UDC 681.518

Skobtsov Yu. A.[1], Skobtsov V. Yu.[2], Shalyto A. A.[3]

[1]Dr. Sc., Professor, Head of Automated Control Systems department, Donetsk National Technical University, Donetsk, Ukraine
[2]PhD, Associate Professor, Leading Researcher of Information Protection Laboratory, United Institute of Informatics Problems of the NAS of Belarus, Minsk, Belarus
[3]Dr. Sc., Professor, Head of Program Technology department, Saint-Petersburg University of Information Technologies, Mechanics and Optics, Saint-Petersburg, Russia

TEST GENERATION AND SIMULATION FOR CROSSTALK FAULTS

The main models of crosstalk faults are defined: 1) induced positive and negative pulses, 2) induced delays. The purpose of the work is to increase the effectiveness of the methods of constructing checking tests for digital systems based on an evolutionary approach and models of non-const fault. Formalized statement of the problem of test generation for a single crosstalk faults-induced pulses and delays. It is shown that this problem is reduced to solving a system of logic equations in the multi-valued alphabet. The 8-valued alphabet and multi-valued functions for basic gates are defined. Simulation method was developed in the 8-valued alphabet for crosstalk faults. On this basis, the genetic algorithm is proposed for test generation of single cross-faults. The test generation problem for fault-induced delay is formalized. A genetic algorithm of test generation for fault-induced is proposed. Developed algorithms and software for test generation for crosstalk faults, which improve the quality of test generation by using evolutionary techniques. The approbation of the developed methods is implemented at circuits of international catalogs ISCAS85, ISCAS89, which showed an increase in the completeness of tests by 15%.

Keywords: test generation, crosstalk faults, genetic algorithms, multi valued logic, fault simulation.

NOMENCLATURE

a_i is an aggressor line;
B_2 is a binary alphabet;
C_8 is a multivalued alphabet;
c_i is a capacitive value;
DF is a reducing delay;
DR is an increasing delay;
f_i is a primary output
FT is a backward front;
f_a is an evaluation of aggression;
f_e is an evaluation of excitation;
f_p is an evaluation of propagation;
f_f is a fitness-function;
$g()$ is a victim-line function;
$h()$ is an aggressor-line function;
n is a number of primary variables;
NG is a negative pulse;
P_c is a crossover probability;
PG is a positive pulse;
P_m is a mutation probability;
RT is a forward front;
$S0$ is a static zero signal;
$S1$ is a static one signal;
s_i is a flag of belonging the subject line a_i;
x_i is a primary input variable.

INTRODUCTION

The diagnostics of modern digital systems and their element base increasing need of the analysis of not only stuck-at faults, but also of more adequate models of physical defects. Technological progress leads to density increasing on chip, rising number of crossings in conductive layers and speeding-up time frequency. As a result, the analysis of physical defects affecting on time parameters of a circuit and crossings in conductive layers is need. Especially it is concerned deep submicron level design.

Increasing transistor number on chip leads to that bigger number of element switches simultaneously and reduces voltage level of elements, and enlarges delay time. At the same time, some crossing lines, which are suppose to be electrically isolated, can interfere with each other. One of such interactions, produced by parasitic coupling between conductors, is called «crosstalk», and can bring to functional problems, to performance degradation and makes worse time characteristics.

The purpose of the work is to increase the effectiveness of the methods of constructing checking tests of digital systems based on an evolutionary approach and models of non-constfault.

1 PROBLEM STATEMENT

Let the states of victim-line g and aggressor-line h are described by the Boolean functions of primary inputs variables, $g(x_1, x_2, ..., x_n)$ and $h(x_1, x_2, ..., x_n)$. Similar on every j-th primary output it is implemented Boolean function $f_j(g, h, x_1, x_2, ..., x_n)$, which depends of primary input variables $\underline{X} = (x_1, x_2, ..., x_n)$ and internal variables g and h.

For crosstalk fault detection with positive pulse (glitch) it is necessary the primary inputs values, which guarantee on aggressor-line h front RT, and on victim-line g – S0 in fault-free case. Then for given positive glitch crosstalk fault test generation problem is reduced to searching solution of following Boolean equations system (1)–(3):

$$h(x_1, x_2, ..., x_n) = RT, \quad (1)$$

$$g(x_1, x_2, ..., x_n) = S0, \quad (2)$$

$$F_j(S0, R, x_1, x_2, ..., x_n) \oplus F_j(PG, R, x_1, x_2, ..., x_n) = 1, \quad (3)$$

where (1) RT guarantees value on the aggressor-line h, (2) – the $S0$ value on the victim-line g and guarantees propagation of positive glitch effect from the victim-line g to some primary output.

