11) is presented below:

00	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
01	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
10	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
11	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

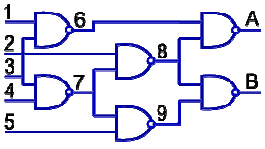
It describes 16 different functions of two variables. At the same time the last table can be represented in the form of codes (vectors) of multi-valued alphabet symbols, which are easy to use to address the problems of synthesis and analysis of Boolean functions:

Q	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
E	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
H	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
J	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	∅	J	H	B	E	I	P	C	Q	S	O	F	A	L	V	Y

Thus, a qubit composed of four bits can be used for describing 16 states of the spatio-temporal frame of functionalities; at that it uses vector logic operations to address the problems of synthesis and analysis.

2. Quantum modeling digital systems

For example, to describe a digital circuit shown in Fig. 1 we must have a structure of interconnected elements, and cubic coverage of logic AND-NOT element.



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

Fig. 1. Fragment of digital circuit

The coverage may be modified taking into account unitary encoding input vectors. As a consequence for any coverage of the functional single input primitive there always exist only two cubes. They show not only all the solutions, but also the inverse of the signals at the outputs, which is interesting from the point of activation of all logical paths in the circuit structure. For example, in the following qubit gate coverage in order to change the state of the output it is necessary to create a pair of consecutive conditions on inputs, where the first three vectors (addresses) have to be in the first cycle, fourth vector that is forming by two input variables - in the second one:

00	1	=	Q	1	=	V	1	=	1110	1	→	1	1	1	0
01	1		E	1		J	0		0001	0					
10	1		H	1											
11	0		J	0											

To simulate the fault-free behavior it is sufficient to have only one cube (zero or unit), because the second one is always complement of the first. Consequently, if we are focused, for example, on unit cube that forms 1 on the output we can remove the bit of primitive output state, which reduces the dimension of the cube or primitive model up to the number of element addresses, where bit with address number I identifies the capability of an address, consisting of the values of the input variables, to form unit state of primitive output.

Qubit or quantum coverage of single output primitive is two cubes the dimension of which is equal to the power of two of the number of input variables, where unit coordinate value defines the use of bit address in forming the corresponding (0,1) state of primitive output. Quantum models of primitives require the creation of a new theory of modeling, direct and inverse implication, test synthesis, fault simulation, fault detection. Here, the basic procedures are associated with the manipulation by addresses, implicitly represented in the coordinates of two cubes of quantum (qubit) coverage hereafter named Q-coverage.

00	0	=	Q	0	=	F	0	=	1101	0
01	0		E	0		H	1		0010	1
10	1		H	1						
11	0		J	0						

Model for analyzing digital system based on the use of qubit data structures and information quanta can be described by four components:

$$F = < L, M, X, Q >,$$
$$L = (L_1, L_2, ..., L_j, ..., L_n);$$
$$M = (M_1, M_2, ..., M_j, ..., M_n);$$

$X = (X_{x+1}, X_{x+2}, ..., X_{x+i}, ..., X_{x+n})$;
$Q = (Q_{x+1}, Q_{x+2}, ..., Q_{x+i}, ..., Q_{x+n})$.

The vector of identifiers for circuit potential lines of digital system, state modeling vector for all circuit lines, ordered set of input variable vectors for each circuit primitive associated to the output numbers, a set of vectors – quantum primitives coverages, strictly associated to the output numbers and the input variables of primitives, n is the number of lines in the circuit, x – the number of input variables are represented here.

An example of qubit model of digital device $F =< L, M, X, Q >$ (Fig. 1) is presented below – two variants of circuit description tables for analyzing fault-free behavior (fault free simulation):

L	1	2	3	4	5	6	7	8	9	A	B
Q	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	0	0	0	0	0	0
X						13	34	27	75	68	89
M	1	1	1	1	1	0	0	1	1	1	0

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
Q	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	0	0	0	0	0	0

