

EMBEDDED SOC F-IP DIAGNOSIS BY USING ALGEBRAIC LOGICAL METHOD

VLADIMIR HAHANOV, VLADIMIR OBRIZAN, EUGENIA LITVINOVA, KA LOK MAN
KHARKOV NATIONAL UNIVERSITY OF RADIOELECTRONICS, UKRAINE

ABSTRACT: SoC Functional Intellectual Property Infrastructure that differ by minimal set of the embedded diagnosis processes in real time and enables to realize such services like functional testing on basis of generable input patterns and analysis of output reactions; SoC diagnosis with given resolution of fault location; fault simulation to carry out the first two procedures on basis of the fault detection table are offered. Structural-algebraic method of embedded fault diagnosis of SoC functional blocks is proposed as well. The method uses preliminary analysis of the fault detection table to reduction its size and subsequent DNF construction computation, which forms all solutions of SoC functional diagnosis in real time.

Keywords: Diagnosis, System-on-Chip, Electronic System Level, Transaction Level Modelling, Infrastructure Intellectual Property, Functional Intellectual Property, Testbench, Fault

I-IP INFRASTRUCTURE

Computational and hardware complexity of modern digital systems on a chip (SoC) is characterized by millions of equivalent gates and requires making and implementation of new high-level design technologies: Electronic System Level (ESL) Design, Transaction Level Modelling (TLM) and embedded service – Infrastructure Intellectual Property (I-IP). It means that search for high-performance methods and facilities [1-11] reduces all researchers to necessity to rise of an abstraction level of Functional Intellectual Property (F-IP) models, which are created and embedded into a chip. Market appeal of the implementation of a digital system to FPGA is determined by the followings: application of relatively cheap chips instead of the universal processors, low power consumption, small overall size, qualitative and reliable realization of the main functions due to on-chip I-IP-infrastructure that is urgent in the century of mobile computers.

The research aim is development of algebra-logical method of SoC Functional Intellectual Property Infrastructure that is intended for the diagnosis of SoC components in real time. The problems: 1) State of the market of SoC Infrastructure Intellectual Property technologies; 2) Algebra-logical (AL) method of Infrastructure Intellectual Property on basis of the cover matrix; 3) Application of the AL-method to diagnosis of SoC components; 4) Practical results.

Modern design technologies of digital systems on chips propose along with creation of functional blocks F-IP development of service modules I-IP, which are oriented on complex solving of the project quality problem and yield increasing in manufacturing that is determined by implementation of the following services into a chip [8]: 1) Observation for state of input and output lines in functioning, verification and testing of standard blocks on basis of utilization of the boundary scan standard IEEE 1500 [10]; 2) Testing of functional modules by means of input of the fault detection

patterns from different test generators, which are oriented on verification of faults or fault-free state; 3) Fault diagnosis by means of analysis of information obtained on the testing stage and utilization of special methods of embedded fault lookup on basis of the standard IEEE 1500 [10]; 4) Repair of functional modules and memory after fixation of negative testing result, fault location and its type on diagnosis stage; 5) Measurement of the general characteristics and parameters of a device operation on basis of on-chip facilities, which enable to make time and volt-ampere measurements; 6) Reliability and fault tolerance of a device operation in working that is obtained by diversification of functional blocks, redundancy of them and repair of SoC in real time.

In Fig. 1 it is represented the reduced structure [8], oriented on solving of the following problem: 1) testing of the functionalities on basis of generable input patterns (Automated Test Pattern Generator) and analysis of output reactions; 2) Fault simulation [5] to ensure the diagnosis and repair on basis of the fault detection table; 3) Fault diagnosis with given resolution by means of utilization of the IEEE 1500.