Similar for crosstalk detection with negative glitch it is necessary primary inputs values that set-up value FT on aggressor-line h and on victim-line g value S1. Then for

DOI 10.15588/1607-3274-2015-4-11

negative glitch test generation problem is reduced to searching solution of following Boolean equations system (4)–(5):

$$h(x_1, x_2, ..., x_n) = FT, \quad (4)$$

$$g(x_1, x_2, ..., x_n) = S1, \quad (5)$$

$$F_j(S1, FT, x_1, x_2, ..., x_n) \oplus F_j(NG, FT, x_1, x_2, ..., x_n) = 1. \quad (6)$$

The test generation problem for the crosstalk delay fault of backward/forward front is reduced to searching solution of following Boolean equations system (7)–(9):

$$h(x_1, x_2, ..., x_n) = FT / RT, \quad (7)$$

$$g(x_1, x_2, ..., x_n) = RT / FT, \quad (8)$$

$$F_j(FT / RT, RT / FT, x_1, x_2, ..., x_n) = DR,$$

where (7) FT/RT guarantees value on the aggressor-line *h*, (8) –RT/ FT value on the victim-line *g* and (9) guarantees propagation of crosstalk delay effect from the victim-line *g* to some primary output. In the right parts of equations (7) and (8) we cited two values through slant line for backward/ forward front cases.

Obviously that checking test for crosstalk faults should consist of pattern pairs, which guarantee mentioned above conditions. If the above formulated equation systems do not have solution then crosstalk induced pulse and delay faults are undetectable at least in this problem setting.

2 REVIEW OF THE LITERATURE

The works in the aria of crosstalk fault simulation and test generation are carried out since middle of 90-th and are urgent for theory and practice of reliable design and diagnostics of modern digital devices. They are based on the application of different approaches such as timing analysis, multivalued logics, critical path method, PODEM algorithm modifications, structural deterministic and simulation based methods etc [1–3]. There are considered two basic types of crosstalk faults: crosstalk induced pulses and crosstalk induced delays.

In the first case, quickly switching conductive line – *aggressor*, can induce short pulse on static line – *victim*. First type faults are often named «crosstalk glitch» and caused by parasitic inducing between neighbor conductor lines, which have inductive and capacitive constituents in general. Depending on amplitude and width these pulses can essentially effect on circuit characteristics [1, 4].

In the second case, an induced delay takes place when on aggressor and victim lines (almost) synchronous signal crossings are happened. If on both lines crossings have the same direction then crossing time is reduced, and hence, signal propagation delay time is reduced. This effect is called crosstalk speedup. If signal crossing have opposite directions then crossing time increases and effect of crosstalk slowdown takes place [2]. If induced noise is over the boundary voltage or induced delay exceeds allowed value on victim line then it can lead to logical failures and functional problems on neighbor flip-flops or outputs.

Research on modeling and testing «crosstalk» faults since the mid90s. In [2] it was suggested the method, that is based on search of input pattern pair that sets up defined value of signal on victim-line and excites crossing on aggressor-line, which provide fault effect propagation to primary outputs. Crosstalk induced pulses were considered in this paper. The authors in [3–5] have developed a mixed test signal generator XGEN for «crosstalk» induced delays. They suggested a mixed test generation method, where analog simulation allowing evaluating signal delays is used in binary patterns search.

Using Laplace transform they received expressions of faults "crosstalk" in the s-domain, which are used to analyze the dependence of the attributes of the pulse in the circuit with lumped parameters and temporal characteristics of fronts. Static timing analysis allows to define the time windows for inputs and outputs of elements. The required time window is determined as the intersection of time windows «aggressor» and «victim». For a given failure – a particular pair of «aggressor-victim» approach allows to set the required values for interacting lines and extend the influence of the fault to the external output.

During test generation is used 11-digit alphabet and modification of PODEM at step justification. The developed algorithm does not guarantee the generation of tests as a result of restrictions on the conditions of propagation effects of malfunctions and used (11-valued) logic system.

In [6], the authors have developed a model path delays in combination with a critical path and the sources set, by interacting with it. It is used as a basis for test generation method for the fault of the «propagation delay» without phase justification. And on the stage justification it applies genetic algorithm, which uses in the process of test generation temporal characteristics.

Further, this approach was developed in [7], which provides a solution to the problem of test generation in the case of many of the «aggressors», so acting a certain way. Here the implication graph is constructed which allows for the logical and structural information schemes to check conflicts in the process of test generation.

The paper [8] is devoted to testing of faults «crosstalk» – induced delays. An algorithm for test construction uses the generation of critical paths based on static timing analysis of the circuit. Testability of sustainability criteria used for checking the sensitivity of paths. To activate the sensor path pair «aggressor-victim» in a way to maximize the impact of the aggressor on the path and cause the signal propagation delay along this path.

In [9], a test generator unit based on accurate delay pattern is proposed. It is activated by a subpath to meet the conditions of signal transitions. Known fault model of the «delay» is modified for testing of induced delays. To reduce the set of analyzed faults preprocessing is performed to critical paths.

The authors in [10] have developed a method based on test generation PODEM. To test the «crosstalk» must be necessary transitions to the external inputs and provide a propagation of the influence of the fault on the external output. Each of these tasks is formulated separately, and the final decision is the result of crossing making these tasks. The 11-valued alphabet and PODEM method modification at justification stage are used during test

generation. Developed method does not guarantee test generation in consequence of restrictions for fault effects propagation conditions and using logical system.

In [12] to solve the problem of test generation for «crosstalk» is used 0–1 integer programming based on traditional algorithms for stuck-at faults. Maximum activation of the aggressor is formulated in terms of linear programming, and the impact of the propagation of the fault is solved by traditional means. It should be noted that all of the above papers in the test generation using mechanisms of «turning back» scheme that require significant computing resources. On the other hand there is a group of methods, which are based on modeling and did not require a «bounce back».

