UDC 519.718
DOI: 10.15587/2312-8372.2018.140351

Riznyk V.,
Solomko M.

RESEARCH OF 5-BIT BOOLEAN FUNCTIONS MINIMIZATION PROTOCOLS BY COMBINATORIAL METHOD

Об'єктом дослідження є комбінаторний метод мінімізації 5-розрядних булевих функцій. Одним з найбільш проблемних місць мінімізації булевих функцій є складність алгоритму мінімізації та гарантія отримання мінімальної функції.

У ході дослідження використовувались протоколи мінімізації 5-розрядних булевих функцій, які застосовуються за наявності у структурі таблиці істинності заданої функції повної бінарної комбінаторної системи з повторенням або неповної бінарної комбінаторної системи з повторенням. Операційні властивості протоколів мінімізації 5-розрядних булевих функцій ґрунтуються на законах та аксіомах алгебри логіки.

Отримано зменшення складності процесу мінімізації 5-розрядних булевих функцій комбінаторним методом, збільшення ймовірністі гарантованої мінімізації 5-розрядних булевих функцій. Це пов'язано з тим, що запропонований метод мінімізації 5-розрядних булевих функцій має ряд особливостей вирішення задачі мінімізації логічної функції, зокрема:

– математичний апарат блок-схеми з повторенням дає можливість отримати більше інформації стосовно ортогональності, суміжності, однозначності блоків таблиці істинності;

– рівносильні перетворення графічними образами у вигляді двовимірних матриць за рахунок більшої інформаційної ємності спроможні з ефектом замінити вербальні процедури алгебричних перетворень;

– протоколи мінімізації 5-розрядних булевих функцій складають бібліотеку протоколів для процесу мінімізації 5-розрядних булевих функцій як стандартні процедури, тому застосування окремого такого протоколу для змінних 5-розрядних булевих функцій зводиться до проведення одного алгебричного перетворення.

Завдяки цьому забезпечується можливість отримати оптимальне зменшення кількості змінних функцій без втрати її функціональності. Ефективність застосування протоколів мінімізації 5-розрядних булевих функцій комбінаторного методу демонструється прикладами мінімізації функції, запозичених з робіт інших авторів з метою порівняння.

У порівнянні з аналогічними відомими методами мінімізації булевих функцій це забезпечує:

– меншу складність процесу мінімізації 5-розрядних булевих функцій;

– збільшення ймовірності гарантованої мінімізації 5-розрядних булевих функцій;

– удосконалення алгебричного методу мінімізації булевої функції за рахунок табличної організації комбінаторного методу, впровадження апарату образного перетворення та протоколів мінімізації.

Ключові слова: *метод мінімізації, мінімізація логічної функції, блок-схема з повторенням, протоколи мінімізації булевих функцій.*

1. Introduction

The problem of minimizing the disjunctive normal form (DNF) of a logical function is one of the many-extremal logical-combinatorial problems and reduces to an optimal reduction in the number of logical elements of the gate circuit without loss of its functionality. It should be noted that in the general formulation this problem has not yet been solved, but it has been well studied in the class of disjunctive conjunctive normal forms (DCNF).

In [1, 2], a combinatorial method for minimizing Boolean functions is considered, the peculiarities of which are the greater informatively of the process of solving the problem in comparison with the algebraic method of minimizing the function, due to tabular organization and the introduction of the image-transformation apparatus.

In this paper, applications of minimization protocols for 5-bit Boolean functions of the combinatorial method are presented. The object of solving the problem of Boolean function minimization by a combinatorial method is a block-diagram with repetition, the properties of which, in turn, allow the rules of the algebra of logic to be supplemented with new rules for simplifying the function, in particular in the form of a minimization protocol.

The evolution of the methods of simplifying logical functions is the result of relentless optimization, therefore the research remains topical, in particular, on the improvement of such factors as the methodology of minimizing the logical function, ensuring the guarantee of obtaining the minimum function, the cost of technology to minimize the logical function.

2. The object of research and its technological audit

The object of research is the minimization protocols for 5-bit functions that are used when there is a complete or incomplete binary combinatorial systems with repetition in the truth table structure.

