https://doi.org/10.15407csc.2024.01.003
UDC 519.7

B.YE. RYTSAR, Doctor of Technical Sciences, Professor, Department of Radioelectronic Devices Systems, Institute of Telecommunications, Radioelectronics and Electronic Engineering,
L'viv polytechnic National University, Ukraine,
Bandera str., 12, L'viv, 79013, Ukraine,
ORCID: https://orcid.org/0000-0002-2929-2954,
bohdanrytsar@gmail.com

A SIMPLE STUCK-AT-FAULTS DETECTION METHOD IN DIGITAL COMBINATIONAL CIRCUITS. II

This article proposes the improved method for detecting (diagnosing) stuck-at-faults (0/1) in PIPO-type digital combinational circuits described by a system of logical functions. Compared to already known methods and algorithms, the presented approach is characterized by a simpler implementation of the search for vectors of the test codes for detection of such malfunctions at arbitrary points of a logic circuit with many outputs due to the usage of several simple numerical set-theoretic operations and procedures. The given examples prove the advantages of the proposed method.

Keywords: *digital PIPO-type combinational circuits, stuck-at-faults detection, vectors of the test codes, numerical set-theoretic approach, operations and procedures.*

Introduction

This article is a continuation of the work [1]. It is related to the application of the method of detection of stuck-at-faults (0/1) type, proposed by the author, in digital combinational circuits of the PIPO type described by the system of functions $F(X) = \{f_1(X), f_2(X), \dots, f_5(X)\}$, $X = \{x_1, x_2, \dots, x_n\}$. In general case, the system $F(X)$ of complete functions in the set-theoretic form is represented by the system of perfect STFs $\{Y_i^1\}$, $i = 1, 2, \dots, s$, of the form [2]:

$$\begin{cases} Y_1^1 = \{m_{11}, m_{12}, ..., m_{1k_1}\}^1, \\ Y_2^1 = \{m_{21}, m_{22}, ..., m_{2k_2}\}^1, \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ Y_s^1 = \{m_{s1}, m_{s2}, ..., m_{sk_s}\}^1, \end{cases} \qquad (1)$$

where m_{ij}, $i = 1,2, \dots, s$, $j = 1, 2, \dots, (2^n - k_i)$ are numerical minterms of the i-th function f_i of the system (1).

The problem of finding the vectors of the test codes in digital PIPO-type combinational devices is complicated due to the fact that a fault of stuck-at-faults (0/1) type can be common for several functions of the system $F(X)$ at the same time. Such fault can be detected in two ways similarly to the minimization of the system of functions [2]: 1) independently — if the detection method [1] is applied to each function separately, and the result is evaluated by taking into account the common test codes of the functions; 2) in a compatible way — if the search for the test codes is performed simultaneously for the so-called system minterms [2] that have system function indexes, and the result is

Цитування: Rytsar B.Ye. A simple Stuck-at-faults Detection Method in Digital Combinational Circuits. II. *Control Systems and Computers*, 2024, 1, 3—17. https://doi.org/10.15407/csc.2024.01.003

evaluated on the basis of the test codes by taking into account the function indexes. In the first case, the search for the test codes is performed using the minterms of the given system (1), and in the second case — using the system minterms that composes the set

$$Y^1_{1,2,...,s} = \{m_{s_1}, m_{s_2}, ..., m_{s_k}\}^1, \qquad (2)$$

where $m_{s_1}, m_{s_2}, ..., m_{s_k}$ are system minterms of the given system $\bar{F}(X)$, and the index of the i-th system minterm $s_i \in \{1, 2, …, s\}$. In SOP, the elements of the set $Y^1_{1,2,...,s}$ will be system conjuncterms of the different ranks, accordingly. The positive and negative sides of both approaches will be considered further on the examples for implementation of the proposed method.

Known algorithms and methods for diagnosing stuck-at-faults (0/1) type in such systems are mainly based on an analytical approach (in particular, using the Shannon function) and modeling of the process of searching for test codes [3—19], which limits their practical implementation especially as the increased number of system variables and functions.

The article proposes a new method of detecting (diagnosing) stuck-at-faults (0/1) in digital PIPO-type combinational circuits, which is based on the author's improved numerical set-theoretic approach [1] characterized by a relatively simpler practical implementation.

Theoretical Part

As stated in [1], the peculiarity of the method is that the location and type of stuck-at-faults (s-a-0 or s-a-1) in an arbitrary combinational circuit is determined using several simple ST operations on the binary minterms of the given function f. The search for a vector of the test codes determining the location and type of damage at a certain circuit level is performed for a conditionally damaged function f. At the 0-level of the circuit the "failure" vector is searched from the set of input variables, and at higher levels — from the set of those variables that occur at the intersection of the conventionally drawn (dashed) line with the signal transmission lines in the circuit under study (*see* Fig.1, 2, 3 in [1]). In these sets, a stuck-at-faults (0/1) "failure" is artificially created for each variable. At the same time, if in the j-th binary position of the i-th variable x_i to simulate "failure" of the s-a-0 type, i.e. $x_i/0$ (the operator $\overset{0\to1}{\Rightarrow}$ replaces the value of the variable x_i from log 0 to log 1), then the perfect STF Y^1 will be transformed into STF $Y^1_{0_j\to1_j}$, and if in j-th binary position of the i-th variable x_i to simulate "failure" of type s-a-1, i.e. $x_i/1$ (the operator $\overset{1\to0}{\Rightarrow}$ replaces the value of the variable x_i from log 1 to log 0), then the perfect STF Y^1 will be transformed into STF $Y^1_{1_j\to0_j}$. The perfect STF Y^1 and its «failured» variant are simplified in a polynomial format due to the elimination of the identical pairs of elements in their set, forming the so-called pseudoperfect STFs $Y^1_{x_i/0}$ and $Y^1_{x_i/1}$, respectively:

$$Y^1_{x_i/0} \overset{0\to1}{\Rightarrow} \{Y^1, Y^1_{0_j\to1_j}\}^{\oplus}, \qquad (3)$$

$$Y^1_{x_i/1} \overset{1\to0}{\Rightarrow} \{Y^1, Y^1_{1_j\to0_j}\}^{\oplus}. \qquad (4)$$

On the sets $Y^1_{x_i/0}$ (3) and $Y^1_{x_i/1}$ (4), it is possible to construct the truth tables for "failured" functions $f_{x_i/0}$ and $f_{x_i/1}$ in order to compare them with the truth table of the given function f. Table 1 illus-

Table 1

"10"	$x_1 x_2 x_3$	f	$f_{x_1/\sim}$		$f_{x_2/\sim}$		$f_{x_3/\sim}$	
			$x_1/0$	$x_1/1$	$x_2/0$	$x_2/1$	$x_3/0$	$x_3/1$
0	0 0 0	f_0	f_0	f_4	f_0	f_2	f_0	f_1
1	0 0 1	f_1	f_1	f_5	f_1	f_3	f_0	f_1
2	0 1 0	f_2	f_2	f_6	f_0	f_2	f_2	f_3
3	0 1 1	f_3	f_3	f_7	f_1	f_3	f_2	f_3
4	1 0 0	f_4	f_0	f_4	f_4	f_6	f_4	f_5
5	1 0 1	f_5	f_1	f_5	f_5	f_7	f_4	f_5
6	1 1 0	f_6	f_2	f_6	f_4	f_6	f_6	f_7
7	1 1 1	f_7	f_3	f_7	f_5	f_7	f_6	f_7

trates the case of an arbitrary function $f(x_1, x_2, x_3)$ as follows: if on some set of variables (*tupl* $\langle x_1, x_2, x_3\rangle$, $x_i \in \{0,1\}$)) their values differ from each other, i.e. $f_{x_i/\sim} \neq f$, where instead of a tilde (~) there can be either a log. 0 or log.1, it means that this set is the desired test code. For example (*see* Table 1), (001) is a test code for $f_{x_1/1}$, because on this input set at the output of the scheme we have $f = f_5$, but not $f = f_1$, that is, here the value x_1=1 is false. Similarly, for $f_{x_2/1}$ the value x_2=1 is false, because $f_3 \neq f_1$, and for $f_{x_3/0}$ false is x_3=0, because $f_0 \neq f_1$.

In [1], set-theoretic intersection and difference operations were used to determine the test codes. We will show that the test codes, with the help of which the place of failureat an arbitrary level of separation and the fault of type stuck-at-faults — s-a-0 or s-a-1 are determined in the studied scheme, can be obtained by a simpler procedure, compared to [1]. It is due to the fact that the obtained minterms of pseudoperfect STFs $Y^1_{x_i/0}$ (3) and $Y^1_{x_i/1}$ (4) are then compared in polynomial format with the minterms of the perfect STF Y^1 of function f, forming, as a result of the simplification of their sets $\{Y^1_{x_i/\sim}, Y^1\}^{\oplus}$, the desired sets of test codes:

$$\begin{Bmatrix} Y^1_{x_i/0} \\ Y^1 \end{Bmatrix}^{\oplus} \Rightarrow \begin{cases} C^1_{x_i/0} \\ C^0_{x_i/0} \end{cases}, \qquad (5)$$

$$\begin{Bmatrix} Y^1_{x_i/1} \\ Y^1 \end{Bmatrix}^{\oplus} \Rightarrow \begin{cases} C^1_{x_i/1} \\ C^0_{x_i/1} \end{cases}, \qquad (6)$$

where $C^1_{x_i/0}$, $C^1_{x_i/1}$ and $C^0_{x_i/0}$, $C^0_{x_i/1}$ — sets of the test codes, meaning , sets of sets of variables in which the i-th variable x_i of type s-a-0 (5), i.e. $x_i/0$, and s-a-1 (6), i.e. $x_i/1$, is failured ; the superscript in these sets indicates that as a result of the failure there was a movement of one or another minterm from the set Y^1 to the set $Y^0_{x_i/\sim}$, or from the set Y^0 to the set $Y^1_{x_i/\sim}$.