These include, for example, [13] where the simulation is used to test the cross-faults. It uses an algorithm for editors fault-candidates, which allows to obtain a compact set of target faults due to the exclusion of those problems that can never be activated and detected. Here editing algorithm for faults-candidates is used that allow get compact set of target faults due to exclusion that faults, which never can not be excited and detected. Under that sequential fault path delay simulator was applied. Except that analog macro model for evaluation of signal delays in consequence of crosstalk interfering is used. Here one aggressor-victim pair is associated with every crosstalk fault. Input patterns are generated randomly during search process. Authors of [14] suggested fault simulator that considers one victim-line and set of aggressor-lines. Binary logical simulation and, as a base, stuck-at faults tests are used in test generation. Note that simulation-based methods demand good heuristics for test pattern generation and generated test sequences have big length, as a rule.

3 MATERIALS AND METHODS

An application of multivalued alphabets, which allow more adequate simulate physical processes in digital devices at logical level, is widely used method to increase effectiveness test generation and simulation algorithms for digital circuits [4]. In table 1 there is represented multivalued alphabet C_8 (its symbols, encoding and physical interpretation), which we suggested to use in test generation and simulation algorithms for crosstalk faults.

Note that the primary inputs variables can take only following four values: «static 0» *S*0 $(0 \to 0)$, «static 1» **S**1 $(1 \to 1)$, forward front RT $(0 \to 1)$, backward front FT $(1 \to 0)$. Internal lines variables also can accept following additional values PG (positive pulse), NG (negative pulse), DR (increasing delay), DF (reducing delay).

From table 1 we can see that initial signals are binary in fact. But it is more effective to simulate and compute in terms of multivalued alphabet symbols, which allow to process several clocks simultaneously. So the C_8 symbols belong to Cartesian product of binary alphabets B_2: $C_8 \subseteq B_2 \times B_2 \times B_2$.

For logical simulation purposes we should define logical functions for basic gates in the terms of C_8 alphabet. We can do it in table form that is enough effective for simulation goals. The table model for the basic gates AND, OR, NOT are defined constructively on the base of physical sense of C_8 symbols and functionality logics of given gates, and there is represented in tables 2–4.

Table 1 – Multivaluedalphabet C_8

Symbol C_8	Interpretation	Binary encoding		
		tt-1	tt	t+1
S0	Static 0	00	00	00
S1	Static 1	11	11	11
RT	Forward front	00	11	11
FT	Backward front	11	00	00
PG	Positive pulse	00	11	00
NG	Negative pulse	11	00	11
DR	Increasing delay	00	00	11
DF	Reducing delay	11	11	00

Table 2 – Definition of gate AND in C_8 alphabet

AND	S0	S1	R	F	PG	NG	DR	DF
S0	S0	S0	S0	S0	S0	S0	S0	S0
S1	S0	S1	R	F	PG	NG	DR	DF
R	S0	R	R	S0	PG	DR	DR	PG
F	S0	F	S0	F	S0	F	S0	F
PG	S0	PG	PG	S0	PG	U	S0	S0
NG	S0	NG	DR	F	S0	NG	DR	F
DR	S0	DR	DR	S0	S0	DR	DR	S0
DF	S0	DF	PG	F	PG	F	S0	S0

Table 3 – Definition of gate OR in C_8 alphabet

OR	S0	S1	R	F	PG	NG	DR	DF
S0	S0	S1	R	F	PG	NG	DR	DF
S1	S1	S1	S1	S1	S1	S1	S1	S1
R	R	S1	R	S1	R	S1	R	S1
F	F	S1	S1	F	DF	NG	NG	DF
PG	PG	S1	R	DF	PG	S1	R	DF
NG	NG	S1	S1	NG	S1	NG	NG	S1
DR	DR	S1	R	NG	R	NG	DR	S1
DF	DF	S1	S1	DF	DF	S1	S1	DF

Table 4 – Definition of gate OR in C_8 alphabet

NOT	S0	S1	R	F	PG	NG	DR	DF
	S1	S0	F	R	NG	PG	DF	DR

Let consider a combinational circuit with primary inputs $(x_1, x_2, ..., x_n)$ and primary outputs $(f_1, f_2, ..., f_m)$ that have crosstalk fault on some line. Under test generation for crosstalk faults it is necessary:

– find out input patterns, which cause required signal transient on aggressor line;

– find out input patterns guaranteeing necessary static signal on victim line;

– find out input patterns propagating arisen crosstalk effect from victim line to primary outputs.

Consider the problem of the test generation for multiple delay faults induced by the various lines of aggressors. In this case, the victim lines are included in some path, connecting the external input to an external output of the circuit [2].

The set of the aggressor lines is formed by those lines, which could affect the victim line and thus cause a delay of signal propagation in the specified path. In solving this problem must be solved at least three sub-tasks [2]: 1) selecting the set of critical paths, forming a line of victims; 2) The choice of the set of aggressor lines for a given critical path; 3) generation of input test pair patterns that check-induced delay for a given circuit path and the set of agressors lines.

Selection of multiple aggressor lines set for given victim line can be formalized as follows. Let $A=\{a_1,a_2,\ldots, a_m\}$ is the set of possible aggressors lines. For each aggressor line a_i, we define c_i – the extent of its effect at the victim, which is determined, above all, the value of the capacitive coupling between these lines.