The method of quantum fault-free simulation is reduced to the definition of address generated by binary states of input variables for each primitive of digital circuit. If the variables create non-binary address, in this case, there is possibility of forming non-binary output state of the primitive, which is defined in the ternary alphabet by symbol X. The output states are formed by consistent modeling, based on simple iterations or Seidel iterations [3,4]. In the second case a preprocessor procedure for ranking lines and circuit primitives is necessary, which considerably reduces the number of passes on the circuit primitives to achieve convergence when the equality of the states of all circuit lines in two adjacent iterations is fixed. In addition, ranking of primitives on the levels of forming outputs allows significantly improving the performance of simulation due to parallel processing functional elements of one level. For example, for the circuit shown in Fig. 1, we can handle concurrently the elements with the numbers 6, 7, and then – 8, 9 and beyond - A, B. In the first case, when simple iterations are used ranking is not required, but cost for simplicity of simulation algorithm is significantly greater number of iterative passes through the circuit primitives to achieve the convergence criterion.

For the synthesis of quasi-optimal data structures of a combinational device it is necessary to use the following rules:

1) To simulate by using Seidel's method, the ranked circuit of a digital device on the structural depth must have the same type of primitives as possible in each level (layer) of operation.

2) It is desirable to have the same number of primitives at each level that means - synthesis of digital device has to be focused on creation a rectangular or matrix like structure of similar logical elements.

3) Implementation of the combinational primitives provides for using addressable memory elements existing in programmable logic devices (FPGA, CPLD), widely used for prototyping.

4) Providing spare primitives for each level of the combinational device for online repairing - one spare element for each type of component that is used in the level.

5) Hardware cost for implementing high-performance combinational device should be determined by the sum of all the primitives associated to the levels of combinational device, extended by set of spares, one for each layer (assuming the existence of the same primitives in each layer):

$$Q = \sum_{i=\overline{1,n}}^{j=\overline{1,m}} P_{ij} + n.$$

6) Implementation of the combinational device based on minimizing hardware cost is determined by the sum of all types of primitives, which are invariant to the levels of combinational device and extended by set of spares, one for each type:

$$Q = \sum_{i=1}^{m} P_i + m.$$

7) Processing the matrix of the combinational elements by using the processor line of primitives, the number of which is equal to the power of maximum level or layer in rectangular structure, which provides the possibility for parallel processing all primitives in each element level in order to improve the performance of the combinational prototype, implemented in the PLD.

3. Repair of combinational units

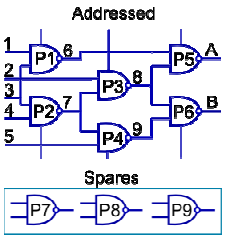
The few works devoted to repair of logic circuits [5-7] describe the two ideas. The first one provides for reconfiguring the structure of logical components off-line, which provides the possibility of replacing each of the faulty primitives. The second one creates the conditions for replacement of faulty components through the use of spare components and extension of multiplexers for readdressing failed primitives.

Qubit data structures is modified by way of adding a row of primitive types

$$F = < L, M, X, P, Q >, P = (P_1, P_2, ..., P_i, ..., P_m),$$

used when synthesizing the digital system, if necessary to repair the system during operation by using spare primitives, which like basic elements are implemented on the basis of memory.

Fig. 2 illustrates an example of a circuit structure, composed from addressable and three spare components. Data structures corresponding to the circuit with three additional elements are presented below:



L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	.	1	2	3	4	5	6
Q	1	1	1	1	1	1	1	1	1	.	.
	1	1	1	1	1	1	1	1	1	.	.
	1	1	1	1	1	1	1	1	1	.	.
	0	0	0	0	0	0	0	0	0	.	.

Fig. 2. An example of a circuit structure of addressable and spare elements

The table uses the numbers of structural primitives, which gives the opportunity to replace any failed element by spare by changing the address number in the line of primitives P. Repair elements in this table begin with the number 7.

The following table shows the line of logical elements, as well as the addresses of types for these primitives, marked with numbers:

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	.	1	1	1	1	1	1
Q	1	1	1	1	.	.	.	.	.	.	.
	1	1	1	1	.	.	.	.	.	.	.
	1	1	1	1	.	.	.	.	.	.	.
	0	0	0	0	.	.	.	.	.	.	.