In order to illustrate the proposed approach, we consider the example 1 from [1] for a function $f(x_1, x_2, x_3)$ that has a perfect STF $Y^1 = \{1, 5, 6, 7\}^1$.

Let the fault s-a-0 acts on the 0-level of the circuit, for example, at the input x_1, i.e. $x_1/0$. Then we get a pseudoperfect STF (3), namely:

$$Y^1_{x_1/0} \overset{0\to1}{\Rightarrow} \begin{Bmatrix} (001), \cancel{(101)}, \cancel{(110)}, \cancel{(111)} \\ (101), \cancel{(101)}, \cancel{(110)}, \cancel{(111)} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \{(001), (101)\}^1.$$

To obtain a set of test codes for this case, we apply the procedure (5):

$$\begin{Bmatrix} Y^1_{x_1/0} \\ Y^1 \end{Bmatrix}^{\oplus} = \begin{Bmatrix} \{\cancel{(001)}, \cancel{(101)}\} \\ \{\cancel{(001)}, \cancel{(101)}, (110), (111)\} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \varnothing \\ \{(110), (111)\}^0_{x_1/0} \end{cases},$$

where $\{(110),(111)\}^0_{x_1/0}$ — set of elements of test codes $C^0_{x_1/0}$, which as a result of failure $x_1/0$ moved from set Y^1 to set $Y^0_{x_1/0}$ and acquired the value of log 0, that is indicated by the superscript (*see* Table 1 in [1]). This means, that when acting on the input of the scheme of codes (110) and (111), at its output we will have f = 0 instead of f = 1, which does not correspond to a perfect STF Y^1.

For the case of s-a-1 failure , for example, at the input x_2, i.e. $x_2/1$, we obtain (4):

$$Y^1_{x_2/1} \overset{1\to0}{\Rightarrow} \begin{Bmatrix} \cancel{(001)}, \cancel{(101)}, (110), (111) \\ \cancel{(001)}, \cancel{(101)}, (100), (101) \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \{(100), (101), (110), (111)\}^1.$$

After applying the procedure (6), we will get the desired test codes:

$$\begin{Bmatrix} Y^1_{x_2/1} \\ Y^1 \end{Bmatrix}^{\oplus} = \begin{Bmatrix} \{(100), \cancel{(101)}, \cancel{(110)}, \cancel{(111)}\} \\ \{(001), \cancel{(101)}, \cancel{(110)}, \cancel{(111)}\} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} (100)^1_{x_2/1} \\ (001)^0_{x_2/1} \end{cases}.$$

Here the minterm (100) has moved from the set Y^0 to the set $Y^1_{x_2/1}$, and the minterm (001) has moved from the set Y^1 to the set $Y^0_{x_2/1}$.

We will place the determined vectors of the test codes, that establish the type and place of fault of type stuck-at-faults (0/1), in a separate table, which is a kind of table of possible errors in the considered circuit.

Practical Part

As mentioned above, in the studied scheme, that is described by the system of the functions (1), the test code vectors for the installation of the type and location of the stuck-at-faults (0/1) failure can be determined by two approaches: independently of each system function separately or jointly for the system in general. If in the first case the given system $F(X)$ of complete functions is considered as a system (1) of perfect STFs $\{Y_i^1\}$, $i = 1, 2, \ldots, s$, then in the second case it is considered as a set (2) of system conjuncterms $\{Y_I^1\}$, $I = 1, 2, \ldots, s$.

Let us illustrate the proposed stuck-at-faults (0/1) detection method on the presented examples.

Example 1. Determine the vectors of test codes to detect possible fault of the stuck-at-faults (0/1) type in the system of functions (*borrowed from* [7, p. 87])

$$\begin{cases} f_1 = \bar{x}_1 x_2 \vee x_1 x_2 x_3 \\ f_2 = x_1 \bar{x}_2 \vee x_1 x_2 x_3 \end{cases}$$

implemented by the logic circuit and shown in Fig. 1; for the comparison purpose, perform the search of vectors using two approaches mentioned above.

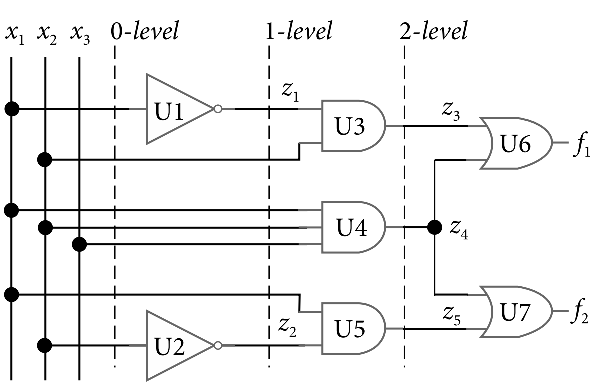

Fig. 1

Solution. In the case of independent determination of the test codes, the given system $F(X) = \{f_1(X), f_2(X)\}$, $X=\{x_1, x_2, x_3\}$, will be represented by the system (1) of perfect STFs $\{Y_1^1, Y_2^1\}$:

$$\begin{cases} Y_1^1 = \{(01-), (111)\}^1 \equiv \{(2,3), (7)\}^1 \\ Y_2^1 = \{(10-), (111)\}^1 \equiv \{(4,5), (7)\}^1 \end{cases},$$

and in the case of joint determination (2) — by the set of system minterms $Y_{1,2}^1 = \{(2,3)_1, (4,5)_2, (7)_{1,2}\}^1$.

In the first case, we will form pseudoperfect STFs $Y^1_{x_i/\sim}$ (3) and (4) on the 0-level of the scheme for each function of the given system, and on their basis we will determine the vectors of the test codes — set (5) $C^1_{x_i/\sim}$ and set (6) $C^0_{x_i/\sim}$. So, we obtain the following sets for the function f_1:

$$Y^1_{1(x_1/0)} \overset{0\to1}{\Rightarrow} \begin{Bmatrix} (010), (011), \cancel{(111)} \\ (110), (111), \cancel{(111)} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(010)}, \cancel{(011)}, (110), \cancel{(111)} \\ \cancel{(010)}, \cancel{(011)}, \cancel{(111)} \end{Bmatrix}^{\oplus} \Rightarrow \begin{cases} \{110\}^1_{x_1/0} \\ \varnothing \end{cases},$$

$$Y^1_{1(x_1/1)} \overset{1\to0}{\Rightarrow} \begin{Bmatrix} \cancel{(010)}, \cancel{(011)}, (111) \\ \cancel{(010)}, \cancel{(011)}, (011) \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(011)}, \cancel{(111)} \\ (010), \cancel{(011)}, \cancel{(111)} \end{Bmatrix}^{\oplus} \Rightarrow \begin{cases} \varnothing \\ \{010\}^0_{x_1/1} \end{cases};$$

$$Y^1_{1(x_2/0)} \overset{0\to1}{\Rightarrow} \begin{Bmatrix} \cancel{(010), (011), (111)} \\ \cancel{(010), (011), (111)} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \varnothing \\ (010), (011), (111) \end{Bmatrix}^{\oplus} \Rightarrow \begin{cases} \varnothing \\ \{(010), (011), (111)\}^0_{x_2/0} \end{cases},$$

$$Y^1_{1(x_2/1)} \overset{1\to0}{\Rightarrow} \begin{Bmatrix} (010), (011), (111) \\ (000), (001), (101) \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} (000), (001), \cancel{(010)}, \cancel{(011)}, (101), \cancel{(111)} \\ \cancel{(010), (011), (111)} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(000), (001), (101)\}^1_{x_2/1}; \\ \varnothing \end{cases}$$

$$Y^1_{1(x_3/0)} \overset{0\to1}{\Rightarrow} \left\{\begin{matrix} (010), \cancel{(011)}, \cancel{(111)} \\ (011), \cancel{(011)}, \cancel{(111)} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(010)}, \cancel{(011)} \\ \cancel{(010)}, \cancel{(011)}, (111) \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \varnothing \\ \{111\}^0_{x_3/0} \end{matrix}\right.,$$

$$Y^1_{1(x_3/1)} \overset{1\to0}{\Rightarrow} \left\{\begin{matrix} \cancel{(010)}, (011), (111) \\ \cancel{(010)}, (010), (110) \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(010)}, \cancel{(011)}, (110), \cancel{(111)} \\ \cancel{(010), (011), (111)} \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \{110\}^1_{x_3/1} \\ \varnothing \end{matrix}\right. .$$

Table 2 on the 0-level of the scheme contains the truth table of the given function f_1, as well as the truth tables of its "failured" variants $f_{1x_1/\sim}$, $f_{1x_2/\sim}$ and $f_{1x_3/\sim}$, where their values that are different from the values of the function f_1 are highlighted in bold.