Next, we define a set of Boolean variables $S=\{s_1, s_2,\ldots, s_m\}$, where each variable s_i corresponds to its aggressor a_i. At the same time

$$s_i = \begin{cases} 1, \text{if } a_i \text{ belong to current aggressor set;} \\ 0, \text{otherwise,} \end{cases}$$

That is, each variable s_i is a flag of belonging the subject line a_i to the current set of line-aggressors. Then the degree of effect of the current set of aggressors to the victim line can be defined as $\sum_{i=1}^{m} s_i \cdot c_i$.

To maximize cross-induced delay for the current set of aggressors, should be excited so that the effect of delay at the victim line was a maximum. So it is necessary to find a Boolean vector S, which takes place $\max \sum_{i=1}^{m} s_i \cdot c_i$ and execute the necessary logical conditions for generation the test pair [2]. Obviously, this task has the overhaul nature and is NP-hard. In the worst case would have to go through all possible set subsets of 2^m aggressor lines.

4 EXPERIMENTS

Crosstalk fault simulation is executed on the base of the event-driven simulation using multivalued alphabet C_8 [4]. The simulation goal is checking crosstalk fault detection by given input sequence. The generalized crosstalk fault simulation algorithm for both crosstalk faults is given below.

```
Crosstalk fault simulation (combinational
logical circuit, input sequence T, line pair
aggressor, victim)){
Circuit description input;
Circuit initialization;
Input sequence T entry;
Crosstalk fault entry;
FOR every pattern of input sequenceT{
Multivalued input pattern generation in C8
alphabet;
If (condition crosstalk fault appearance){
Crosstalk fault effect entry and activation
for given line pair(aggressor, victim);
Input pattern fault-free and fault simulation;
If (there is primary output with crosstalk
effect) given crosstalk fault is detected;
}
}
}
```

For positive and negative pulse induced crosstalk effect entry PG and NG values should be set up on victim-line in the process of fault simulation.

The example 1 of multivalued logical simulation in C_8 alphabet is represented on figure 1 for circuit $C17$ from benchmark ISCAS85. The positive glitch fault between aggressor-line 19GAT(7) and victim-line 16GAT(8) is considered. Note that the values for one test pair detected given fault are represented here (Fig. 1).

The example 2 of delay induced fault for circuit C17 between victim-line h and aggressor-line i is presented on the Fig. 2. In this case we can set forward front value RT on aggressor-line I and backward front value FT at victim-line.

Genetic algorithm and multi-valued logic can be used in test pattern search process [15, 16]. At that the genetic algorithm provides a mechanism of random direct search of test patterns pairs, which satisfy noted conditions. To our point of view, application of evolutionary methods in test generation for considered faults is more justified then for classical stuck-at faults. It can be drawn an analogy with numerical optimization problems where evolutionary approach is used in cases of inapplicability of classical gradient methods first of all. It is necessary to note that genetic algorithms allow reduce synthesis problem to analysis problem (in some sense). Under condition of analysis tools presence (fault-free or faulty circuit simulators) genetic algorithms provide direct random search for synthesis problem solution. Under that synthesis and analysis problems can be considered at different levels.

Classical «simple» GA uses three basic operators: reproduction, crossover and mutation. Using these operators, the population (the set of individuals-solutions of considered problem) evolves from one generation to another. Classical steady state GA may be represented as the following sequence of operations that is shown in flow chart of fig. 3.

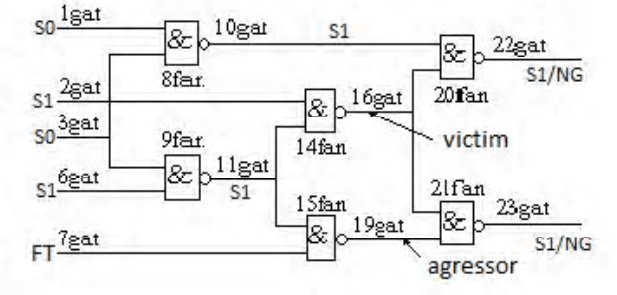

Figure 1 – Logical simulation of positive glitch in circuit C17

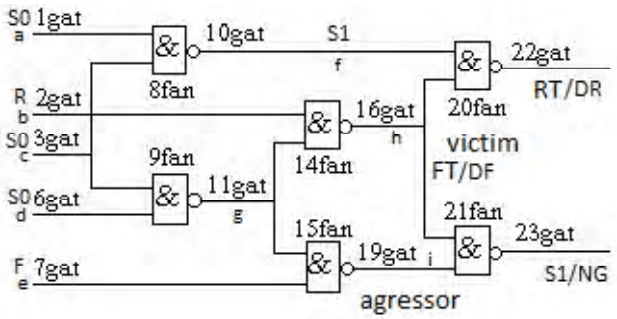

Figure 2 – Logical simulation of delay induced fault in circuit C1

In developed test generation genetic algorithm the initial population of size N is generated randomly. Here each chromosome $(x_1, x_2, ..., x_n)$ is of length $2n$ bits, where n is the inputs number. Note that each input $x_i \in B_2 \times B_2$ corresponds two consecutive bits, representing one of four values S0, S1, RT and FT. The set of such pattern pairs is population.