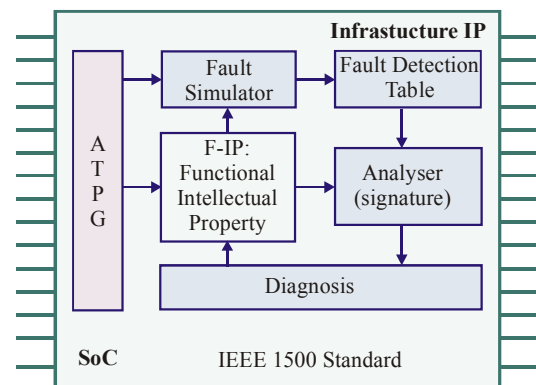


Fig. 1. Infrastructure Intellectual Property of SoC DSP

1. Automated Test Pattern Generator for verification of functionalities and single faults consists of a set of input patterns generators, which provide creation of the

complete tests [1,2] for the F-IP's. Generator module analyses the structural-functional model of a tested block and assigns a subset of such synthesizers, which provide given fault cover quality (F^c) and functional modes (P^c):

$$F^c \left(\bigcup_{i=1}^{n_{\min}} T_i \right) \geq F_{\min}^c; P^c \left(\bigcup_{i=1}^{n_{\min}} T_i \right) \geq P_{\min}^c, \quad (1)$$

$$T = \{T_1^{PR}, T_2^{SA}, T_3^{SP}, T_4^{AD}, T_5^{BS}, T_6^{ME}, T_7^{DF}, T_8^{RC}\}.$$

Generalized structure of Testbench synthesis [1] is represented in Fig. 2 and includes HDL-code generator that is designed for functional testing and verification on the stage of project development.

A number of test generators on the SoC development stage can be considerably greater than a subset of ones that embedded into a chip later. So, on the simulation and verification stage the analysis of covering features of every test generator is performed to search for the minimal aggregate configuration of them that is satisfied expression (1). It is important to say that within the next 5 years the test synthesis ideology for SoC will borrow the best traditions of ESL-, TLM-design [7,11].

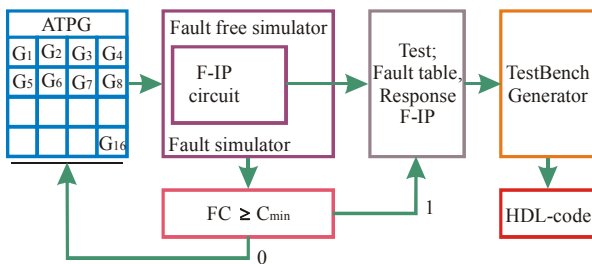


Fig. 2. Structure of the Testbench synthesis process for F-IP

It means: 1) Use of the Testbench libraries of the lead companies in the world for testing and verification of standardize functionalities, which are designated as F-IP. 2) Application of I-IP standard solution for on-chip testing of SoC components. 3) Creation of own test libraries for new-developing functionalities. 4) Adoption of new technology of the test synthesis for a digital system, based on the discrete mapping [11] of covering of functionalities and faults of the initial specification by means of minimal Testbench set from a test library. 5) Application of the on-chip testability facilities, such as IEEE boundary scan and six I-IP components, to increase of the technological effectiveness of test synthesis procedures.

ALGEBRA-LOGICAL METHOD OF THE FAULT DIAGNOSIS

The general role is assigned to the boundary scan technology [10] that is implemented into a chip now has to simplify solution of practically all problems of SoC Functional Intellectual Property Infrastructure. Diagnosis procedure, based on the boundary scan register uses information from the fault detection table that is a fault set, which are covered by test patterns. Using result information of a diagnostic experiment that is represented in an experimental validation vector form $V = (V_1, V_2, \dots, V_i, \dots, V_n)$, as well as the fault detection table F [5], the diagnosis procedure is carried out in compliance with the expression, written in the product

of disjunctions form for all faults [6], which can give an experimental reaction in the form of V that determined by unit and zero values:

$$F = \bigwedge_{\forall V_i=1} \left(\bigvee_{j=1, \dots, m} F_j \right). \quad (5)$$

The conjunctive normal form, derived from the fault detection table, is transformed to the disjunctive normal form by means of equivalent transformations (conjunction, minimization and absorption) [6]. Therefore we have the Boolean function, where terms are the logical product, which represent full solution set in the fault combination form (they give the experimental validation vector V at SoC outputs or its component):

$$F = \bigwedge_{\forall V_i=1} \left(\bigvee_{j=1, \dots, m} F_j \right) = \left[\begin{matrix} a_{\forall ab=b} \\ a_{\forall a=a} \end{matrix} \right] = \bigvee_{i=1}^{2^m} \left(\bigwedge_{j=1}^m k_j F_j \right), k_j = \{0,1\}. \quad (6)$$