We put the defined test codes for the function f_1 in Table 3.

We get the following sets for the function f_2:

$$Y^1_{2(x_1/0)} \overset{0\to1}{\Rightarrow} \left\{\begin{matrix} \cancel{(100), (101), (111)} \\ \cancel{(100), (101), (111)} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \varnothing \\ (100), (101), (111) \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \varnothing \\ \{(100), (101), (111)\}^0_{x_1/0} \end{matrix}\right.,$$

$$Y^1_{2(x_1/1)} \overset{1\to0}{\Rightarrow} \left\{\begin{matrix} (100), (101), (111) \\ (000), (001), (011) \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} (000), (001), (011), \cancel{(100), (101), (111)} \\ \cancel{(100), (101), (111)} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \{(000), (001), (011)\}^1_{x_1/1} \\ \varnothing \end{matrix}\right.,$$

$$Y^1_{2(x_2/0)} \overset{0\to1}{\Rightarrow} \left\{\begin{matrix} (100), (101), \cancel{(111)} \\ (110), (111), \cancel{(111)} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(100)}, \cancel{(101)}, (110), \cancel{(111)} \\ \cancel{(100), (101), (111)} \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \{110\}^1_{x_2/0} \\ \varnothing \end{matrix}\right.,$$

$$Y^1_{2(x_2/1)} \overset{1\to0}{\Rightarrow} \left\{\begin{matrix} \cancel{(100)}, \cancel{(101)}, (111) \\ \cancel{(100)}, \cancel{(101)}, (101) \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(101)}, \cancel{(111)} \\ (100), \cancel{(101)}, \cancel{(111)} \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \varnothing \\ \{100\}^0_{x_2/1} \end{matrix}\right.,$$

$$Y^1_{2(x_3/0)} \overset{0\to1}{\Rightarrow} \left\{\begin{matrix} (100), \cancel{(101)}, \cancel{(111)} \\ (101), \cancel{(101)}, \cancel{(111)} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(100)}, \cancel{(101)} \\ \cancel{(100)}, \cancel{(101)}, (111) \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \varnothing \\ \{111\}^0_{x_3/0} \end{matrix}\right.,$$

$$Y^1_{2(x_3/1)} \overset{1\to0}{\Rightarrow} \left\{\begin{matrix} \cancel{(100)}, (101), (111) \\ \cancel{(100)}, (100), (110) \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(100)}, \cancel{(101)}, (110), \cancel{(111)} \\ \cancel{(100), (101), (111)} \end{matrix}\right\}^{\oplus} \Rightarrow \left\{\begin{matrix} \{110\}^1_{x_3/1} \\ \varnothing \end{matrix}\right.,$$

Table 2

«10»	$x_1x_2x_3$	f_1	$f_{1x_1/\sim}$		$f_{1x_2/\sim}$		$f_{1x_3/\sim}$	
			$x_1/0$	$x_1/1$	$x_2/0$	$x_2/1$	$x_3/0$	$x_3/1$
0	0 0 0	0	0	0	0	**1**	0	0
1	0 0 1	0	0	0	0	**1**	0	0
2	0 1 0	1	1	**0**	**0**	1	1	1
3	0 1 1	1	1	1	**0**	1	1	1
4	1 0 0	0	0	0	0	0	0	0
5	1 0 1	0	0	0	0	**1**	0	0
6	1 1 0	0	**1**	0	0	0	0	**1**
7	1 1 1	1	1	1	**0**	1	**0**	1

Table 3

Stuck-at-fault	$x_1/\sim$	$x_2/\sim$	$x_3/\sim$
s-a-0	$(110)^1$	$\begin{pmatrix} 010 \\ 011 \\ 111 \end{pmatrix}^0$	$(111)^0$
s-a-1	$(010)^0$	$\begin{pmatrix} 000 \\ 001 \\ 101 \end{pmatrix}^1$	$(110)^1$

Table 4 for the function f_2 of the given system is constructed by analogy with Table 2.

So, for the function f_2 we have test codes placed in Table 5:

Comparing the data of Tables 3 and 5, it can be noted that these data are consistent with the data of Tables 2 and 4, respectively. For example, if the code (010) is active at the input (*see* Tables 2 and 4), then at the outputs of the undamaged circuit we have f_1=1 and f_2=0, instead, at this code in the failured circuit we will have $f_{1x_1/1}=0$, $f_{1x_2/0}=0$ and f_2=0, as shown in Tables 3 and 5.

To determine the vectors of test codes in the second case, we will consider the set (2) of system minterms $Y_{1,2}^1=\{2_1, 3_1, 4_2, 5_2, 7_{1,2}\}^1$ ($Y_{1,2}^0=\{0_{1,2}, 1_{1,2}, 2_2, 3_2, 4_1, 5_1, 6_{1,2}\}^0$). Here, unlike the previous case, only those pairs of elements that have not only the same values, but also the same indices will be simplified in the polynomial format. At the same time, only the system minterms of the set $Y_{1,2}^1$ will be eliminated from pairs of identical elements with different function indices, since during the procedure (3) or (4) they have already been replaced by the elements of the corresponding pseudoperfect STFs. We will cross out such elements with a line with the opposite slope.

Table 4

"10"	$x_1x_2x_3$	f_2	$f_{2x_1/\sim}$		$f_{2x_2/\sim}$		$f_{2x_3/\sim}$	
			$x_1/0$	$x_1/1$	$x_2/0$	$x_2/1$	$x_3/0$	$x_3/1$
0	0 0 0	0	0	**1**	0	0	0	0
1	0 0 1	0	0	**1**	0	0	0	0
2	0 1 0	0	0	0	0	0	0	0
3	0 1 1	0	0	**1**	0	0	0	0
4	1 0 0	1	**0**	1	1	**0**	1	1
5	1 0 1	1	**0**	1	1	1	1	1
6	1 1 0	0	0	0	**1**	0	0	**1**
7	1 1 1	1	**0**	1	1	1	**0**	1

Table 5

Stuck-at-fault	$x_1/\sim$	$x_2/\sim$	$x_3/\sim$
s-a-0	$\begin{pmatrix}100\\101\\111\end{pmatrix}^0$	$(110)^1$	$(111)^0$
s-a-1	$\begin{pmatrix}000\\001\\011\end{pmatrix}^1$	$(100)^0$	$(110)^1$

So, in the second case, the vectors of the test codes on the 0-levels of the scheme are obtained as follows:

$$Y_{1,2(x_1/0)}^1 \overset{0\to 1}{\Rightarrow} \begin{Bmatrix} (010)_1, (011)_1, \sout{(100)_2, (101)_2, (111)}_{1,2} \\ (110)_1, (111)_1, \sout{(100)_2, (101)_2, (111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(010)}_1, \cancel{(011)}_1, (110)_1, (111)_1 \\ \cancel{(010)}_1, \cancel{(011)}_1, (100)_2, (101)_2, \cancel{(111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(110)_1, (111)_1\}_{x_1/0}^1 \\ \{(100)_2, (101)_2\}_{x_1/0}^0 \end{cases},$$

where the system minterm $(111)_{1,2}$ is eliminated from consideration because it was replaced by the element $(111)_1$ of the pseudoperfect STF during the procedure (3);

$$Y_{1,2(x_1/1)}^1 \overset{1\to 0}{\Rightarrow} \begin{Bmatrix} \cancel{(010)}_1, \cancel{(011)}_1, (100)_2, (101)_2, (111)_{1,2} \\ \cancel{(010)}_1, \cancel{(011)}_1, (000)_2, (001)_2, (011)_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} (000)_2, (001)_2, (011)_{1,2}, \sout{(100)_2, (101)_2, (111)}_{1,2} \\ (010)_1, \bcancel{(011)}_1, \sout{(100)_2, (101)_2, (111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(000)_2, (001)_2, (011)_{1,2}\}_{x_1/1}^1 \\ \{(010)_1\}_{x_1/1}^0 \end{cases},$$

$$Y_{1,2(x_2/0)}^1 \overset{0\to 1}{\Rightarrow} \begin{Bmatrix} \cancel{(010)}_1, \cancel{(011)}_1, (100)_2, (101)_2, \cancel{(111)}_{1,2} \\ \cancel{(010)}_1, \cancel{(011)}_1, (110)_2, (111)_2, \cancel{(111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(100)}_2, \cancel{(101)}_2, (110)_2, (111)_2 \\ (010)_1, (011)_1, \cancel{(100)}_2, \cancel{(101)}_2, \bcancel{(111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(110)_2, (111)_2\}_{x_2/0}^1 \\ \{(010)_1, (011)_1\}_{x_2/0}^0 \end{cases},$$