Generation of new population is done with using following genetic operators. The tournament selection is used here. On tournament selection [4,5] m individuals are chosen randomly, then the best one of them is selected as potential parent. This procedure is continued until intermediate population is not formed. Here selection parameter is $2 \le m \le N (m=2,3)$. Then crossover is performed with a high probability P_c. As crossover operator we apply uniform crossover in this case (Fig. 4).

The formed offspring are mutated with a low probability $P_m << 1$ and inserted in current population. The classical mutation operator is used. But it is applied for two-bit groups instead one bit according to values S0, S1, RT and FT.

Parent selection is biased towards patterns with good cover faults properties. We define fitness-function as linear combination of three component:

$$f_f = w_e f_e + w_p f_p + w_c f_a, \qquad (10)$$

where f_e evaluates input pattern – individual ability to excite necessary value on victim-line, f_p evaluates individual ability to propagate crosstalk effect to primary outputs, f_a evaluates input pattern – individual ability to take into account aggressor lines effect. To evaluate fitness-function components results of logical simulation in multi-valued alphabet C_8 is used. The next population is generated based on the current population with using the same genetic operators. The process is repeated until the stop criterion is fulfilled.

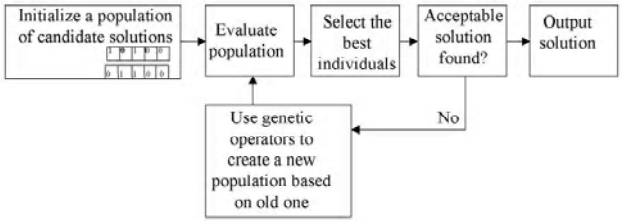

Figure 3 – Classical «simple» GA flowchart

Mask OC	1	0	1	0	1
1-st parent	1	1	0	1	1
	0	0	0	1	0
	↓		↓		↓
Offspring	1	0	0	0	1
	0	1	0	0	0
		↑		↑	
2-nd parent	0	0	0	0	1
	1	1	1	0	1

Figure 4 – Uniform crossover operator

When implementing the proposed approach based Finite State Machines programming [17], which allows to use object-oriented features of faults. In addition, this approach allows to go from a simple genetic algorithm to genetic programming that enables to operate with more complex data structures of various dimensions for potential solutions and extend a class of simulated faults.

5 RESULTS

For example 1, the simulation results show that positive glitch fault between aggressor-line 19GAT(7) and victim-line 16GAT(8) is detected on the both output lines for the multivalued input pattern 1gat=S0, 2gat=S1, 3gat=S0, 6gat=S1, 7gat=FT, which corresponds to input test pair 1gat=0->0, 2gat=1->1, 3gat=0->0, 6gat=1->1, 7gat=0->1.

For example 2, on the figure 2 it is represented the example of multivalued test pattern 1gat=S0, 2gat=RT, 3gat=S0, 6gat=S0, 7gat=FT, which detects given delay induced fault for circuit C17 at the lines 16gat=DF (forward front delay) 22gat=DR (backward front delay).

The proposed method testing was performed at the circuits of international catalogs ISCAS85, ISCAS89, which according to international standards adopted to test new methods of test generation. For a subset of ISCAS85, ISCAS89 a number of computer experiments were performedwhich confirm the effectiveness of the developed methods. The results of test generation for single cross-fault type including positive (negative) induced impulse and induced delay are represented at table 5.

Except single cross-faults, the experiments were also performed for cross faults with many aggressors. The results are shown in Table 6.

6 DISCUSSION

It is seen that the average genetic algorithm provides a 2-fold greater fault coverage than random method. At the same time, for the combinational circuits fault coverage is 70%. Note that for certain schemes (eg. S526, S420.1, S1238) pseudo-random method does not generate check tests. For sequential circuits are somewhat worse results, and the average completeness of test is about 40%. But it should be noted that many of the faults marked as targets (the number of which is shown in the 2nd column tabl.6 is not feasible, due to contradictory logical conditions and, therefore, the actual fault coverage for cross fault is much higher.

CONCLUSIONS

It was shown for combinational digital circuits that application of genetic algorithms and multivalued event-driven simulation in C_8 alphabet allows effectively solving the test generation problem for crosstalk faults.

Table 5 – Test generation for single cross talk faults

Circuit	Fault coverage			
	Positive impulse PG	Negative impulse NG	Delay front DR	Delay back front DR DF
C432	98.05	99	100	98.01
C499	99	84.81	99.4	99.8
C880	97.09	95.13	98.16	98.85
C1355	88.15	94.41	99.18	99.43
C1908	89.75	96.77	95.63	93.88
C2670	90.31	94.15	94.71	90.45
C5315	99.70	99.87	99.71	99.72
C6288	99.91	98.97	99.31	99.32
C7552	97.23	98.28	99.05	97.03

Table 6 – Test generation for cross talk faults with many aggressors

Circuit	Number of target faults	Fault coverage, %		Time (sec)
		Pseudo random	Genetic algorithm	
C17	42	45,23	62.05	0.18
C432	9327	59.52	70.15	241.14
C449	21879	30.27	71.81	591.17
C880	9279	2.31	47.53	628.14
S27	74	6.78	37.87	0.39
S208	743	30.96	32.43	6.42
S208.1	558	1.63	45.21	12.72
S298	537	43.04	62.58	9.43
S344	1190	64.38	66.14	18.53
S349	1197	30.51	57.92	19.24
S526	891	0.3	19.23	21.34
S386	4195	15.81	25.51	58.04
S510	1098	36.83	43.87	14.73
S420.1	1276	0.3	36.93	63.95
S820	7738	26.34	35.41	166.42
S1196	10630	0.3	14.61	782.45
S1238	5822	0.2	13.51	531.23
S1488	4305	18.43	22.45	184.52
S1494	4283	18.23	18.43	165.42
average		22.70	41.24	185.02

The problem of test generation for the new class not s-a-constant faults is characteristic for modern digital circuits, which have a high density of elements and high operating frequencies. It is proposed a multi-valued alphabet C_8, developed models of multi-valued logic elements to this alphabet, which allows to effectively simulate the cross-faults.