Represented procedure in general case diagnoses some fault subset that later needs a refinement by application of additional flexing of internal points by means of the boundary scan register. An example of defect finding is considered on basis of the following fault detection table (columns are faults, rows are test patterns) that is product of the deductive fault analysis and the experimental validation vector:

$T_i \backslash F_j$	F_1	F_2	F_3	F_4	F_5	F_6	V
T_1	1			1			1
T_2		1			1		1
T_3			1	1	1		0
T_4	1		1				1
T_5		1			1	1	1

A number of units in the experimental validation vector V forms quantity of CNF disjunctive terms (6). Every term is line-by-line writing of faults (by logic operation OR), which influence on functional outputs. Table representation in the analytical form (conjunctive normal form) makes possible to reduce volume of information for defect finding essentially. Subsequent transformation of CNF to DNF on basis of the Boolean algebra identities enables to reduce the Boolean function that is illustrated by the following result:

$$F = (F_1 \vee F_4)(F_2 \vee F_5)(F_3 \vee F_4 \vee F_5)(F_1 \vee F_3)(F_2 \vee F_5 \vee F_6) = \\ = F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_1 F_3 F_5 \vee F_3 F_4 F_5 \vee F_1 F_2 F_4 \vee F_1 F_4 F_5 \vee \\ \vee F_1 F_5 = F_1 F_5 \vee F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5 \vee F_1 F_2 F_4. \quad (7)$$

To decrease of a number of computing at carrying out of conjunction in the first line (7) the initial notation can be simplified in accordance with the Boolean algebra laws: $(F_2 \vee F_5) \wedge (F_2 \vee F_5 \vee F_6) = (F_2 \vee F_5)$.

The derived result

$$F = F_1 F_5 \vee F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5 \vee F_1 F_2 F_4$$

represents all possible solutions (fault covering of the fault detection table rows of SoC functionality on condition that the experimental validation vector has all unit coordinates $V = (11111)$. Taking into account the actual value of the experimental validation vector $V=(11011)$, it is carried out the simulation of function F by substitution of zero fault values, which are verified theoretically, but they give zero coordinate in the

vector V. Such fault are: $\bar{F} = F_3 \vee F_4 \vee F_5$. Final result is determined by the next function:

$$F = F_1 F_5 \vee F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5 \vee F_1 F_2 F_4 \Big|_{F_3 \vee F_4 \vee F_5 = 0} = 0.$$

Whatever combination is DNF conjunctive term, presented in solution

$$F = F_1 F_5 \vee F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5 \vee F_1 F_2 F_4,$$

covers all rows of the fault detection table according with definition, so addition of any zero row transforms function F to zero without fail. So, correct solution that corresponds to the experimental validation vector must take into account zero coordinates of the vector V. Subject to the stated above it is necessary to eliminate the term $(F_3 \vee F_4 \vee F_5)$ from expression (7) on CNF forming stage

$$F = (F_1 \vee F_4)(F_2 \vee F_5)(F_1 \vee F_3)(F_2 \vee F_5 \vee F_6) = F_1 F_2 \vee F_1 F_5 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5. \quad (8)$$

The result represents all possible solutions, which make a device reaction, determined given experimental validation vector:

$$F = F_1 F_2 \vee F_1 F_5 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5.$$

Additional simulation of last Boolean function gives final solution in the form of two faults combination:

$$F = F_1 F_2 \vee F_1 F_5 \vee F_2 F_3 F_4 \vee F_3 F_4 F_5 \Big|_{F_3 \vee F_4 \vee F_5 = 0} = F_1 F_2.$$

Algebra-logical method can be formally considered by an example of the following fault detection table M₁ and it can be represented by five algorithm items.

$$M_1 = \begin{array}{c|cccccc|c} T_i \backslash F_j & F_1 & F_2 & F_3 & F_4 & F_5 & F_6 & V \\ \hline T_1 & 1 & & & 1 & & & 1 \\ T_2 & & 1 & & & 1 & & 1 \\ T_3 & & & 1 & 1 & & 1 & 1 \\ T_4 & 1 & & 1 & & & & 1 \\ T_5 & & 1 & & & 1 & 1 & 0 \end{array}$$

1. Detection of all rows, which correspond to zero of the experimental validation vector for nulling of all 1-coordinates of found rows. In this case it is the row T₅.
2. Detection of all columns, which have zero values of rows coordinates with zero state of the vector V. Nulling of unit values of found columns. In this case it is F₂, F₅, F₆.
3. Removal the rows and the columns, which have only zero coordinate values (found in items 1 and 2), from the fault detection table.