Minimization protocols for the 5-bit Boolean functions of the combinatorial method make up a protocol library for the process of minimizing 5-bit Boolean functions as

ISSN 2226-3780

standard procedures, so the application of a separate protocol is reduced to carrying out one algebraic transformation.

The effectiveness of the application of the minimization protocols for the 5-bit Boolean functions of the combinatorial method consists in greatly simplifying the procedure for minimizing the logical functions, making it possible to dispense with hardware and software automation tools to minimize the 5-bit Boolean functions.

The disadvantages of using the minimization protocols for 5-bit Boolean functions by combinatorial method associated with the small volume of existing theoretical developments for their detection, so the prospect of using the protocols for minimizing 5-bit logical functions is based on practical chances of optimal minimization of logical functions. With increasing computation time, when minimizing a function by combinatorial method, it is necessary to search for new protocols for minimizing 5-bit Boolean functions and expanding the protocol library.

3. The aim and objectives of research

The aim of research is simplification of the process of 5-bit Boolean functions minimization with the help of new combinatorial method minimization protocols.

To achieve this aim, it is necessary to solve the following tasks:

1. To establish the adequacy of the application of minimization protocols for the 5-bit Boolean functions of the combinatorial method for the process of minimizing Boolean functions.
2. To determine the operational properties of the protocols for 5-bit Boolean functions minimization when using structures of complete and incomplete binary combinatorial systems with repetition.
3. To verify the combinatorial method when applying the minimization protocols for 5-bit Boolean functions.
4. To conduct a comparative analysis of the performance and complexity of algorithms for minimizing Boolean functions obtained using the 5-bit Boolean functions of the combinatorial method with other minimization methods.

4. Research of existing solutions of the problem

The conditions for logical minimization of the Boolean function represented in DNF are considered in [3]. If the function satisfies the following conditions, then to simplify it, the classical Quine-McCluskey minimization algorithm is applied, which allows automation. It is noted that the number of function variables for the program code is limited by the computer memory. The author of the publication [3] describes the optimization method when the process involves not only searching for an equivalent logical expression, but also involves defining specific conditions under which logical expressions can be further reduced. These types of elements in the logical design are considered as the «degree of freedom». In such cases, the user can optimize the given design based on the degree of freedom. Therefore, the search for alternative solutions is desirable, since it can provide the optimal Boolean expression in the end.

Generalized rules for the simplification of conjunctors a polynomial set-theoretic format based on the proposed theorem for various initial conditions for the transformation of a pair of conjunctors, the gamming distance between which can be arbitrary, are discussed in [4]. These rules can be useful for minimizing in the polynomial set-theoretical format arbitrary logical functions of n variables. The effectiveness of the proposed rules is demonstrated by examples of minimization of the function borrowed from the work of well-known authors with a view to comparing the results. Given the comparative examples, the proposed rules give grounds for confirming the expediency of applying them in procedures for minimizing any logical function of n variables in polynomial form.

Boolean function minimization using a truth table, in which one variable is sequentially reduced until all variables, is exhausted in [5]. In the standard method, the truth table (TT) reflects the given logical function. Then the function is expressed as the sum of the minimum conditions corresponding to the sets of variables on which the function gets the value of one. Finally, this function is reduced with the help of Boolean identities. Thus, all simplifications are concentrated in one place after TT. This procedure does not always lead to minimal implementation. In [5], a simplification is considered, which at the end of each stage carries out a TT reduction. It is shown that the method is systemic and certainly leads to a minimal function. It is easy to use than on the basis of only Boolean toponyms, Carnot and Quine-McClusky maps and can handle any number of variables. This is explained by several examples.

The algorithm and program for minimizing combinational logic functions up to 20 variables are presented in [6], where the number of variables is limited by the memory of the computer system. The algorithm is based on the sequential clustering of terms, beginning with the grouping of terms with one change. The clustering algorithm ends when the variables can no longer be grouped. This algorithm is similar to the Quine-McCluskey algorithm, but it is simpler, since it eliminates a number of actions of the Quine-McCluskey algorithm.

The method of logic-minimizing image compression, which depends on the logical function, is demonstrated in [7]. The minimization process treats adjacent pixels of the image as separate minterms, representing a logical function and compresses 24-bit color images using the function minimization procedure. The compression ratio of such method is on average 25 % larger than the existing methods of image compression.