$$Y^1_{1,2(x_2/1)} \overset{1\to 0}{\Rightarrow} \begin{Bmatrix} (010)_1, (011)_1, \cancel{(100)}_2, \cancel{(101)}_2, (111)_{1,2} \\ (000)_1, (001)_1, \cancel{(100)}_2, \cancel{(101)}_2, (101)_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} (000)_1, (001)_1, \cancel{(010)}_1, \cancel{(011)}_1, (101)_{1,2}, \cancel{(111)}_{1,2} \\ \cancel{(010)}_1, \cancel{(011)}_1, (100)_2, \cancel{(101)}_2, \cancel{(111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(000)_1, (001)_1, (101)_{1,2}\}^1_{x_2/1} \\ \{(100)_2\}^0_{x_2/1} \end{cases},$$

$$Y^1_{1,2(x_3/0)} \overset{0\to 1}{\Rightarrow} \begin{Bmatrix} (010)_1, \cancel{(011)}_1, (100)_2, \cancel{(101)}_2, \cancel{(111)}_{1,2} \\ (011)_1, \cancel{(011)}_1, (101)_2, \cancel{(101)}_2, \cancel{(111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(010)_1, (011)_1, (100)_2, (101)_2} \\ \cancel{(010)_1, (011)_1, (100)_2, (101)_2}, (111)_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \varnothing \\ \{(111)_{1,2}\}^0_{x_3/0} \end{cases},$$

$$Y^1_{1,2(x_3/1)} \overset{1\to 0}{\Rightarrow} \begin{Bmatrix} \cancel{(010)}_1, (011)_1, \cancel{(100)}_2, (101)_2, (111)_{1,2} \\ \cancel{(010)}_1, (010)_1, \cancel{(100)}_2, (100)_2, (110)_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(010)_1, (011)_1, (100)_2, (101)_2}, (110)_{1,2}, \cancel{(111)}_{1,2} \\ \cancel{(010)_1, (011)_1, (100)_2, (101)_2}, \cancel{(111)}_{1,2} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(110)_{1,2}\}^1_{x_3/1} \\ \varnothing \end{cases}.$$

Table 6 contains the truth tables of the given system and its "failured" variants, where different from $f_{1,2}$ the values of the functions $f_{1,2x_1/\sim}$, $f_{1,2x_2/\sim}$, and $f_{1,2x_3/\sim}$ are highlighted in bold.

Table 7 summarizes the vectors of the test codes for the functions of the given system, which are consistent with the data of the corresponding Tables 3, 5 and 6.

It is possible to determine the vectors of the test codes from Table 6 for system functions on the 0-level of the scheme as follows. Let, for example, the code (010) acts on the circuit input. But at its outputs, instead of $f_1 = 1$ and $f_2 = 0$, we have $f_1 = f_2 = 0$. Then two situations are possible here: either s-a-1 failure occurs at the input x_1, i.e. $x_1/1$, or s-a-0 failure occurs at the input x_2, i.e. $x_2/0$ (*see* the data in Tables 6 and 7).

Based on the above, we note that the second way of determining the vectors of test codes, compared to the first way, is shorter in terms of the algorithm' implementation steps.

Let us now consider the definition of the test codes on the 1-level of the scheme (*see* Fig. 1). It is represented by the system

$$\begin{cases} f_1(z_1, z_2, x_3) = z_1\bar{z}_2 \vee \bar{z}_1\bar{z}_2 x_3 \\ f_2(z_1, z_2, x_3) = \bar{z}_1 z_2 \vee \bar{z}_1\bar{z}_2 x_3 \end{cases},$$

where $z_1 = \bar{x}_1$, and $z_2 = \bar{x}_2$, and in set-theoretic form — the system

$$\begin{cases} Y_1^1 = \{(10-), (001)\}^1 \equiv \{1, 4, 5\}^1 \\ Y_2^1 = \{(01-), (001)\}^1 \equiv \{1, 2, 3\}^1 \end{cases}$$

Table 6

"10"	$x_1x_2x_3$	$f_{1,2}$	$f_{1,2x_1/\sim}$		$f_{1,2x_2/\sim}$		$f_{1,2x_3/\sim}$	
			$x_1/0$	$x_1/1$	$x_2/0$	$x_2/1$	$x_3/0$	$x_3/1$
0	0 0 0	0	0	$\mathbf{1_2}$	0	$\mathbf{1_1}$	0	0
1	0 0 1	0	0	$\mathbf{1_2}$	0	$\mathbf{1_1}$	0	0
2	0 1 0	1_1	1_1	**0**	**0**	1_1	1_1	1_1
3	0 1 1	1_1	1_1	$\mathbf{1_{1,2}}$	**0**	1_1	1_1	1_1
4	1 0 0	1_2	**0**	1_2	1_2	**0**	1_2	1_2
5	1 0 1	1_2	**0**	1_2	1_2	$\mathbf{1_{1,2}}$	1_2	1_2
6	1 1 0	0	$\mathbf{1_1}$	0	$\mathbf{1_2}$	0	0	$\mathbf{1_{1,2}}$
7	1 1 1	$1_{1,2}$	$\mathbf{1_1}$	$1_{1,2}$	$\mathbf{1_2}$	$1_{1,2}$	**0**	$1_{1,2}$

Table 7

Stuck-at-fault in $f_{1,2}$	$x_1/\sim$	$x_2/\sim$	$x_3/\sim$
s-a-0	$\begin{pmatrix}100\\101\end{pmatrix}^0_{x_1/0}$ $\begin{pmatrix}110_1\\111_1\end{pmatrix}^1_{x_1/0}$	$\begin{pmatrix}010\\011\end{pmatrix}^0_{x_2/0}$ $\begin{pmatrix}110_2\\111_2\end{pmatrix}^1_{x_2/0}$	$(111)^0_{x_3/0}$
s-a-1	$(010)^0_{x_1/1}$ $\begin{pmatrix}000_2\\001_2\\011_{1,2}\end{pmatrix}^1_{x_1/1}$	$(100)^0_{x_2/1}$ $\begin{pmatrix}000_1\\001_1\\101_{1,2}\end{pmatrix}^1_{x_2/1}$	$(110_{1,2})^1_{x_3/1}$

and the set of system minterms $Y^1_{1,2} = \{(1)_{1,2}, (2,3)_2, (4,5)_1\}^1$. Tables 8 and 9 display test codes on the 1-level of the scheme for $Y^1_{1,2}$, determined using the procedures (3—6).

We have a system of functions on the 2-level of the scheme

$$\begin{cases} f_1(z_3, z_4, z_5) = z_3 \vee z_4 \\ f_2(z_3, z_4, z_5) = z_4 \vee z_5 \end{cases},$$

where $z_3=z_1x_2$, $z_4=x_1x_2x_3$, $z_5=x_1z_2$. In the set-theoretic form, it is a system of perfect STFs $\{Y_1^{\,1}, Y_2^{\,1}\}$:

Table 8

"10"	$z_1 z_2 x_3$	$f_{1,2}$	$f_{1,2z_1/\sim}$		$f_{1,2z_2/\sim}$		$f_{1,2x_3/\sim}$	
			$z_1/0$	$z_1/1$	$z_2/0$	$z_2/1$	$x_3/0$	$x_3/1$
0	0 0 0	0	0	$\mathbf{1_1}$	0	$\mathbf{1_2}$	0	$\mathbf{1_{1,2}}$
1	0 0 1	$1_{1,2}$	$1_{1,2}$	$\mathbf{1_1}$	$1_{1,2}$	$\mathbf{1_2}$	$\mathbf{0}$	$1_{1,2}$
2	0 1 0	1_2	1_2	$\mathbf{0}$	$\mathbf{0}$	1_2	1_2	1_2
3	0 1 1	1_2	1_2	$\mathbf{0}$	$\mathbf{1_{1,2}}$	1_2	1_2	1_2
4	1 0 0	1_1	$\mathbf{0}$	1_1	1_1	$\mathbf{0}$	1_1	1_1
5	1 0 1	1_1	$\mathbf{1_{1,2}}$	1_1	1_1	$\mathbf{0}$	1_1	1_1
6	1 1 0	0	$\mathbf{1_2}$	0	$\mathbf{1_1}$	0	0	0
7	1 1 1	0	$\mathbf{1_2}$	0	$\mathbf{1_1}$	0	0	0

Table 9

Stuck-at-fault in $f_{1,2}$	$z_1/\sim$	$z_2/\sim$	$x_3/\sim$
s-a-0	$(100)^0_{z_1/0}$ $\begin{pmatrix}101_{1,2}\\110_2\\111_2\end{pmatrix}^1_{z_1/0}$	$(010)^0_{z_2/0}$ $\begin{pmatrix}011_{1,2}\\110_1\\111_1\end{pmatrix}^1_{z_2/0}$	$(001)^0_{x_3/0}$
s-a-1	$\begin{pmatrix}010\\011\end{pmatrix}^0_{z_1/1}$ $\begin{pmatrix}000_1\\001_1\end{pmatrix}^1_{z_1/1}$	$\begin{pmatrix}100\\101\end{pmatrix}^0_{z_2/1}$ $\begin{pmatrix}000_2\\001_2\end{pmatrix}^1_{z_2/1}$	$(000_{1,2})^1_{x_3/1}$