The modified genetic algorithm is designed for generation of test input pattern pairs, checking single fault «induced impulses».

Also the problem of the test generation for single cross faults such as «induced delay is considered. Has received the further development of the method of multi-valued logic simulation in the alphabet, which allows you to simulate the new class is not constant – as a cross-fault-induced pulses and induced delays. Modified genetic algorithm for constructing test sets of pairs of input sequences that cross-check single failure of the» induced delay. The problem of test generation for fault-induced delay with many lines-.aggressors is considered.

A software module developed by a multi-valued simulation and genetic algorithms for constructing tests for non-classical (constant) cross-faults, which are integrated into the system simulation and test generation ASMIDA.

Completed testing and verification of the effectiveness of the developed software modules for the logic circuits are not the fault of the international classical catalogs and ISCAS85 ISCAS89, who confirmed to achieved high performance characteristics.

REFERENCES

1. Скобцов Ю. А. Моделирование, тестирование и диагностика цифровых устройств / Ю. А. Скобцов, Д. В. Сперанский, В. Ю. Скобцов : учебное пособие. – М. : Национальный открытый университет «ИНТУИТ», 2012. – 439 с.
2. Rubio A. An approach to the analysis and detection of crosstalk faults in digital VLSI circuits / A. Rubio, N. Itazaki, X. Xu, K. Kinoshita // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. – 1994. – Vol. 13, № 3. – P. 387–394.
3. Chen W. Y. Analytic Models for Crosstalk Delay and Pulse Analysis under Non-Ideal Inputs / W. Y. Chen, S. K. Gupta, M. A. Breuer // International Test Conference, Washington, Nov. 1997 : proceedings. – Los Alamitos : IEEE, 1997. – P. 809–818. DOI:10.1109/TEST.1997.639695.
4. Chen W. Y. Test generation for Cross-Induced Delay / W. Y. Chen, S. K. Gupta, M. A. Breuer //International Test Conference, Atlantic City, 28–30 September 1999 : proceedings. – Los Alamitos : IEEE, 1997. – P.191–200. DOI:10.1109/TEST.1999.805609
5. Chen W. Y. Test generation for Cross-Induced Faults: Framework and computational results / W. Y. Chen, S. K. Gupta, M. A. Breuer // Journal of Electronic Testing : Theory and Applications. – 2002. – Vol. 16. – P. 17–28.
6. Krstic A. Delay Testing Considering Cross-Induced Effects / A. Krstic, J.-J. Liou, Y.-M. Jiang, K.-T. Cheng // International Test Conference, Baltimore, 30 October-01 November 2001 : proceedings. – Los Alamitos: IEEE, 2001. – P. 558–567. DOI:10.1109/TEST.2001.966674.
7. Bai X. HyAC. A Hybrid Structural SAT Based ATPG for Crosstalk / X. Bai, S. Dey, A. Krstic // International Test Conference, Washington, 30 September–2 October 2003 : proceedings. – Los Alamitos: IEEE, 2003. – P. 112–121. DOI:10.1109/TEST.2003.1270831.
8. Arunachalam A. A Novel Algorithm for Testing Crosstalk Induced Delay Faults in VLSI Cicuits / A. Arunachalam, R. Arunachalam // International Conference on VLSI Design, Kolkata, 3–7 January 2005 : proceedings. – Los Alamitos: IEEE, 2005. – P. 479–484.
9. Li H. Selection of Crosstalk-induced Faults in Enhanced Delay test / H. Li, X. Li // Journal of Electronic Testing: Theory and Applications. – 2005. – Vol. 21, № 2. – P. 181–195.
10. Palit A. K. Test Pattern Generation for Crosstalk Faults in DSM chips using Mofified PODEM / A. K. Palit, K. K. Duganapalli, W. Anheier // Eectronics System integration Technology Conference, Greenwich, 1–4 September 2003 : proceedings. – Los Alamitos : IEEE, 2003. – P. 393–398. DOI:10.1109/ESTC.2008.4684311.
11. Chun S. XPDF-ATPG: An Efficient Test Pattern Generation for Crosstalk-Induced Faults/ S. Chun, Y. Kim, M.-H. Yang, S. Kang // Asian Test Symposium, Sapporo, 23–24 November 2008 : proceedings. – Los Alamitos : IEEE, 2008. – P. 83–88.
12. Ganeshpure K. P. On ATPG for Multiple Aggressor Crosstalk Faults in Presence of Gate Delays / K. P. Ganeshpure, S. Kundu / / International Test Conference, Santa Clara, 23–25 October 2007 : proceedings. – Los Alamitos : IEEE, 2007. – P. 1–7.
13. Chary Sh. Automatic path delay test generation for combined Resistive Vias Resistive bridges and Capacitive Crosstalk delay faults / Sh. Chary, M. L. Bushnell // International conference on VLSI Design, Hyderabad, 3–7 January 2006. – Los Alamitos : IEEE, 2006. – P. 413–418.
14. Phadoongsidhi M. SCINDY: Logic Crosstalk Delay Fault Simulation in Sequential Circuits / M. Phadoongsidhi, K. K. Saluja // International Conference on VLSI Design, Kolkata, 3–7 January 2005 : proceedings. – Los Alamitos : IEEE, 2005. – P. 820–823.
15. Скобцов Ю. А. Эволюционные вычисления : учебное пособие / Ю. А. Скобцов, Д. В. Сперанский. – М. : Национальный Открытый Университет «ИНТУИТ», 2015. – 331 с.
16. Skobtsov Yu. A. Evolutionary test generation methods for digital devices / Yu. A. Skobtsov, V. Yu. Skobtsov // Design of Digital Systems and Devices / [eds.: M.Adamski et al.]. – Berlin: Springer-Verlag, 2011. – P. 331–361. – (Lecture Notes in Electrical Engineering, Vol. 79).
17. Шалыто А. А. Автоматное программирование / А. А. Шалыто, Н. И. Поликарпова. – Спб. : Питер, 2009. – 176 с.