The use of the genetic algorithm for selecting side objects of the procedure for minimizing the logical function using the Carnot map is demonstrated in [8].

A new heuristic algorithm for maximum minimization of Boolean functions is presented in [9]. To implement the proposed algorithm, graphical data is used and conditions are presented to achieve the maximum degree of Boolean function minimization.

Discussion about the role of the autosymmetry degree of variables of a Boolean function and why it deserves attention on minimizing a logical function is presented in [10]. The regularity of the variables of a Boolean function can be expressed by the degree of autosymmetry, which in the end gives a new tool for effective minimization.

Optimal simplification of Boolean functions using Carnot maps using the object-oriented minimization algorithm and analyzing the performance of the proposed algorithm is considered in [11].

A method for increasing the efficiency of minimizing a logical function by applying M-terms is demonstrated in [12]. It is noted that the implementation of the method is possible for any number of variables.

In contrast to the discussed sources of Section 4, in this paper, the object of solving the problem is the minimization protocols 5-bit Boolean functions by combinatorial method in the presence of complete or incomplete binary combinatorial systems with repetition in the truth table structure.

The mathematical apparatus of the block-diagram with repetition makes it possible to obtain more information about the orthogonality, contiguity, uniqueness of truth table blocks. Equivalent transformations by graphic images in the form of two-dimensional matrices have a large information capacity in their properties, so they can effectively replace verbal procedures of algebraic transformations.

5. Methods of research

5.1. Algorithm for Boolean functions minimization by combinatorial method. Minimizing the logical function represented in DNF using a block-diagram with a repetition in the part of gluing variables is reduced to finding blocks with the same variables in the corresponding bits, except for one variable. Taking into account the tabular organization of the combinatorial method, this makes it possible to improve the search efficiency of the minimal function [1, 2].

The minimization of logical functions by combinatorial method provides the following algorithm:

– on the first stage reveals blocks (constituents) with variables for which an operation of super-gluing variables is possible in the absence of a super-gluing operation, a simple gluing operation is performed;

– the next stage is search for sets of pairs of blocks (implicants) with the possibility of minimizing them by replacing (gluing, absorbing) the variables in these pairs. The obtained sets of blocks are again minimized in a similar way, etc., until a dead-end DNF (DDNF);

– in the general case, in the final stages of minimization, it is possible to use the Blake-Poretsky method [13], as well as an increase in the number of variables with a value of unity.

Among the set of DDNFs there are also minimal functions (MDNF). After minimizing the logical function, the minimized function is verified using the specified truth table.

The beginning of the minimization procedure by combinatorial method reduces to the search for a local extremum of the minimal function. However, the apparatus of algebraic transformations of the method makes it possible to carry out transitions from one local extremum of the minimal function to another, which, thus, allows to find the global extremum of the minimal function.

During minimization of logic functions by combinatorial method, the following rules of logic algebra [1] are used:

– gluing of variables:

$$ab+a\overline{b}=a;$$

– general gluing of variables:

$$xy+\overline{x}z=xy+\overline{x}z+yz;$$

– substitution of a variable:

$$a+\overline{a}b=a+b;$$

– absorbing of a variable:

$$ab+a=a(b+1)=a;$$

– idempotency of variables:

$$a+a=a,\ aa=a;$$

– addition of a variable:

$$a+\overline{a}=1,\ a\overline{a}=0;$$

– repetition of a constant:

$$a+0=a,\ a\cdot 1=a$$

and other.

Algebraic transformations are expediently replaced by equivalent transformations by means of sub-matrices (graphical images). The procedure for gluing with sub-matrices can be demonstrated as follows:

$$\overline{x}_1\overline{x}_2+\overline{x}_1x_2=\overline{x}_1(\overline{x}_2+x_2)=\overline{x}_1,$$

$$\begin{matrix}0&0\\0&1\end{matrix}\rightarrow\begin{matrix}0\\0\end{matrix}\rightarrow 0,$$

$$x_1x_2+\overline{x}_1x_2+x_2(x_1+\overline{x}_1)=x_2,$$

$$\begin{matrix}1&1\\0&1\end{matrix}\rightarrow\begin{matrix}1\\1\end{matrix}\rightarrow 1.$$

With the help of sub-matrices (graphical images), it is possible to represent other algebraic transformations [1, 2].