Table 10

"10"	$z_3\, z_4\, z_5$	$f_{1,2}$	$f_{1,2z_3/\sim}$		$f_{1,2z_4/\sim}$		$f_{1,2x_5/\sim}$	
			$z_3/0$	$z_3/1$	$z_4/0$	$z_4/1$	$x_5/0$	$x_5/1$
0	0 0 0	0	0	$\mathbf{1_1}$	0	$\mathbf{1_{1,2}}$	0	$\mathbf{1_2}$
1	0 0 1	1_2	1_2	$\mathbf{1_{1,2}}$	1_2	$\mathbf{1_{1,2}}$	$\mathbf{0}$	1_2
2	0 1 0	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$\mathbf{0}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$
3	0 1 1	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$\mathbf{1_2}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$
4	1 0 0	1_1	$\mathbf{0}$	1_1	1_1	$\mathbf{1_{1,2}}$	1_1	$\mathbf{1_{1,2}}$
5	1 0 1	$1_{1,2}$	$\mathbf{1_2}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$\mathbf{1_1}$	$1_{1,2}$
6	1 1 0	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$\mathbf{1_1}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$
7	1 1 1	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$

Table 11

Stuck-at-fault in $f_{1,2}$	$z_3/\sim$	$z_4/\sim$	$x_5/\sim$
s-a-0	$(100)^0_{z_3/0}$ $(101_2)^1_{z_3/0}$	$(010)^0_{z_4/0}$ $\begin{pmatrix}011_2\\110_1\end{pmatrix}^1_{z_4/0}$	$(001)^0_{x_5/0}$ $(101_1)^1_{z_5/0}$
s-a-1	$\begin{pmatrix}000_1\\001_{1,2}\end{pmatrix}^1_{z_3/1}$	$\begin{pmatrix}000_{1,2}\\001_{1,2}\\100_{1,2}\end{pmatrix}^1_{z_4/1}$	$\begin{pmatrix}000_2\\100_{1,2}\end{pmatrix}^1_{x_5/1}$

Table 12

"10"	$x_1 x_2 x_3$	f_1	f_2	f_3
0	0 0 0	0	1	1
1	0 0 1	0	0	1
2	0 1 0	0	0	0
3	0 1 1	0	0	1
4	1 0 0	0	0	0
5	1 0 1	1	1	0
6	1 1 0	1	0	0
7	1 1 1	1	0	0

$$\begin{cases} Y_1^1 = \{(1--),(-1-)\}^1 \equiv \{2, 3, 4, 5, 6, 7\}^1 \\ Y_2^1 = \{(-1-),(--1)\}^1 \equiv \{1, 2, 3, 5, 6, 7\}^1 \end{cases},$$

and a set of system minterms — $Y_{1,2}^1 = \{(1)_2, (2, 3, 5, 6, 7)_{1,2}, (4)_1\}^1$. Tables 10 and 11, which are constructed similarly to Tables 6—9, contain all values of test code vectors on the 2-level of the scheme.

Answer. The algorithm for determining the vectors of the test codes for detecting stuck-at-faults (0/1) failure in the system of functions based on a set of system minterms $\{Y_I^1\}$, $I = 1, 2, \ldots, s$ is shorter in terms of implementation steps in comparison to the algorithm based on the system of perfect STFs $\{Y_i^1\}$, $i = 1, 2, \ldots, s$.

Example 2. Determine the vectors of test codes for the detection of stuck-at-faults (0/1) failure in a scheme that implements the system of functions $F(X) = \{f_1(X), f_2(X), f_3(X)\}$, $X = \{x_1, x_2, x_3\}$ on the PLA, which is described by the truth table (*see* Table 12) (*borrowed* from [20, p. 197]):

Solution. The given system $F(X)$, $X = \{x_1, x_2, x_3\}$, is represented by a system of perfect STFs

$$\begin{cases} Y_1^1 = \{5,6,7\}^1 \\ Y_2^1 = \{0,5\}^1 \\ Y_3^1 = \{0,1,3\}^1 \end{cases},$$

and a set of system minterms $Y_{1,2,3}^1 = \{0_{2,3}, 1_3, 3_3, 5_{1,2}, 6_1, 7_1\}^1$.

Since the given system must be implemented on the PLA, it is important to ensure the optimal complexity of such an implementation — the information capacity of the matrix structure of the scheme [2]. It is determined by the minimal number of (vertical and horizontal) lines of both matrices. In a given system, the number of lines of conjuncterms is equal to the number of system minterms, that are six. They can be reduced to four

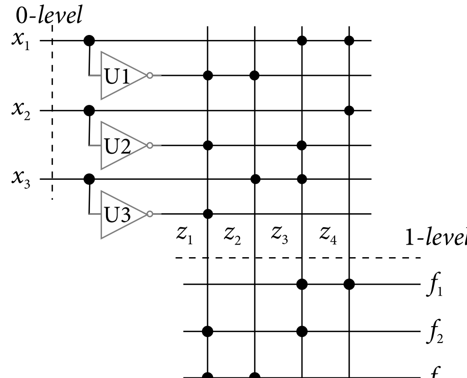

Fig. 2

if the decoupling method [2] is applied to the given system minterms:

$$Y_{1,2,3}^1 = \{000_{2,3}, 001_3, 011_3, 101_{1,2}, 110_1, 111_1\}^1 \overset{S}{\Rightarrow}$$

$$\overset{S}{\Rightarrow} \begin{bmatrix} 00-_{2,3} & 00-_3 & 01-_3 & 10-_{1,2} & \underline{11-_1} & \underline{11-_1} \\ 0-0_{2,3} & \underline{0-1_3} & \underline{0-1_3} & 1-1_{1,2} & 1-0_1 & 1-1_1 \\ -00_{2,3} & -01_3 & -11_3 & -01_{1,2} & -10_1 & -11_1 \end{bmatrix} \Rightarrow$$

$$\Rightarrow \{(000)_{2,3}, (0-1)_3, (101)_{1,2}, (11-)_1\}^1.$$

In Fig. 2 is shown the implementation of the given system on the PLA, taking into account the optimal information capacity, where the following designations of conjuncterm lines are introduced on the 1-level of the scheme:

$$z_1 = \bar{x}_1\bar{x}_2\bar{x}_3,\ z_2 = \bar{x}_1 x_3, \quad z_3 = x_1\bar{x}_2 x_3,\ z_4 = x_1 x_2. \tag{7}$$

According to (7), now we have the set of system minterms: $Y_{1,2,3}^1 = \{(0)_{2,3},(1,3)_3,(5)_{1,2},(6,7)_1\}^1$.

First, we determine the vectors of the test codes on the 0-level of the scheme, applying the procedures (3—6) to the binary system minterms of the set $Y_{1,2,3}^1$:

$$Y_{1,2,3(x_1/0)}^1 \overset{0\to 1}{\Rightarrow} \begin{Bmatrix} (000)_{2,3}, (001)_3, (011)_3, \cancel{(101)_{1,2}, (110)_1, (111)_1} \\ (100)_{2,3}, (101)_3, (111)_3, \cancel{(101)_{1,2}, (110)_1, (111)_1} \end{Bmatrix}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{Bmatrix} \cancel{(000)_{2,3}, (001)_3, (011)_3}, (100)_{2,3}, (101)_3, (111)_3 \\ \cancel{(000)_{2,3}, (001)_3, (011)_3}, \cancel{(101)_{1,2}}, (110)_1, \cancel{(111)_1} \end{Bmatrix}^{\oplus} \Rightarrow \begin{cases} \{(100)_{2,3}, (101)_3, (111)_3\}^1 \\ \{(110)_1\}^0 \end{cases},$$

$$Y^1_{1,2,3(x_1/1)} \overset{1\to 0}{\Rightarrow} \left\{\begin{array}{l} \cancel{(000)_{2,3}, (001)_3, (011)_3}, (101)_{1,2}, (110)_1, (111)_1 \\ \cancel{(000)_{2,3}, (001)_3, (011)_3}, (001)_{1,2}, (010)_1, (011)_1 \end{array}\right]^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{array}{l} (001)_{1,2}, (010)_1, (011)_1, \cancel{(101)_{1,2}, (110)_1, (111)_1} \\ (000)_{2,3}, \cancel{(001)}_3, \cancel{(011)}_3, \cancel{(101)_{1,2}, (110)_1, (111)_1} \end{array}\right\}^{\oplus} \Rightarrow \left\{\begin{array}{l} \{(001)_{1,2}, (010)_1, (011)_1\}^1 \\ \{(000)_{2,3}\}^0 \end{array}\right. ,$$