$$M_1 = \begin{array}{c|cccccc|c} T_i \backslash F_j & F_1 & F_2 & F_3 & F_4 & F_5 & F_6 & V \\ \hline T_1 & 1 & & & 1 & & & 1 \\ T_2 & & 0 & & & 0 & & 1 \\ T_3 & & & 1 & 1 & & 0 & 1 \\ T_4 & 1 & & 1 & & & & 1 \\ T_5 & & 0 & & & 0 & 0 & 0 \end{array} = \begin{array}{c|cccc|c} T_i \backslash F_j & F_1 & F_3 & F_4 & V \\ \hline T_1 & 1 & 1 & 1 & 1 \\ T_3 & & 1 & 1 & 1 \\ T_4 & 1 & 1 & & 1 \end{array}$$

4. Making CNF by unit values of the experimental validation vector:

$$F = (F_1 \vee F_4) \wedge (F_3 \vee F_4) \wedge (F_1 \vee F_3) = F_1 F_3 \vee F_1 F_4 \vee F_3 F_4.$$

5. Transformation CNF to DNF with subsequent minimization of the function. It brings to gaining of sought-for result in the fault combination form:

$$F = F_1 F_3 \vee F_1 F_4 \vee F_3 F_4.$$

The proposed algorithm is oriented on preliminary analysis of the fault detection table to decrease its size and amount of subsequent computing related to DNF making that forms all solutions of SoC functionalities diagnosis. Further refinement of a diagnosis is possible by application of the multiprobe on basis of the boundary scan register [10].

ALGEBRA-LOGICAL DIAGNOSIS MODEL

The fault diagnosis of F-IP functional blocks based on discrepancy between model and experimental reactions on a test forms unit coordinates of the experimental validation vector $V(T) = (V_1, V_2, \dots, V_i, \dots, V_n)$ for every input pattern. Communication between the vector V and the fault detection table ($T = [T_{tr}], t = \bar{1}, p; r = \bar{1}, q + n$ of dimension $p \times n$, p is a number of test-vectors, n is a number of stages of the boundary scan register) and circuit structure gives a set of elements, which are suspected as faulty on a current test-vector. To organize computational processes, which result in exact diagnosis, it is important metrics or initial information representation form. An interesting solution of the diagnosis problem can be get by application of the Boolean algebra and the fault detection table M that is the Cartesian product of the test T on the set of given faults F, in the aggregate with the experimental validation vector V, where realization of the covering task gives maximally exact result in the DNF form and every term is a possible variant of presence of faults in a device. Thereby, the diagnosis process model is represented by components:

$$\left\{ \begin{array}{l} A = \langle T, F, M, V \rangle, \\ T = (T_1, T_2, \dots, T_i, \dots, T_n); \\ F = (F_1, F_2, \dots, F_j, \dots, F_m); \end{array} \right. \left\{ \begin{array}{l} M = |M_{ij}|, i = \bar{1}, n; j = \bar{1}, m; \\ V = (V_1, V_2, \dots, V_i, \dots, V_n); \\ \{V_i, T_i, M_{ij}, F_j\} \in \{0,1\}. \end{array} \right. \quad (9)$$

The diagnosis problem solution consists of analysis of the fault detection table, formed at fault simulation, by writing of logical product of disjunctions (CNF), which are formed by unit values of the fault detection table rows (5). Then CNF is transformed to DNF (6) by means of equivalent transformations. Therefore it is turned out the Boolean function, where terms (logical products) are full solution set that is fault combinations, giving the experimental validation vector, formed in the process of diagnosis experiment, at functional outputs.

The following matrix $M = T \times F$ is an example of algebra-logical analysis of faults on basis of the fault detection table in functional blocks of SoC, quantity of them is equal to 10. A test of the length 11 input patterns verifies all faults set in the table. The experimental validation vector of a digital unit $V = (10001001001)$, formed in the process of diagnosis experiment, fixes discrepancy between unit outputs and the model (the gold standard) on four (1, 5, 8 and 11) test patterns.