Example 1. Minimize the logical function $F(a,b,c,d)=\Sigma(3,7,11,12,13,14,15)$ (Table 1) by an algebraic method. Note: the value in Σ is the minterm for rows when the function $F(a,b,c,d)$ returns «1» at the output.

Table 1

The truth table of a logical function $F(a,b,c,d)$

No.	a	b	c	d	a	b	c	d	F
3	0	0	1	1	$\overline{a}$	$\overline{b}$	c	d	**1**
7	0	1	1	1	$\overline{a}$	b	c	d	**1**
11	1	0	1	1	a	$\overline{b}$	c	d	**1**
12	1	1	0	0	a	b	$\overline{c}$	$\overline{d}$	**1**
13	1	1	0	1	a	b	$\overline{c}$	d	**1**
14	1	1	1	0	a	b	c	$\overline{d}$	**1**
15	1	1	1	1	a	b	c	d	**1**

$$F(a,b,c,d)=\Sigma(3,7,11,12,13,14,15)=$$
$$=\overline{a}\overline{b}cd+\overline{a}bcd+a\overline{b}cd+ab\overline{c}\overline{d}+ab\overline{c}d+abc\overline{d}+abcd=$$
$$=cd(\overline{a}\overline{b}+\overline{a}b+a\overline{b})+ab(\overline{c}\overline{d}+\overline{c}d+c\overline{d}+cd)=$$
$$=cd(\overline{a}[\overline{b}+b]+a\overline{b})+ab(\overline{c}[\overline{d}+d]+c[\overline{d}+d])=$$
$$=cd(\overline{a}[1]+a\overline{b})+ab(\overline{c}[1]+c[1])=ab+a\overline{b}cd+\overline{a}cd=$$
$$=ab+cd(a\overline{b}+\overline{a})=ab+cd(a+\overline{a})(\overline{a}+\overline{b})=ab+\overline{a}cd+\overline{b}cd=$$
$$=ab+cd(\overline{a}+\overline{b})=ab+cd.$$

ISSN 2226-3780

Example 2. Minimize the logical function $F(a,b,c,d) = \Sigma(3,7,11,12,13,14,15)$ by combinatorial method:

$$F = \begin{vmatrix} 3 & 0\ 0\ 1\ 1 \\ 7 & 0\ 1\ 1\ 1 \\ 11 & 1\ 0\ 1\ 1 \\ 12 & 1\ 1\ 0\ 0 \\ 13 & 1\ 1\ 0\ 1 \\ 14 & 1\ 1\ 1\ 0 \\ 15 & 1\ 1\ 1\ 1 \end{vmatrix} = \begin{vmatrix} \sim 0\ 1\ 1 \\ 0\ 1\ 1\ 1 \\ 1\ 1 \sim \sim \end{vmatrix} = \begin{vmatrix} \sim 0\ 1\ 1 \\ \sim 1\ 1\ 1 \\ 1\ 1 \sim \sim \end{vmatrix} = \begin{vmatrix} \sim \sim 1\ 1 \\ 1\ 1 \sim \sim \end{vmatrix}.$$

Minimized function:

$$F = ab + cd.$$

The operation of super-gluing of variables in the first matrix is carried out for blocks 12–15, highlighted in red. The result of minimization by combinatorial method coincides with the result of minimization obtained with the help of algebraic method, however, the process of minimizing a function by a combinatorial method is simple.

Example 3. Minimize the logic function $F(x_1, x_2, x_3, x_4)$ by the combinatorial method given by the following truth table $\Sigma(0, 1, 2, 3, 5, 7, 8, 10, 11, 12, 13)$ [14].

In [14], the minimization of the function reduces to the synthesis of the infimum disjunctive normal form (IDNF) of the logical function, using the perfect matrix arrangement (PMA) of the 4-dimensional cube E^4 (Fig. 1). The vertices of the cube E^4 of a given function where $F(x_1, x_2, x_3, x_4) = 1$ are highlighted by darkening. The darkened vertices correspond to blocks of the truth table $\Sigma(0, 1, 2, 3, 5, 7, 8, 10, 11, 12, 13)$ of the given logical function.