$$Y^1_{1,2,3(x_2/0)} \overset{0\to 1}{\Rightarrow} \left\{\begin{array}{l} (000)_{2,3}, (001)_3, \cancel{(011)}_3, (101)_{1,2}, \cancel{(110)}_1, \cancel{(111)}_1 \\ (010)_{2,3}, (011)_3, \cancel{(011)}_3, (111)_{1,2}, \cancel{(110)}_1, \cancel{(111)}_1 \end{array}\right]^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{array}{l} \cancel{(000)}_{2,3}, \cancel{(001)}_3, (010)_{2,3}, \cancel{(011)}_3, \cancel{(101)}_{1,2}, (111)_{1,2} \\ \cancel{(000)}_{2,3}, \cancel{(001)}_3, \cancel{(011)}_3, \cancel{(101)}_{1,2}, (110)_1, \cancel{(111)}_1 \end{array}\right\}^{\oplus} \Rightarrow \left\{\begin{array}{l} \{(010)_{2,3}, (111)_{1,2}\}^1 \\ \{(110)_1\}^0 \end{array}\right. ,$$

$$Y^1_{1,2,3(x_2/1)} \overset{1\to 0}{\Rightarrow} \left\{\begin{array}{l} \cancel{(000)}_{2,3}, \cancel{(001)}_3, (011)_3, \cancel{(101)}_{1,2}, (110)_1, (111)_1 \\ \cancel{(000)}_{2,3}, \cancel{(001)}_3, (001)_3, \cancel{(101)}_{1,2}, (100)_1, (101)_1 \end{array}\right]^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{array}{l} \cancel{(001)}_3, \cancel{(011)}_3, (100)_1, (101)_1, \cancel{(110)}_1, \cancel{(111)}_1 \\ (000)_{2,3}, \cancel{(001)}_3, \cancel{(011)}_3, \cancel{(101)}_{1,2}, \cancel{(110)}_1, \cancel{(111)}_1 \end{array}\right\}^{\oplus} \Rightarrow \left\{\begin{array}{l} \{(100)_1, (101)_1\}^1 \\ \{(000)_{2,3}\}^0 \end{array}\right. ,$$

$$Y^1_{1,2,3(x_3/0)} \overset{0\to 1}{\Rightarrow} \left\{\begin{array}{l} (000)_{2,3}, \cancel{(001)_3, (011)_3, (101)_{1,2}}, (110)_1, \cancel{(111)}_1 \\ (001)_{2,3}, \cancel{(001)_3, (011)_3, (101)_{1,2}}, (111)_1, \cancel{(111)}_1 \end{array}\right]^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{array}{l} \cancel{(000)}_{2,3}, (001)_{2,3}, \cancel{(110)}_1, \cancel{(111)}_1 \\ \cancel{(000)}_{2,3}, \cancel{(001)}_3, (011)_3, (101)_{1,2}, \cancel{(110)}_1, \cancel{(111)}_1 \end{array}\right\}^{\oplus} \Rightarrow \left\{\begin{array}{l} \{(001)_{2,3}\}^1 \\ \{(011)_3, (101)_{1,2}\}^0 \end{array}\right. ,$$

$$Y^1_{1,2,3(x_3/1)} \overset{1\to 0}{\Rightarrow} \left\{\begin{array}{l} \cancel{(000)}_{2,3}, (001)_3, (011)_3, (101)_{1,2}, \cancel{(110)}_1, (111)_1 \\ \cancel{(000)}_{2,3}, (000)_3, (010)_3, (100)_{1,2}, \cancel{(110)}_1, (110)_1 \end{array}\right]^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{array}{l} (000)_3, \cancel{(001)}_3, (010)_3, \cancel{(011)}_3, (100)_{1,2}, \cancel{(101)_{1,2}, (110)_1, (111)_1} \\ \cancel{(000)}_{2,3}, \cancel{(001)}_3, \cancel{(011)}_3, \cancel{(101)_{1,2}, (110)_1, (111)_1} \end{array}\right\}^{\oplus} \Rightarrow \left\{\begin{array}{l} \{(000)_3, (010)_3, (100)_{1,2}\}^1 \\ \{\varnothing\}^0 \end{array}\right. .$$

Table 13 contains the truth table of the given system and its "failured" variants on the 0-level of the scheme, where the "failured" values $f_{1,2,3x_1/\sim}$, $f_{1,2,3x_2/\sim}$ and $f_{1,2,3x_3/\sim}$, due to stuck-at-faults (0/1) function, are highlighted in bold.

The vectors of test codes determined on the 0-level of the scheme are placed in Table 14 — they are sets of variables for the values of "failured" functions of the given system highlighted in bold in Table 13.

On the 1-level of the scheme (*see* Fig. 2), the vectors of the test codes of the given system will determine the conjuncterms (6) of the system of functions

$$\begin{cases} f_1(z_1, z_2, z_3, z_4) = z_3 \vee z_4 \\ f_2(z_1, z_2, z_3, z_4) = z_1 \vee z_3, \\ f_3(z_1, z_2, z_3, z_4) = z_1 \vee z_2 \end{cases}$$

which in set-theoretic form has the form

$$\begin{cases} Y_1^1 = \{(--1-),(---1)\}^1 \\ Y_2^1 = \{(1---),(--1-)\}^1 \\ Y_3^1 = \{(1---),(-1--)\}^1 \end{cases}$$

Hence the set of system conjuncterms is

$$Y_{1,2,3}^1 = \{(2,\ 3,\ 6,7,10,\ 11,14,15)_{1,2},\\ (1,3,5,7,9,11,13,15)_1,\\ (8,9,10,11,12,13,14,15)_{2,3},\\ (4,5,6,7,12,13,14,15)_3\}^1.$$

The vectors of the test codes of the given sytem on the 1-level are obtained after applying the procedures (3—6) to the binary system minterms of the set $Y_{1,2,3}^1$:

$$Y_{1,2,3}^1 = \{(0001)_1, (0010)_{1,2}, (0011)_{1,2}, (0100)_3,\\ (0101)_{1,3}, (0110)_{1,2,3}, (0111)_{1,2,3}, (1000)_{1,2,3},\\ (1001)_{1,2,3}, (1010)_{1,2,3}, (1011)_{1,2,3}, (1100)_{2,3},\\ (1101)_{1,2,3}, (1110)_{1,2,3}, (1111)_{1,2,3}\}^1 .$$

For the illustrative purposes, we consider the definition of test codes only for the set $Y_{1,2,3(z_1/\sim)}^1$, since the test codes for $Y_{1,2,3(z_2/\sim)}^1$, $Y_{1,2,3(z_3/\sim)}^1$, $Y_{1,2,3(z_4/\sim)}^1$ are defined in a similar way:

Table 13

"10"	$x_1 x_2 x_3$	$f_{1,2,3}$	$f_{1,2,3x_1/\sim}$		$f_{1,2,3x_2/\sim}$		$f_{1,2,3x_3/\sim}$	
			$x_1/0$	$x_1/1$	$x_2/0$	$x_2/1$	$x_3/0$	$x_3/1$
0	0 0 0	$1_{2,3}$	$1_{2,3}$	$\mathbf{0}$	$1_{2,3}$	$\mathbf{0}$	$1_{2,3}$	$\mathbf{1}_{\mathbf{3}}$
1	0 0 1	1_3	1_3	$\mathbf{1}_{\mathbf{1,2}}$	1_3	1_3	$\mathbf{1}_{\mathbf{2,3}}$	1_3
2	0 1 0	0	0	$\mathbf{1}_{\mathbf{1}}$	$\mathbf{1}_{\mathbf{2,3}}$	0	0	$\mathbf{1}_{\mathbf{3}}$
3	0 1 1	1_3	1_3	$\mathbf{1}_{\mathbf{1}}$	1_3	1_3	$\mathbf{0}$	1_3
4	1 0 0	0	$\mathbf{1}_{\mathbf{2,3}}$	0	0	$\mathbf{1}_{\mathbf{1}}$	0	$\mathbf{1}_{\mathbf{1,2}}$
5	1 0 1	$1_{1,2}$	$\mathbf{1}_{\mathbf{3}}$	$1_{1,2}$	$1_{1,2}$	$\mathbf{1}_{\mathbf{1}}$	$\mathbf{0}$	$1_{1,2}$
6	1 1 0	1_1	$\mathbf{0}$	1_1	$\mathbf{0}$	1_1	1_1	1_1
7	1 1 1	1_1	$\mathbf{1}_{\mathbf{3}}$	1_1	$\mathbf{1}_{\mathbf{1,2}}$	1_1	1_1	1_1