Article was submitted 30.04.2015.
After revision 26.06.2015.

Скобцов Ю. О.[1], Скобцов В. Ю.[2], Шалыто А. А.[3]

[1]Д-р техн. наук, професор, зав.кафедрою АСУ Донецького національного технічного університету, Донецьк, Україна

[2]Канд. техн. наук, доцент, провідний науковий співробітник лабораторії проблем захисту інформації Об'єднаного інституту проблем інформатики НАН Беларусі, Мінськ, Беларусь

[3]Д-р техн. наук, професор, зав. кафедри технології програмування Університету інформаційних технологій, механіки та оптики, Санкт-Петербург, Росія

ПОБУДОВА ТЕСТІВ І МОДЕЛЮВАННЯ ДЛЯ ПЕРЕХРЕСНИХ ПОШКОДЖЕНЬ

Розглянуто основні моделі перехресних несправностей (crosstalkfaults): 1) індуковані позитивні і негативні імпульси, 2) індуковані затримки. Мета роботи – підвищення ефективності методів побудови перевіряльних тестів цифрових систем на основі еволюційного підходу і моделей неконстантних несправностей. Формалізована постановка задачі генерації перевіряльного тесту для одиночних перехресних несправностей – індукованих імпульсів і затримок. Показано, що ця задача зводиться до розв'язання системи логічних рівнянь в багатозначному алфавіті. Визначено 8-значний багатозначний алфавіт і функції основних вентилів у цьому алфавіті. Отримав розвиток метод моделювання перехресних несправностей в 8-значному алфавіті. На цій основі розроблено генетичний алгоритм побудови перевіряльних тестів для одиночних перехресних несправностей. Розглянуто задачу генерації перевіряльного тесту для несправності індукована затримка. Запропоновано генетичний алгоритм побудови перевіряльних тестів для несправності індукована затримка. Розроблено алгоритмічне та програмне забезпечення для генерації тестів для перехресних несправностей, яке дозволяє підвищити якість перевіряльних тестів за рахунок використання еволюційних методів та моделей неконстантних несправностей . Проведено апробацію розроблених методів на схемах міжнародних каталогів ISCAS85 , ISCAS89 , яка показала збільшення повноти тестів на 15%.

Ключові слова: побудова тестів, перехресні несправності, генетичний алгоритм, багатозначна логіка, моделювання пошкоджень.

Скобцов Ю. А.[1], Скобцов В. Ю.[2], Шалыто А. А.[3]

[1]Д-р техн. наук профессор, зав. кафедрой АСУ Донецкого национального технического университета, Донецк, Украина

[2]Канд. техн. наук, доцент, ведущий научный сотрудник лаборатории проблем защиты информации Объединенного института проблем информатики Национальной академии наук Беларуси, Минск, Беларусь

[3]Д-р техн. наук, профессор, зав. кафедрой технологии программирования Университета информационных технологий, механики и оптики, Санкт-Петербург, Россия

ПОСТРОЕНИЕ ТЕСТОВ И МОДЕЛИРОВАНИЕ ДЛЯ ПЕРЕКРЕСТНЫХ НЕИСПРАВНОСТЕЙ

Рассмотрены основные модели перекрестных неисправностей (crosstalkfaults): 1) индуцированные положительные и отрицательные импульсы, 2) индуцированные задержки. Цель работы – повышение эффективности методов построения проверяющих тестов цифровых систем на основе эволюционного подхода и моделей неконстантных неисправностей. Формализована постановка задачи генерации проверяющего теста для одиночных перекрестных неисправностей – индуцированных импульсов и задержек. Показано, что эта задача сводится к решению системы логических уравнений в многозначном алфавите. Определен 8-значный многозначный алфавит и функции основных вентилей в этом алфавите. Получил развитие метод моделирования перекрестных неисправностей в 8-значном алфавите. На этой основе разработан генетический алгоритм построения проверяющих тестов для одиночных перекрестных неисправностей. Рассмотрена задача генерации проверяющего теста для неисправности индуцированная задержка Предложен генетический алгоритм построения проверяющих тестов для неисправности индуцированная задержка со многими линиями-.агрессорами. Разработано алгоритмическое и программное обеспечение для генерации тестов для перекрестных неисправностей, которое позволяет повысить качество проверяющих тестов за счёт использования эволюционных методов. Проведена апробация разработанных методов на схемах международных каталогов ISCAS85, ISCAS89, которая показала увеличение полноты тестов на 15%.