This data structure is focused on software simulation, and spare primitives begin with the number 2. If it is possible to reprogram the logic in the memory element with the same number of input variables, then the procedure should be performed after fixing the faulty element, where it is known - which element in the structure refused and the type of it. This repair procedure is focused to PLD-implementation of digital systems. If the qubit models of circuits have no spare primitives, the corresponding tables will be as follows:

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	.	1	1	1	1	1	1
Q	1	.	.	.	.	.	.	.	.	.	.
	1	.	.	.	.	.	.	.	.	.	.
	1	.	.	.	.	.	.	.	.	.	.
	0	.	.	.	.	.	.	.	.	.	.

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	.	1	2	3	4	5	6
Q	1	1	1	1	1	1	.	.	.	.	.
	1	1	1	1	1	1	.	.	.	.	.
	1	1	1	1	1	1	.	.	.	.	.
	0	0	0	0	0	0	.	.	.	.	.

Thus, quantum data structures are focused on compact description of digital product functionality by means of qubit vectors, speedup simulation procedures by using addresses of output states of primitives, and repairing the logic gates due to their implementation in PLD memory elements.

Processing circuit on a chip is reduced to determine the address, compiled from binary bits of the simulation vector, which determines a logical function. Each primitive has processing loop that contains three procedures:

1) Address reading the numbers of input variables from the corresponding column of the matrix X to form the address of the states of the input variable for the simulation vector: $A = X_{ij}, i = \overline{1,n}; j = \overline{1,s_p - 1}$;

2) Generating the address (binary code) to compute the logic function by concatenating the corresponding states of the input variables in the simulation vector $A = M(X_{ij}) * M(X_{ir})$;

3) Saving the result of executing the logical function (the status of the output) to the corresponding bit of the simulation vector $M(X_{is_p}) = P[M(X_{ij}) * M(X_{ir})]$.

The processing of all circuit primitives in this case is strongly consistent that result in considerable slowdown of procedure for forming states 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 functioning SoC infrastructure IP, shown in Fig. 5.

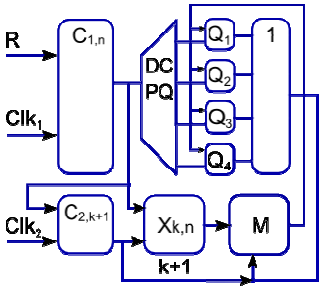


Fig. 5. Operational structure of combinational circuit

Combinational circuit becomes operating unit, where there are operating and control automata. Replaceable components in the operating automaton are primitive types - functional elements or structural primitives.

Operational unit for implementation of element-addressable combinational circuits contains the following components: counter for processing the

current primitive C_1; 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 C_2; decoder of primitive types DC; memory for storing the simulation vector M; matrix memory for storing the numbers of inputs-outputs of the structural primitives X; memory line, realizing the functional primitives P(Q); register for generating the input address word for the processed primitive RG; logic element Or for switching results of processing functional primitives.

Flow-chart of control algorithm for simulating the structure of combinational circuit is shown in Fig. 3 and includes the following items:

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

$$P=(P_1,P_2,...,P_i,...,P_n);\ Q=(Q_1,Q_2,...,Q_j,...,Q_m);$$
$$X=[X_{pq}];p=\overline{1,n};q=\overline{1,s_p}.$$

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 the input stimulus $\#_{j=1}^{k} M(X_{ij})$ for logic element P_i (of the type Q_i); performing the procedure for determining the state of its output and subsequent writing into the correspondent coordinate of the simulation vector $M(X_{k+1})$:

$$M(X_{k+1})=\{P_i,Q_i\}[\#_{j=1}^{k} M(X_{ij})].$$

5. Iteration of items 3 and 4 in order to obtain the states of the outputs for 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.

The operating and control automata are represented here; they are focused on consistent processing combinational primitives that allow realizing the readdressing procedure for primitives in case of failure of one of them. It is easy to develop similar automata for parallel processing layers of ranged circuit primitives that maximum approaches the performance of the device to realization in PLD chips.