Table 14

Stuck-at-fault in $f_{1,2,3}$	$x_1/\sim$	$x_2/\sim$	$x_3/\sim$
s-a-0	$(110_1)^0_{x_1/0}$ $\begin{pmatrix}100_{2,3}\\101_3\\111_3\end{pmatrix}^1_{x_1/0}$	$(110)^0_{x_2/0}$ $\begin{pmatrix}010_{2,3}\\111_{1,2}\end{pmatrix}^1_{x_2/0}$	$\begin{pmatrix}011\\101\end{pmatrix}^0_{x_3/0}$ $(001_{2,3})^1_{x_3/0}$
s-a-1	$(000)^0_{x_1/1}$ $\begin{pmatrix}001_{1,2}\\010_1\\011_1\end{pmatrix}^1_{x_1/1}$	$(000)^0_{x_2/1}$ $\begin{pmatrix}100_1\\101_1\end{pmatrix}^1_{x_2/1}$	$\begin{pmatrix}000_3\\010_3\\100_{1,2}\end{pmatrix}^1_{x_3/1}$

$$Y_{1,2,3(z_1/0)}^1 \overset{0\to 1}{\Rightarrow} \left\{\begin{matrix} (0001)_1, (0010)_{1,2}, (0011)_{1,2}, (0100)_3, (0101)_{1,3}, (0110)_{1,2,3}, (0111)_{1,2,3}, \\ \cancel{(1000)_{1,2,3}, (1001)_{1,2,3}, (1010)_{1,2,3}, (1011)_{1,2,3}, (1100)_{2,3}, (1101)_{1,2,3}, (1110)_{1,2,3}, (1111)_{1,2,3}} \\ (1001)_1, (1010)_{1,2}, (1011)_{1,2}, (1100)_3, (1101)_{1,3}, \cancel{(1110)_{1,2,3}, (1111)_{1,2,3}}, \\ \cancel{(1000)_{1,2,3}, (1001)_{1,2,3}, (1010)_{1,2,3}, (1011)_{1,2,3}, (1100)_{2,3}, (1101)_{1,2,3}, (1110)_{1,2,3}, (1111)_{1,2,3}} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{\begin{matrix} \cancel{(0001)_1, (0010)_{1,2}, (0011)_{1,2}, (0100)_3, (0101)_{1,3}, (0110)_{1,2,3}, (0111)_{1,2,3}}, \\ (1001)_1, (1010)_{1,2}, (1011)_{1,2}, (1100)_3, (1101)_{1,3}, \cancel{(1110)_{1,2,3}, (1111)_{1,2,3}} \\ \cancel{(0001)_1, (0010)_{1,2}, (0011)_{1,2}, (0100)_3, (0101)_{1,3}, (0110)_{1,2,3}, (0111)_{1,2,3}}, \\ (1000)_{1,2,3}, \cancel{(1001)}_{1,2,3}, \cancel{(1010)}_{1,2,3}, \cancel{(1011)}_{1,2,3}, \cancel{(1100)}_{2,3}, \cancel{(1101)}_{1,2,3}, \cancel{(1110)_{1,2,3}, (1111)_{1,2,3}} \end{matrix}\right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(1001)_1, (1010)_{1,2}, (1011)_{1,2}, (1100)_3, (1101)_{1,3}\}^1 \\ \{(1000)\}^0 \end{cases};$$

$$Y^{1}_{1,2,3(z_1/1)} \overset{1\to 0}{\Rightarrow} \left\{ \begin{matrix} \cancel{(0001)_1, (0010)_{1,2}, (0011)_{1,2}, (0100)_3, (0101)_{1,3}, (0110)_{1,2,3}, (0111)_{1,2,3}}, \\ (1000)_{1,2,3}, (1001)_{1,2,3}, (1010)_{1,2,3}, (1011)_{1,2,3}, (1100)_{2,3}, (1101)_{1,2,3}, (1110)_{1,2,3}, (1111)_{1,2,3} \\ \cancel{(0001)_1, (0010)_{1,2}, (0011)_{1,2}, (0100)_3, (0101)_{1,3}, (0110)_{1,2,3}, (0111)_{1,2,3}}, \\ (0000)_{1,2,3}, (0001)_{1,2,3}, (0010)_{1,2,3}, (0011)_{1,2,3}, (0100)_{2,3}, (0101)_{1,2,3}, (0110)_{1,2,3}, (0111)_{1,2,3} \end{matrix} \right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \left\{ \begin{matrix} \cancel{(1000)_{1,2,3}, (1001)_{1,2,3}, (1010)_{1,2,3}, (1011)_{1,2,3}, (1100)_{2,3}, (1101)_{1,2,3}, (1110)_{1,2,3}, (1111)_{1,2,3}}, \\ (0000)_{1,2,3}, (0001)_{1,2,3}, (0010)_{1,2,3}, (0011)_{1,2,3}, (0100)_{2,3}, (0101)_{1,2,3}, \cancel{(0110)}_{1,2,3}, \cancel{(0111)}_{1,2,3} \\ \cancel{(0001)}_1, \cancel{(0010)}_{1,2}, \cancel{(0011)}_{1,2}, \cancel{(0100)}_3, \cancel{(0101)}_{1,3}, \cancel{(0110)}_{1,2,3}, \cancel{(0111)}_{1,2,3}, \\ \cancel{(1000)_{1,2,3}, (1001)_{1,2,3}, (1010)_{1,2,3}, (1011)_{1,2,3}, (1100)_{2,3}, (1101)_{1,2,3}, (1110)_{1,2,3}, (1111)_{1,2,3}} \end{matrix} \right\}^{\oplus} \Rightarrow$$

$$\Rightarrow \begin{cases} \{(0000)_{1,2,3}, (0001)_{1,2,3}, (0010)_{1,2,3}, (0011)_{1,2,3}, (0100)_{2,3}, (0101)_{1,2,3}\}^{1} \\ \{\varnothing\}^{0} \end{cases}.$$

Table 15

"10"	$z_1z_2z_3z_4$	$f_{1,2,3}$	$f_{1,2,3z_1/\sim}$		$f_{1,2,3z_2/\sim}$		$f_{1,2,3z_3/\sim}$		$f_{1,2,3z_4/\sim}$	
			$z_1/0$	$z_1/1$	$z_2/0$	$z_2/1$	$z_3/0$	$z_3/1$	$z_4/0$	$z_4/1$
0	0 0 0 0	0	1_{2}	$\mathbf{1_{1,2,3}}$	0	$\mathbf{1_{3}}$	0	$\mathbf{1_{1,2}}$	0	$\mathbf{1_{1}}$
1	0 0 0 1	1_{1}	0	$\mathbf{1_{1,2,3}}$	1_{1}	$\mathbf{1_{1,3}}$	1_{1}	$\mathbf{1_{1,2}}$	$\mathbf{0}$	1_{1}
2	0 0 1 0	$1_{1,2}$	1_{2}	$\mathbf{1_{1,2,3}}$	$1_{1,2}$	$\mathbf{1_{1,2,3}}$	$\mathbf{0}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$
3	0 0 1 1	$1_{1,2}$	0	$\mathbf{1_{1,2,3}}$	$1_{1,2}$	$\mathbf{1_{1,2,3}}$	$\mathbf{1_{1}}$	$1_{1,2}$	$1_{1,2}$	$1_{1,2}$
4	0 1 0 0	1_{3}	1_{2}	$\mathbf{1_{2,3}}$	$\mathbf{0}$	1_{3}	1_{3}	$\mathbf{1_{1,2,3}}$	1_{3}	$\mathbf{1_{1,3}}$
5	0 1 0 1	$1_{1,3}$	0	$\mathbf{1_{1,2,3}}$	$\mathbf{1_{1}}$	$1_{1,3}$	$1_{1,3}$	$\mathbf{1_{1,2,3}}$	$\mathbf{1_{3}}$	$1_{1,3}$
6	0 1 1 0	$1_{1,2,3}$	$1_{1,2}$	$1_{1,2,3}$	$\mathbf{1_{1,2}}$	$1_{1,2,3}$	$\mathbf{1_{3}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
7	0 1 1 1	$1_{1,2,3}$	$1_{1,2}$	$1_{1,2,3}$	$\mathbf{1_{1,2}}$	$1_{1,2,3}$	$\mathbf{1_{1,3}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
8	1 0 0 0	$1_{1,2,3}$	$\mathbf{0}$	$1_{1,2,3}$	$1_{1,2,3}$	$\mathbf{1_{2,3}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
9	1 0 0 1	$1_{1,2,3}$	$\mathbf{1_{1}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
10	1 0 1 0	$1_{1,2,3}$	$\mathbf{1_{1,2}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
11	1 0 1 1	$1_{1,2,3}$	$\mathbf{1_{1,2}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
12	1 1 0 0	$1_{2,3}$	$\mathbf{1_{3}}$	$1_{2,3}$	$\mathbf{1_{1,2,3}}$	$1_{2,3}$	$1_{2,3}$	$\mathbf{1_{1,2,3}}$	$1_{2,3}$	$\mathbf{1_{1,2,3}}$
13	1 1 0 1	$1_{1,2,3}$	$\mathbf{1_{1,3}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$\mathbf{1_{2,3}}$	$1_{1,2,3}$
14	1 1 1 0	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$\mathbf{1_{2,3}}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$
15	1 1 1 1	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$	$1_{1,2,3}$

Table 16

Stuck-at-fault in $f_{1,2,3}$	$z_1/\sim$	$z_2/\sim$	$z_3/\sim$	$z_4/\sim$
s-a-0	$(1000)^0_{z_1/0}$ $\begin{pmatrix}1001_1\\1010_{1,2}\\1011_{1,2}\\1100_3\\1101_{1,3}\end{pmatrix}^1_{z_1/0}$	$(0100)^0_{z_2/0}$ $\begin{pmatrix}0101_1\\0110_{1,2}\\0111_{1,2}\\1100_{1,2,3}\end{pmatrix}^1_{z_2/0}$	$(0010)^0_{z_3/0}$ $\begin{pmatrix}0011_1\\0110_3\\0111_{1,3}\\1110_{2,3}\end{pmatrix}^1_{z_3/0}$	$(0001)^0_{z_4/0}$ $\begin{pmatrix}0101_3\\1101_{2,3}\end{pmatrix}^1_{z_4/0}$
s-a-1	$\begin{pmatrix}0000_{1,2,3}\\0001_{1,2,3}\\0010_{1,2,3}\\0011_{1,2,3}\\0100_{2,3}\\0101_{1,2,3}\end{pmatrix}^1_{z_1/1}$	$\begin{pmatrix}0000_3\\0001_{1,3}\\0010_{1,2,3}\\0011_{1,2,3}\\1000_{2,3}\end{pmatrix}^1_{z_2/1}$	$\begin{pmatrix}0000_{1,2}\\0001_{1,2}\\0100_{1,2,3}\\0101_{1,2,3}\\1100_{1,2,3}\end{pmatrix}^1_{z_3/1}$	$\begin{pmatrix}0000_1\\0100_{1,3}\\1100_{1,2,3}\end{pmatrix}^1_{z_4/1}$

Table 15 contains the truth table of the given system, as well as the truth tables of its "failured" variants, where their different values from the values of the given system are highlighted in bold.