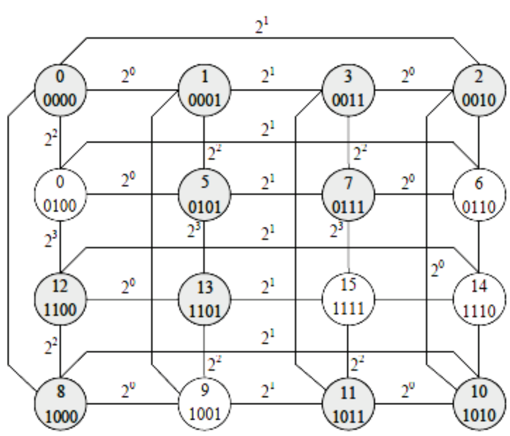

Fig. 1. Perfect matrix arrangement of a 4-dimensional cube E^4

Minimizing the function $F(x_1, x_2, x_3, x_4)$ by combinatorial method is reduced to the following procedure:

$$F = \begin{vmatrix} 0 & 0\ 0\ 0\ 0 \\ 1 & 0\ 0\ 0\ 1 \\ 2 & 0\ 0\ 1\ 0 \\ 3 & 0\ 0\ 1\ 1 \\ 5 & 0\ 1\ 0\ 1 \\ 7 & 0\ 1\ 1\ 1 \\ 8 & 1\ 0\ 0\ 0 \\ 10 & 1\ 0\ 1\ 0 \\ 11 & 1\ 0\ 1\ 1 \\ 12 & 1\ 1\ 0\ 0 \\ 13 & 1\ 1\ 0\ 1 \end{vmatrix} = \begin{vmatrix} \sim 0\ 0\ 0 \\ 0 \sim 0\ 1 \\ \sim 0\ 1 \sim \\ 0\ 1\ 1\ 1 \\ 1\ 1\ 0 \sim \end{vmatrix} = \begin{vmatrix} \sim 0 \sim 0 \\ 0 \sim 0\ 1 \\ \sim 0\ 1 \sim \\ 0 \sim 1\ 1 \\ 1\ 1\ 0 \sim \end{vmatrix} = \begin{vmatrix} \sim 0 \sim 0 \\ 0 \sim \sim 1 \\ \sim 0\ 1 \sim \\ 1\ 1\ 0 \sim \end{vmatrix}.$$

Blocks 2, 3, 10, 11 (highlighted in red) are minimized by the protocol of super-gluing variables. Other blocks are minimized by simple gluing and semi-gluing protocols [1, 2].

Minimized function:

$$F = \overline{x}_2\overline{x}_4 + x_1\overline{x}_4 + \overline{x}_2 x_3 + x_1 x_2 \overline{x}_3. \quad (1)$$

The result of minimization (1) coincides with the result of synthesizing the infimum disjunctive normal form of the logical function [14], but the combinatorial method is a simple procedure.

5.2. Protocols for 5-bit Boolean functions minimization. For a 5-bit logical function, the protocols for super-gluing of variables are:

– 1st protocol:

$$\begin{vmatrix} 0\ 0\ 0\ 0\ x \\ 0\ 0\ 0\ 1\ x \\ 0\ 0\ 1\ 0\ x \\ 0\ 0\ 1\ 1\ x \\ 0\ 1\ 0\ 0\ x \\ 0\ 1\ 0\ 1\ x \\ 0\ 1\ 1\ 0\ x \\ 0\ 1\ 1\ 1\ x \\ 1\ 0\ 0\ 0\ x \\ 1\ 0\ 0\ 1\ x \\ 1\ 0\ 1\ 0\ x \\ 1\ 0\ 1\ 1\ x \\ 1\ 1\ 0\ 0\ x \\ 1\ 1\ 0\ 1\ x \\ 1\ 1\ 1\ 0\ x \\ 1\ 1\ 1\ 1\ x \end{vmatrix} = x; \quad (2)$$

– second protocol:

$$\begin{vmatrix} 0\ 0\ 0\ x\ y \\ 0\ 0\ 1\ x\ y \\ 0\ 1\ 0\ x\ y \\ 0\ 1\ 1\ x\ y \\ 1\ 0\ 0\ x\ y \\ 1\ 0\ 1\ x\ y \\ 1\ 1\ 0\ x\ y \\ 1\ 1\ 1\ x\ y \end{vmatrix} = xy; \quad (3)$$