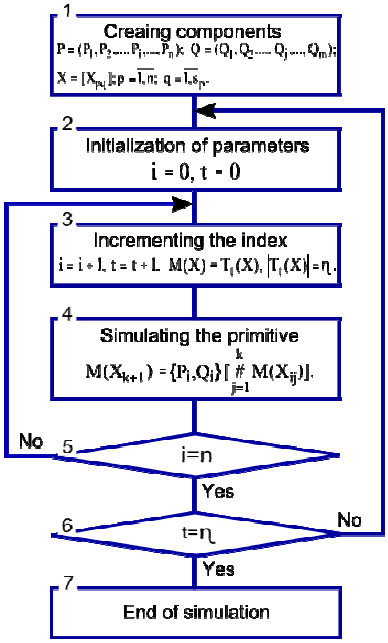


Fig. 3. Flow-chart of control algorithm for simulation

4. Conclusion

A quantum method for fault-free simulation of digital devices with online repairing components of digital systems is presented; it is characterized by significantly better performance due to the address processing functional primitives, defined by qubit coverage.

Case-studies of repairing faulty modules based on the addressable logical primitives are considered.

Directions for future research:

1. Using the or-operation for the analysis of fault detection table in order to detect multiple faults.

2. Research and application of Hasse processor to detect single and/or multiple faults by fault detection table and reduction of the procedure to searching for optimal coverage.

3. Modeling fault-free behavior of the sequential circuits based on qubit data structures of digital devices.

4. Synthesis of qubit models of functional primitives and digital systems, based on the use of quantum primitives, as well as the decomposition (analysis) of functionality quanta on qubit primitive models.

5. Simulation of transition faults of automaton variables by modifying the deductive algorithm.

6. Development of encoders and decoders for DSP-processors based on the use of qubit data structures, which allow increasing the performance of coding tools and compactness of data representation.

5. References

[1] *Hahanov V.I.* Digital System-on-Chip Design and Test. Kharkov: Novoye Slovo, 2009. 484 p.

[2] *Hahanov V.I.* Technical diagnosis of digital and microprocessor structures. K.: ICIO. 1995. 242 p.

[3] Gorbatov V.A. Fundamentals of Discrete Mathematics. M. : Vysshaya Shkola. 1986. 311 p.

[4] Hahanov V.I. Technical diagnosis of digital and microprocessor structures. K.: ICIO. 1995. 242 p.

[5] Hahanov V.I., Litvinova E.I., Hahanova I.V., Murad Ali Abbas. Infrastructure for repairing logic PLD-circuits // Radioelectronics & Informatics. 2012. No 2. P. 54-57.

[6] Hahanov V., Litvinova E., Gharibi W., Murad Ali Abbas. Qubit models for SoC Synthesis Parallel and cloud computing. USA. 2012. Vol.1. Iss 1. P. 16-20.

[7] Hahanov V.I., Litvinova E.I., Chumachenko S.V., Baghdadi Ammar Awni Abbas, Eshetie Abebech, Mandefro. Qubit Model for solving the coverage problem // Proc. of IEEE East-West Design and Test Symposium. IEEE. USA. Kharkov. 14-17 September 2012. P.142 - 144.

[8] Chzen G., Menning E., Metz G. Fault diagnosis of digital computing systems. M.: Mir. 1972. 230 p.

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

Approved for publication: 11.09.2013. Format 60×841/8.
Relative printer's sheets: 52. Circulation: 200 copies.
Published by SPD FL Stepanov V.V.
Ukraine, 61168, Kharkov, Ak. Pavlova st., 311

Матеріали симпозіуму «Схід-Захід Проектування та Діагностування – 2013»
Макет підготовлено у Харківському національному університеті радіоелектроніки
Редактори: Володимир Хаханов, Світлана Чумаченко, Євгенія Литвинова
Пр. Леніна, 14, ХНУРЕ, Харків, 61166, Україна

Підписано до публікації: 11.09.2013. Формат 60×84$^{1}/_{8}$.
Умов. друк. Арк. 52. Тираж: 200 прим.
Видано: СПД ФЛ Степанов В.В.
Вул. Ак. Павлова, 311, Харків, 61168, Україна