$T_i \backslash F_j$	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	V
T_1				1						1	1
T_2		1					1				0
T_3			1			1			1		0
T_4	1										0
T_5					1			1			1
T_6	1	1									0
T_7			1								0
T_8				1							1
T_9					1	1					0
T_{10}							1				0
T_{11}								1	1	1	1

$M = T \times F =$

In compliance with quantity of units in the experimental validation vector V a number of disjunctive terms CNF that is equal to 4 is formed. Every term is line-by-line writing of faults by logic operation OR which influence on distortion of functional output signals. Then CNF is transformed to DNF on basis of the Boolean algebra rules. It enables to get result:

$$F = (F_4 \vee F_{10})(F_5 \vee F_8)(F_4)(F_8 \vee F_9 \vee F_{10}) = (F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8) \quad (10)$$

The last result contains the fault F_4 in all terms, it means that the fault is present in SoC functionality without fail. If to put forward hypothesis about existence of single fault or minimal quantity of multiple faults, the solution determinate by third term $F = F_4 F_8$ is preferable (in a circuit there exist two faults, which form the experimental validation vector that is equal to $V=(10001001001)$ on the outputs).

SUMULATION FOR F-IP DIAGNOSIS REFINEMENT

Obtained disjunctive form (6) is basic model for defect finding. It does not always identify a functional fault definitely, so it needs in procedures, which improve a diagnosis. First of all it should be noted that all rows of the matrix $M = T \times F$, which were marked by zero values of the experimental validation vector, can be joined in a disjunction of faults (6), which can not be present in a circuit. The creation of form (5) from concerned fault detection table enables to determine all faults, which can not be present in a circuit:

$$\begin{aligned} \bar{F} &= (F_2 \vee F_7) \vee (F_3 \vee F_6 \vee F_9) \vee (F_1) \vee (F_1 \vee F_2) \vee \\ &\vee (F_3) \vee (F_5 \vee F_6) \vee (F_7) = \\ &= (F_1 \vee F_2 \vee F_3 \vee F_5 \vee F_6 \vee F_7 \vee F_9). \end{aligned} \quad (12)$$

Analysis of the expressions, represented by formulas (11) and (12) results in interesting conclusions: 1) Faults, which can not be present in a circuit, are determined in the DNF terms, obtained by zero rows concerning the experimental validation vector; 2) Faults, which are in DNF (14), must be removed from function (12); 3) In this case removal of the fault F_5 results in breakup of two terms $F_4 F_5 F_9 \vee F_4 F_5 F_{10}$, as far as without the fault F_5 every of them separately can not form given experimental validation vector; 4) So, it is make the sole conclusion – double fault that is determined by the term $F = (F_4 F_8)$ is present in a circuit; 5)

Computational complexity of gaining of exact and full solution set is determined by expression

$$Q = 2^{m+1}(2m+1), m \text{ is a number of faults.}$$

If to designate absence of the concrete fault $F_i = 0$, it can to form input conditions for DNF (11) for subsequent simulation of the function on the following conditions: $(F_1, F_2, F_3, F_5, F_6, F_7, F_9) = (0000000)$.

Then simulation result of the function $F = (F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8)$ is equal to $F_4 F_8$. Really, if the faults $(F_1, F_2, F_3, F_5, F_6, F_7, F_9)$, which are verified on the test patterns theoretically, give the negative result (don't distort the output states), it means they are absent in a circuit. Support of this fact is corroborated by the following proofs.

Lemma 1. Full set of all possible fault combinations, which are verified by the test T , is determined as DNF, and obtained by transformation of a conjunctive form

$$F = \bigwedge_{\forall V_i=1} \left(\bigvee_{j=1, \dots, m} F_j \right) = \bigvee_{i=1}^m \left(\bigwedge_{j=1}^m F_j \right),$$

every term of that is written by unit values of the fault detection table row [18] $M = T \times F$ corresponding to the experimental validation vector state $V_i = 1$.

Initial information, formed in compliance with unit values of the experimental validation vector, is full model of faulty behavior of a real object, which forms the experimental validation vector with fixed quantity of units (fault detection table rows) that is equal to k . Every row forms a fault disjunction, written by OR. A number of such disjunctions is equal to k , they are logical multiplied and form full and consistent set of events (faults), which are present in a circuit simultaneously. By multiplication of elementary disjunctions with subsequent simplification of the expressions and using the axioms ($a \vee ab = b$; $a \vee a = a$) DNF that includes all possible combinations, written in the elementary conjunctions form, is turned out. Considering identity of made transformations the obtained function is equivalent to the initial CNF at logic and it is technological notation of all solutions (fault combinations), which are in a circuit, essentially.