– third protocol:

$$\begin{vmatrix} 0\ 0\ x\ y\ z \\ 0\ 1\ x\ y\ z \\ 1\ 0\ x\ y\ z \\ 1\ 1\ x\ y\ z \end{vmatrix} = xyz; \quad (4)$$

– fourth protocol:

$$\begin{vmatrix} 0\ x\ y\ z\ t \\ 1\ x\ y\ z\ t \end{vmatrix} = xyzt. \quad (5)$$

The 1st protocol (2) uses the combinatorial system 2-(4, 16)-design, the 2^{nd} (3) – 2-(3, 8)-design, the 3^{rd} (4) – 2-(2, 4)-design, the 4^{th} (5) – 2-(1, 2)-design.

The variables x, y, z, t, which form a complete binary combinatorial system with a repetition of 2-(n, b)-design, can occupy any bit of the minterm of a 5-bit logical function.

The protocol of super-gluing variables on a configuration with two combinatorial systems 2-(2, 4)-design can have the following form:

$$\begin{vmatrix} 0&0&1&0&1 \\ 0&0&1&1&1 \\ 0&1&1&0&1 \\ 0&1&1&1&1 \\ 1&0&1&0&1 \\ 1&0&1&1&1 \\ 1&1&1&0&1 \\ 1&1&1&1&1 \end{vmatrix} = \begin{vmatrix} \sim & \sim & 1 & \sim & 1 \end{vmatrix} \qquad (6)$$

or

$$\begin{vmatrix} 0&0&0&1&0 \\ 0&0&0&1&1 \\ 0&1&0&1&0 \\ 0&1&0&1&1 \\ 1&0&0&1&0 \\ 1&0&0&1&1 \\ 1&1&0&1&0 \\ 1&1&0&1&1 \end{vmatrix} = \begin{vmatrix} \sim & \sim & 0 & 1 & \sim \end{vmatrix}. \qquad (7)$$

The matrix (6) contains two configurations, indicated in red and blue with 2-(2, 4)-design systems. The result of minimizing the red configuration is the block:

$$\begin{vmatrix} \sim & \sim & 1 & 0 & 1 \end{vmatrix}. \qquad (8)$$

The result of minimizing the blue configuration is the block:

$$\begin{vmatrix} \sim & \sim & 1 & 1 & 1 \end{vmatrix}. \qquad (9)$$

The operation of gluing the interchangeable blocks (8) and (9) gives the final result of the protocol for super-gluing variables on a configuration with two combinatorial systems 2- (2, 4)-design:

$$\begin{vmatrix} \sim & \sim & 1 & \sim & 1 \end{vmatrix}.$$

The protocol of simple gluing of variables on a configuration with two combinatorial systems 2-(1, 2)-design has, for example, the following:

$$\begin{vmatrix} 0&0&0&0&0 \\ 0&0&0&1&0 \\ 1&0&0&0&0 \\ 1&0&0&1&0 \end{vmatrix} = \begin{vmatrix} \sim & 0 & 0 & \sim & 0 \end{vmatrix}. \qquad (10)$$

The matrix (10) contains two configurations, indicated in red and blue with 2-(1, 2)-design systems. The result of minimizing the red configuration will be:

$$\begin{vmatrix} 0 & 0 & 0 & \sim & 0 \end{vmatrix}. \qquad (11)$$

The result of minimizing the blue configuration is the block:

$$\begin{vmatrix} 1 & 0 & 0 & \sim & 0 \end{vmatrix}. \qquad (12)$$

The operation of gluing the replacement blocks (11) and (12) yields the final result of the protocol of simple gluing variables on a configuration with two combinatorial systems 2-(1, 2)-design:

$$\begin{vmatrix} \sim & 0 & 0 & \sim & 0 \end{vmatrix}.$$

The protocol of super-gluing of variables on a configuration in which there is one column with the same variables y, and the second column contains equally the variables x and $\bar{x}$, with the combinatorial system 2-(3, 8)-design with lexicographic order takes the form:

$$\begin{vmatrix} 0&0&0&\bar{x}&y \\ 0&0&1&\bar{x}&y \\ 0&1&0&\bar{x}&y \\ 0&1&1&\bar{x}&y \\ 1&0&0&x&y \\ 1&0&1&x&y \\ 1&1&0&x&y \\ 1&1&1&x&y \end{vmatrix} = \begin{vmatrix} 0&\sim&\sim&\bar{x}&y \\ 1&\sim&\sim&x&y \end{vmatrix}. \qquad (13)$$