Ключевые слова: генерация тестов, перекрестные неисправности, генетический алгоритм, многозначная логика, моделирование неисправностей.

REFERENCES

1. Skobtsov Yu. A., Spieransy D. V., Skobtsov V. Yu. Modelirovanie, testirovaniei diagnostika tsifrovyx ustroystv. Moscow, National Open University «INTUIT», 2012, 439 p.
2. Rubio A., Itazaki N., Xu X., Kinoshita K. An approach to the analysis and detection of crosstalk faults in digital VLSI circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1994, Vol.13, No. 3, pp. 387–394.
3. Chen W. Y., Gupta S. K., Breuer M. A. Analytic Models for Crosstalk Delay and Pulse Analysis under Non-Ideal Inputs, *International Test Conference, Washington, Nov. 1997 : proceedings*. Los Alamitos, IEEE, 1997, pp. 809–818. DOI:10.1109/TEST.1997.639695.
4. Chen W. Y., Gupta S. K., Breuer M. A. Test generation for Cross-Induced Delay, *International Test Conference, Atlantic City, 28–30 September 1999 : proceedings*. Los Alamitos, IEEE, 1997, pp. 191–200. DOI:10.1109/TEST.1999.805609
5. Chen W. Y., Gupta S. K., Breuer M. A. Test generation for Cross-Induced Faults: Framework and computational results, *Journal of Electronic Testing: Theory and Applications*, 2002, Vol. 16, pp. 17–28.
6. Krstic A., Liou J.-J., Jiang Y.-M., Cheng K.-T. Delay Testing Considering Cross-Induced Effects, *International Test Conference, Baltimore, 30 October 01 November 2001 : proceedings*. Los Alamitos: IEEE, 2001, pp. 558–567. DOI:10.1109/TEST.2001.966674.
7. Bai X. Dey S., Krstic A. HyAC. A Hybrid Structural SAT Based ATPG for Crosstalk, *International Test Conference, Washington, 30 September – 2 October 2003 : proceedings. – Los Alamitos: IEEE, 2003*, pp. 112–121. DOI:10.1109/TEST.2003.1270831.
8. Arunachalam A., Arunachalam R. A Novel Algorithm for Testing Crosstalk Induced Delay Faults in VLSI Cicuits, *International Conference on VLSI Design, Kolkata, 3–7 January 2005 : proceedings*. Los Alamitos, IEEE, 2005, pp. 479–484.
9. Li H., Li X. Selection of Crosstalk-induced Faults in Enhanced Delay test, Journal of Electronic Testing: Theory and Applications, 2005, Vol. 21, No. 2, pp. 181–195.
10. Palit A. K., Duganapalli K. K., Anheier W. Test Pattern Generation for Crosstalk Faults in DSM chips using Mofified PODEM, Eectronics System integration Technology Conference, Greenwich, 1–4 September 2003 : proceedings. Los Alamitos, IEEE, 2003, pp. 393–398. DOI:10.1109/ESTC.2008.4684311.
11. Chun S. Kim Y., Yang M.-H., Kang S. XPDF-ATPG: An Efficient Test Pattern Generation for Crosstalk-Induced Faults, *Asian Test Symposium, Sapporo, 23–24 November 2008 : proceedings*. Los Alamitos, IEEE, 2008, pp. 83–88.
12. Ganeshpure K. P., Kundu S. On ATPG for Multiple Aggressor Crosstalk Faults in Presence of Gate Delays, *International Test Conference, Santa Clara, 23–25 October 2007 : proceedings*. Los Alamitos, IEEE, 2007, pp. 1–7.
13. Chary Sh., Bushnell M. L. Automatic path delay test generation for combined Resistive Vias Resistive bridges and Capacitive Crosstalk delay faults, *International conference on VLSI Design, Hyderabad, 3–7 January 2006*. Los Alamitos, IEEE, 2006, pp. 413–418.
14. Phadoongsidhi M., Saluja K. K. SCINDY: Logic Crosstalk Delay Fault Simulation in Sequential Circuits, International Conference on VLSI Design, Kolkata, 3–7 January 2005 : proceedings. Los Alamitos, IEEE, 2005, pp. 820–823.
15. Skobtsov Yu. A., Spieransky D. V. Evolutsionnye vychislenia. Moscow, National Open University «INTUIT», 2015, 331 p.
16. Skobtsov Yu. A., Skobtsov V. Yu. Evolutionary test generation methods for digital devices, Design of Digital Systems and Devices. [eds.: M.Adamski et al.]. Berlin, Springer-Verlag, 2011, pp. 331–361. – (Lecture Notes in Electrical Engineering, Vol. 79).
17. Shalyto A. A., Polikarpova N. I. Avtomatnoe programmirovanie. Sankt-PeterburgPiter, 2009, 176 p.