Lemma 2. All faults, verified in the fault detection table rows $M = T \times F$ and marked by zero values of the experimental validation vector $V_i = 0$ are absent in a real object.

Really, the table $M = T \times F$ has unit and zero rows concerning the experimental validation vector value: $M_p(0110) \rightarrow V_p = 1$; $M_q(0101) \rightarrow V_q = 0$.

The row p detects presence of two faults $F_2 \vee F_3$ in a circuit. The row q evidences of theoretical verification of the faults $F_2 \vee F_4$ if the vector is equal to 1: $V_q = 1$. But practically the signal $V_q = 0$ shows non-essentiality of the faults $F_2 \vee F_4$ for distortion of circuit outputs. Or these faults are absent in a tested device. Put zero signals for $F_2 \vee F_4$ in the function $F = F_2 \vee F_3$ and obtain the result: $F = F_2 \vee F_3 \Big|_{F_2=F_4=0} = F_3$. Analogous, all faults which

are determined in the rows, corresponding to zero values of the experimental validation vector, are absent in a circuit. But if it is true they must be removed from DNF, written by unit values of the vector V . So, there are DNF terms and a fault set, which can not exist in a circuit for given experimental validation vector and the procedure of substitution of zero signals in the variables of elementary conjunctions of DNF function can be carried out. But, in consideration of the fact $0 \wedge a \wedge b \wedge c \dots = 0$ the result of substitution and subsequent transformations to obtain minimal function will have only the terms, which don't have variables (faults) with zero signal value. It means that the faults which concern to zero fault detection table rows (concerning the vector V), will be removed from DNF.

Theorem 1. Minimal set of all possible fault combinations, which are determined by the fault detection table $M = T \times F$, is computed by DNF simulation on an initial conditions set

$$F = \bigvee_{i=1}^{2^m} \left(\bigwedge_{j=1}^m k_j F_j \right) \Big|_{(\forall F_q=0) \leftarrow (\exists M_{pq}=1) \& (V_p=0)},$$

specified by zero values of all verified faults, which correspond to zero signals of the experimental validation vector.

In compliance with lemma 1 full set of all possible fault combinations, verified by a test, is determined in DNF form

$$F = \bigvee_{i=1}^{2^m} \left(\bigwedge_{j=1}^m k_j F_j \right),$$

that forms all solutions, which satisfy to unit values of the experimental validation vector $V_q = 1$. It can be decreased by removal of the faults, which are verified by a test theoretically, but really they don't distort the output states on the test patterns, that mean complete absence of them in a real circuit. So, they can be removed from DNF terms, which are a full set of all possible combinations. The removal mechanism, according to lemma 2, is substitution of zero variable values to DNF terms and subsequent simulation (simplification) of the function. If a term has zero-component one of the variables F_i , according to the algebra of logic whole term is turned into 0, that means removal of it from DNF. So, after minimization subject to the condition of lemma 2 the minimal DNF is turned out that contains the minimal quantity of possible fault combinations (single and multiple ones) that can not be decreased without additional diagnostic information incoming from the multiprobe on basis of BS-register.

CONDITIONAL F-IP DIAGNOSIS

To decrease of precautionary faults field essentially the half-division method is used [10], it is based on use of an interactive procedure of internal check point flexing that provides the obtained fault DNF by additional information to decrease a fault set. In this case as such tester the boundary scan register can be used that is able to determine an internal line state for fault removal or its confirmation. The check point choice strategy is oriented on approximately half-division of

precautionary set (removal of half faults by simulation on every step) and simplification of the initial DNF. The essence of the half-division method on disjunctive normal form that represents all possible fault combinations in a circuit can be demonstrated by the following example:

$$F = (F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8).$$

Choice of the first check point $F_9 = 0$ turns the Boolean function into reduced expression:

$$F = \begin{cases} F_9 = 0 \rightarrow (F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8) = F_4 F_5 F_{10} \vee F_4 F_8; \\ F_9 = 1 \rightarrow (F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8) = F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8. \end{cases}$$

If $F_9 = 1$, it means confirmation of a line fault and decrease of DNF size do not happen. It is necessary to orient the check point choice algorithm on maximal decrease of the initial DNF after definition of the initial conditions ($F_j = \{0,1\}$) for simulation. Weights of DNF powers, obtained in the process of simulation the both verification states, can be used as the check point choice criterion. Check point choice rules are regulated by the following assertions.