A protocol for super-gluing of variables on a configuration in which there is one column with the same variables y, and the second column contains equally the variables x and $\bar{x}$, with the combinatorial system 2-(3, 8)-design and the sequential lexicographic order becomes:

$$\begin{vmatrix} y&\bar{x}&0&0&0 \\ y&\bar{x}&0&1&0 \\ y&\bar{x}&1&0&0 \\ y&\bar{x}&1&1&0 \\ y&x&0&0&1 \\ y&x&0&1&1 \\ y&x&1&0&1 \\ y&x&1&1&1 \end{vmatrix} = \begin{vmatrix} y&\bar{x}&\sim&\sim&0 \\ y&x&\sim&\sim&1 \end{vmatrix}. \qquad (14)$$

The protocol of gluing variables on a configuration in which there is one column with the same variables y, and the second column contains equally the variables x and $\bar{x}$, with the combinatorial system 2-(3, 8)-design with non-lexicographic order, will change the form of the record:

$$\begin{vmatrix} \bar{x}&y&0&1&1 \\ \bar{x}&y&1&0&0 \\ \bar{x}&y&1&0&1 \\ \bar{x}&y&1&1&1 \\ x&y&0&0&0 \\ x&y&0&0&1 \\ x&y&0&1&0 \\ x&y&1&1&0 \end{vmatrix} = \begin{vmatrix} \bar{x}&y&1&0&\sim \\ \bar{x}&y&\sim&1&1 \\ x&y&0&0&\sim \\ x&y&\sim&1&0 \end{vmatrix}. \qquad (15)$$

ISSN 2226-3780

The protocol of gluing variables on a configuration in which there is one column with the same variables y, and the second column contains equally the variables x and $\bar{x}$, with the redundant 2-(3, 8)-design with non-lexicographic order, takes on a different form:

$$\begin{vmatrix} y & \bar{x} & 0 & 0 & 1 \\ y & \bar{x} & 0 & 1 & 0 \\ y & \bar{x} & 0 & 1 & 1 \\ y & \bar{x} & 1 & 0 & 0 \\ y & \bar{x} & 1 & 0 & 1 \\ y & \bar{x} & 1 & 1 & 1 \\ y & x & 0 & 0 & 1 \\ y & x & 0 & 1 & 1 \\ y & x & 1 & 0 & 0 \\ y & x & 1 & 0 & 1 \\ y & x & 1 & 1 & 0 \\ y & x & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} y & \bar{x} & \sim & \sim & 1 \\ y & \bar{x} & 0 & 1 & \sim \\ y & \bar{x} & 1 & 0 & \sim \\ y & x & \sim & \sim & 1 \\ y & x & 1 & \sim & \sim \end{vmatrix} = \begin{vmatrix} y & \sim & \sim & \sim & 1 \\ y & \bar{x} & 0 & 1 & \sim \\ y & \sim & 1 & 0 & \sim \\ y & x & 1 & \sim & \sim \end{vmatrix}. \quad (16)$$

In general, the configuration of the truth table of a given function, in addition to the sub-matrix of a complete combinatorial system with a repetition of 2-(n, b)-design, also contains sub-matrices of an incomplete combinatorial system with a repetition of 2-(n, x/b) – design, where x is the number of blocks of incomplete combinatorial system with repetition.

The properties of an incomplete combinatorial system with repetition 2-(n, x/b) – design also allow the establishment of protocols that ensure the effective minimization of Boolean functions [2].