Table 16 contains the vectors of test codes on the 1-level of the scheme determined from Table 15.

Answer. The vectors of test codes for detecting stuck-at-faults (0/1) at any point of a circuit with many outputs can be determined by both numeric set-theoretic method and tabularly.

Conclusion

A new method of stuck-at-faults (0/1) at any point of a digital PIPO-type combinational circuit described by system of logic functions is proposed. Algorithm for the compatible determination of vectors of test codes, based on a set of system conjuncterms, differs in a smaller number of implementation steps in comparison with an independent algorithm. The proposed stuck-at-faults (0/1) detection algorithm is simpler compared with other known algorithms in terms of the implementation complexity. It is due to the fact, that the procedures of finding the test codes, depending on the accepted assumption regarding the conditional malfunction of the studied circuit, are performed by simple replacement of log 0 on log 1 (3) or log 1 on log 0 (4) in the binary system minterms of the functions of the given system. Therefore, the estimation of the complexity of the proposed algorithm does not belong to the NP class of power or exponential type.

REFERENCES

1. Rytsar, B.Ye. (2023). "A Simple Stuck-at-faults Detection Method in Digital Combinational Circuits". *Control Systems and Computers*, no 1 (301), pp. 5—17. https://doi.org/10.15407/csc.2023.01.005
2. Rytsar, B.Ye. (2004). *Teoretyko-mnozhynni optymizatsiyni metody lohikovoho syntezu kombinatsiynykh merezh: dys. doktora tekhn.nauk*. Lviv, 348 p. [Рицар Б.Є. Теоретико-множинні оптимізаційні методи логікового синтезу комбінаційних мереж: дис. доктора техн.наук / Рицар Б. Є. Львів, 2004. 348 с.]

3. Parag, K. Lala. (2009). *An Introduction to Logic Circuit Testing*. Morgan & Claypool. 111 p.
4. *Kohavi, Z., Jha, N.* (2010). "Switching and Finite Automata Theory". Cambridge University Press, pp. 206—250.
5. Fujiwara, H. (1986). Logic testing and design for testability. in Comp. Syst. Series. Cambridge, MA: Mass. Inst. Tech.
6. Sasao, T. (1997). "Easily testable realizations for generalized Reed-Muller expressions". IEEE Trans. On Computers, vol. 46, no. 6, pp. 709—716.
7. Sasao, T. (1999). "Switching Theory for Logic Synthesys". Kluwer Academic Publishers, pp. 303—310.
8. Priyadarshini, K.M., Agarwal, V., Ravindran, R.E., Romitha, K.M., Kanchan, P., Harshita, K. (2020). "Logical Fault Modelling Algorithm for Stuck-at-fault". International Journal of Recent Techn. Eng., 8 (5), pp. 4302—4306.
9. Faraj, K. (2011). "Design Error Detection and Correction System based on Reed Muller Matrix for Memory Protection". International Journal of Computer Applications, 34 (8), pp. 42—48.
10. Reed, I.S. (1973). "Boolean Difference Calculus and Fault Finding". SIAM J. Appl. Math., 24 (1), pp. 134—141.
11. Samangadkar, S.S., Dudam, S.S., Sinha, A.K. (2015). "Fault Diagnosis in Combinational Logic Circuits: A Survey". International Journal for Scientific Research and Development, 3 (2), pp. 2051—2054.
12. Zakrevskiy, A.D., Pottosin, Yu.V., Cheremisinova, L.D. (2007). Logicheskiye osnovy proyektirovaniya diskretnykh ustroystv. M.: Fizmatlit., 592 p. [Закревский А.Д., Поттосин Ю.В., Черемисинова Л.Д. Логические основы проектирования дискретных устройств. Физматлит. М.: 2007. 592 с.]
13. Karkouri, Y, Aboulhamid, M. Multiple Stuck-at Fault in Logic Circuits. URL: http://www.iro.umontreal.ca/~aboulham/pdfs_sources/KCCVLSI.pdf.
14. Kajihara, S., Pomeranz, I., Kinoshita, K., Reddy, M. (1993). "Cost-Effective Generation of Minimal Test Sets for Stuck-at Faults in Combinational Logic Circuits". 30th ACWIEEE Design Automation Conference, pp. 102—106.
15. Liu, Y., Xu, Q. (2012). On modeling faults in FinFET logic circuits. In 2012 IEEE International Test Conference. Paper 11.3, pp. 1—9. shILAotDjLlJRNNc.pdf.
16. Nurcahyo, S. (2007). "General Algorithm for Testing the Combinational Logic Gates inside Digital Integrated Circuits". Int. Journal Eng. Research and App., 7 (7). (Part — 5), pp. 1—5.
17. Jameil, A.K. (2015). "A new single stuck fault detection algorithm for digital circuits". International Journal of Eng. Res. Gen. Sci, 3(1), pp. 1050—1056.
18. Malihi, L., Malihi, R. (2020). "Single stuck-at-faults detection using test generation vector and deep stacked-sparse-autoencoder". SN Applied Sciences, 2(10), 1715. https://link.springer.com/article/10.1007/s42452-020-03460-0.
19. Zhang, Z., Zhao, J. (2017). "A deep belief network based fault diagnosis model for complex chemical processes". Computers & chemical engineering, 107, pp. 395—407.
20. Astola, J.T., Stankovic, R.S. (2006). Fundamentals of Switching Theory and Logic Design. A Hands on Approach. Springer, 342 p.

Received 29.11.2023

Б.Є. Рицар, доктор технічних наук, професор,
Національний університет України «Львівська політехніка»,
79013, м. Львів, вул. С. Бандери, 12, Україна,
ORCID: https://orcid.org/0000-0002-2929-2954,
bohdanrytsar@gmail.com

ПРОСТИЙ МЕТОД ВИЯВЛЕННЯ НЕСПРАВНОСТЕЙ ТИПУ
STUCK-AT-FAULTS У ЦИФРОВИХ КОМБІНАЦІЙНИХ. СХЕМАХ. II

Вступ. Ця стаття є продовженням раніше опублікованої статті автора в науковому журналі "*Control Systems and Computers*". В основі статті лежить удосконалений автором числовий теоретико-множинний метод виявлення (діагностування) несправностей типу *stuck-at-faults* (0/1), який застосовано до цифрових комбінаційних схем типу *PIPO*, що описуються системою логікових функцій. Визначити вектори тестових кодів для виявлення місця та типу пошкодження в таких схемах доволі складно, оскільки у досліджуваній схемі пошкодження типу *stuck-at-faults* (0/1) може бути спільним для кількох функцій системи. У статті показано, що з двох можливих шляхів визначення цих векторів — незалежного і сумісного — ефективнішим щодо кроків реалізації алгоритму виявлення є останній. На відміну від відомих методів й алгоритмів пропонований підхід відрізняється простішою реалізацією пошуку векторів тестових кодів для виявлення несправностей типу *stuck-at-faults* (0/1) у довільних точках логікової схеми з багатьма виходами завдяки застосуванню кількох простих числових теоретико-множинних операцій і процедур.

Метою статті є запропонувати простий щодо реалізації числовий теоретико-множинний метод виявлення (діагностування) несправностей типу *stuck-at-faults* (0/1) у цифрових комбінаційних схемах типу *PIPO*, що описуються системою логікових функцій.

Методи. Запропоновано удосконалений числовий теоретико-множинний метод виявлення (діагностування) несправностей типу *stuck-at-faults* (0/1) у цифрових комбінаційних схемах типу *PIPO*.

Результати. Переваги пропонованого методу проілюстровано на прикладах комбінаційних схем типу *PIPO*, що описуються системою логікових функцій. Визначення векторів тестових кодів у довільній точці досліджуваної схеми виконується простими числовими теоретико-множинними операціями та процедурами. Наведені приклади засвідчують переваги пропонованого методу.

Ключові слова: *цифрові комбінаційні схеми типу PIPO, несправності типу stuck-at-faults, вектори тестових кодів, числовий теоретико-множинний метод, операції та процедури.*