Assertion 1. If F_j is present in all DNF terms, there exists given fault in a circuit without fail and it is not necessary to test it. Otherwise, if to suppose that verification result is zero, all terms is turned into zero and this fact contradicts to the existence condition of nonzero values of the experimental validation vector V .

Assertion 2. There is a single fault combination in a circuit that is determined by a single DNF term. If it is found one confirmed solution in the DNF term form other terms should be removed from consideration by reversal of them to zero.

So, the check point minimization problem is reduced to the carrying out of two alternative strategies: 1) consideration of variables in the terms of minimal length to corroborate all faults in a term by flexing; 2) verification of such variables, which turn maximal quantity of DNF terms to zero.

If there exists the function $F = (F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8)$ that has a term of minimal length 2 and the variable F_4 in all terms, a single and the better solution is the verification F_8 , which gives a required fault set at positive result and remaining two terms, which should be probed, at negative one:

$$F = \begin{cases} F_8 = 0 \rightarrow (F_4 F_5 F_9 \vee F_4 F_5 F_{10}); \\ F_8 = 1 \rightarrow (F_4 F_8). \end{cases}$$

The verification F_5 gives the following results of after probe simulation of two function variants:

$$F = \begin{cases} F_5 = 0 \rightarrow F_4 F_8; \\ F_5 = 1 \rightarrow F_4 F_5 F_9 \vee F_4 F_5 F_{10} \vee F_4 F_8. \end{cases}$$

Then, after ($F_5 = 1$), two verifications from three ones (F_9, F_{10}, F_8) should be carried out, they remove all terms except one that defines a solution:

$$F = \begin{cases} F_9 = 0 \rightarrow F_4 F_5 F_{10} \vee F_4 F_8; \\ F_9 = 1 \rightarrow F_4 F_5 F_9. \end{cases} \quad F = \begin{cases} F_{10} = 0 \rightarrow F_4 F_8; \\ F_{10} = 1 \rightarrow F_4 F_5 F_9. \end{cases}$$

The finishing diagnosis procedure criterion is obtaining of one DNF term that identifies the presence of a multiple fault in SoC functionality. Below it is proposed one more example of carrying out the interactive diagnosis procedure by using DNF:

$$F = (F_3 F_4 F_5 F_9 \vee F_2 F_5 F_6 F_{10} \vee F_1 F_4 F_8 F_9 \vee F_2 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}).$$

There is a multiple fault $F = F_1 F_2 F_8 F_{10}$ in a device.

1) Weight count of every variable that is a part of DNF is carried out:

F_i	F_1	F_2	F_3	F_4	F_5	F_6	F_8	F_9	F_{10}
$W(F_i)$	2	2	1	4	2	1	3	3	2

2) Fault presence probability in a circuit is correlated with their weighting coefficients. So, to get a single solution in the DNF term it is necessary to choose the variables with minimal weight as check points, which will turn the terms into zero. So the first and second check points are (F_3, F_6), they have minimal weight:

$$F = (F_3 F_4 F_5 F_9 \vee F_2 F_5 F_6 F_{10} \vee F_1 F_4 F_8 F_9 \vee F_2 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}) \Big|_{(F_3=0)} =$$

$$= F_2 F_5 F_6 F_{10} \vee F_1 F_4 F_8 F_9 \vee F_2 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}.$$

$$F = (F_2 F_5 F_6 F_{10} \vee F_1 F_4 F_8 F_9 \vee F_2 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}) \Big|_{(F_6=0)} =$$

$$= F_1 F_4 F_8 F_9 \vee F_2 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}.$$

3) After every step re-calculation of weighting coefficients is performed, it enables to correct following steps:

F_i	F_1	F_2	F_3	F_4	F_5	F_6	F_8	F_9	F_{10}
$W(F_i)$	2	1	-	3	-	-	3	2	1

Here it is established the event of the faults (F_4, F_8) presence in a circuit, which aren't subject of probing in compliance with the condition of assertion 1.

Verification of the fault F_2 gives the following result:

$$F = (F_1 F_4 F_8 F_9 \vee F_2 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}) \Big|_{(F_2=0)} =$$

$$= F_1 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}.$$

Coefficients re-calculation:

F_i	F_1	F_2	F_3	F_4	F_5	F_6	F_8	F_9	F_{10}
$W(F_i)$	2	-	-	2	-	-	2	1	1

makes provision for presence of the faults (F_1, F_4, F_8) in a circuit and additional verification of one in the lines (F_9, F_{10}):

$$F = (F_1 F_4 F_8 F_9 \vee F_1 F_4 F_8 F_{10}) \Big|_{(F_9=0)} = F_1 F_4 F_8 F_{10}.$$

So, in the process of carrying out of four flexing, which are represented by the lines (F_3, F_6, F_2, F_9), the exact diagnosis was obtained: there is the multiple fault $F = (F_1, F_4, F_8, F_{10})$ in a circuit.

CONCLUSION

Scientific novelty and practical importance of the research: 1) Algebra-logical method and algorithm of fault embedded diagnosis in functional blocks of SoC that uses preliminary analysis of the fault detection table for decrease of its size and the volume of subsequent calculations, related with DNF forming, which determines all solutions of SoC functionalities diagnosis, are proposed. 2) It is proposed the reduced SoC Functional Intellectual Property Infrastructure that is characterized by minimal set of the embedded diagnosis processes in real time and enables to realize the services: testing of the nominal functions on basis of generable input patterns and analysis of output reactions; fault diagnosis with given resolution of fault

location by means of utilization of the IEEE 1500 standard. The algebra-logical representation of the covering problem has appeal that is directed on optimal solution of all synthesis and analysis problems of complex systems, where the mapping problem exists: 1) a specification – a set of library components; 2) faults – test patterns; 3) functionalities – Testbench; 4) faulty elements – reserved ones; 5) object states – surveillance lines. Further research is oriented on development of testability structure of the system and hardware BIRA module for embedded repair of whatever components in appearance of faults on production and operating stages.

THE AUTHORS

Vladimir Hahanov – Dean of the Computer Engineering Faculty, Doctor of Science, Professor. Eugenia Litvinova – assistant professor, Doctor of philosophy. Vladimir Obrizan, postgraduate student of the Computer-Aided Design Department, Computer Engineering Faculty, Kharkov National University of Radioelectronics, Ukraine, Lenin Ave. 14, Kharkov, Ukraine, 61166, phone: (057) 70-21-421, (057) 70-21-326. E-mail: hahanov@kture.kharkov.ua

REFERENCES

- [1] J.Bergeron. Writing testbenches: functional verification of HDL models, Springer, 2003, 512 p.
- [2] P.Rashinkar, P.Paterson, L.Singh. System-on-chip Verification: Methodology and Techniques, Kluwer Academic Publishers, 2002, 393 p.
- [3] IEEE-1800 IEEE Standard for System Verilog Language, 2005, 586 p.
- [4] Y.Zhong, S.G.Dropsho, X.Shen, A.Studer, C.Ding. Miss Rate Prediction Across Program Inputs and Cache Configurations, IEEE Trans. on Computers, 2007, P. 328-343.
- [5] V.Hahanov, H.Kteaman, W.Ghribi, E.Fomina. HEDEFS – Hardware embedded deductive fault simulation, Proceedings of the 3rd IFAC Workshop, Rydzyna, Poland, 2006, P. 25-29.
- [6] K.Rossen. Discrete Mathematics and its Applications, McGraw Hill, 2003, 824 p.
- [7] V.I.Hahanov, I.V.Hahanova. VHDL + Verilog = Synthesis for minutes. Kharkov: SMIT, 2007, 264 p.
- [8] Z.Yervant. What is Infrastructure IP?. IEEE Design & Test of Computers, May-June 2002, P. 5-7.
- [9] Z.Yervant, G.Dmytris. Gest editors' introduction: Design for Yield and reliability, IEEE Design & Test of Computers, May-June 2004, P. 177-182.
- [10] IEEE 1500 Web Site. <http://grouper.ieee.org/groups/1500/>.
- [11] D.Densmore, R.Passerone, A.Sangiovanni-Vincentelli. A Platform-Based taxonomy for ESL design, Design&Test of Computers, September-October 2006, P. 359-373.