The protocol of gluing variables on a configuration in which there is one column with the same variables y, and the second column includes in the same number of variables x and $\bar{x}$, with an excess combinatorial system 2-(3, 6/8)-design with non-lexicographic order has, for example, this form:

$$\begin{vmatrix} y & \bar{x} & 0 & 0 & 1 \\ y & \bar{x} & 0 & 1 & 0 \\ y & \bar{x} & 0 & 1 & 1 \\ y & \bar{x} & 1 & 0 & 1 \\ y & \bar{x} & 1 & 1 & 1 \\ y & x & 0 & 0 & 1 \\ y & x & 0 & 1 & 1 \\ y & x & 1 & 0 & 1 \\ y & x & 1 & 1 & 0 \\ y & x & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} y & \bar{x} & \sim & \sim & 1 \\ y & \bar{x} & 0 & 1 & \sim \\ y & x & \sim & \sim & 1 \\ y & x & 1 & 1 & \sim \end{vmatrix} = \begin{vmatrix} y & \sim & \sim & \sim & 1 \\ y & \bar{x} & 0 & 1 & \sim \\ y & x & 1 & 1 & \sim \end{vmatrix}. \quad (17)$$

The protocol for gluing together variables in a configuration in which there is one column with the same variables y, and the second column holds in an unequal number of variables x and $\bar{x}$, with an excess combinatorial system 2-(3, 6/8)-design with non-lexicographic order, will be, for example:

$$\begin{vmatrix} y & \bar{x} & 0 & 0 & 0 \\ y & \bar{x} & 0 & 0 & 1 \\ y & \bar{x} & 0 & 1 & 0 \\ y & \bar{x} & 1 & 1 & 0 \\ y & x & 0 & 1 & 0 \\ y & x & 1 & 1 & 0 \\ y & x & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} y & \bar{x} & 0 & 0 & \sim \\ y & \bar{x} & \sim & 1 & 0 \\ y & x & \sim & 1 & 0 \\ y & x & 1 & 1 & \sim \end{vmatrix} = \begin{vmatrix} y & \bar{x} & 0 & 0 & \sim \\ y & \sim & \sim & 1 & 0 \\ y & x & 1 & 1 & \sim \end{vmatrix} \quad (18)$$

or

$$\begin{vmatrix} y & \bar{x} & 0 & 0 & 0 \\ y & \bar{x} & 0 & 0 & 1 \\ y & \bar{x} & 0 & 1 & 0 \\ y & \bar{x} & 1 & 0 & 0 \\ y & \bar{x} & 1 & 1 & 0 \\ y & x & 0 & 1 & 0 \\ y & x & 1 & 0 & 0 \\ y & x & 1 & 1 & 0 \\ y & x & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} y & \bar{x} & 0 & 0 & \sim \\ y & \bar{x} & \sim & \sim & 0 \\ y & x & \sim & 1 & 0 \\ y & x & 1 & \sim & 0 \\ y & x & 1 & 1 & \sim \end{vmatrix} = \begin{vmatrix} y & \bar{x} & 0 & 0 & \sim \\ y & \bar{x} & \sim & \sim & 0 \\ y & \sim & \sim & 1 & 0 \\ y & \sim & 1 & \sim & 0 \\ y & x & 1 & 1 & \sim \end{vmatrix}. \quad (19)$$

The protocol of gluing variables on a configuration with a combinatorial system 2-(3, 6/8)-design can have the following form:

$$\begin{vmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 0 & \sim & 1 & 1 \\ 0 & \sim & 1 & 0 & 1 \\ 0 & 1 & 0 & \sim & 1 \end{vmatrix}. \quad (20)$$

The protocol of gluing variables on a configuration with a combinatorial system 2-(2, 3/4)-design is as follows:

$$\begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{vmatrix} = \begin{vmatrix} \sim & 0 & 0 & 0 & 0 \\ 1 & \sim & 0 & 0 & 0 \end{vmatrix}. \quad (21)$$

Protocols (2)–(7), (10), (13)–(21) constitute a protocol library for the process of 5-bit Boolean functions minimization as standard procedures. Therefore, the use of a separate protocol for variables of 5-bit Boolean functions reduces to carrying out one algebraic transformation.

6. Research results

Example 4. Minimize the logic function $F(x_1, x_2, x_3, x_4, x_5)$ by a combinatorial method without the minimization protocol, which is specified by the following truth table Σ(1, 2, 3, 4, 5, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18, 20, 22, 26, 28, 30, 31) [1].

In one of the solutions, pairs of blocks that allow procedures for pasting and replacing variables are first identified.

At the first stage, it is possible to carry out pasting of constituents and substitution of variables.

$F =$

No.	1					2					3					4				
	x_1	x_2	x_3	x_4	x_5	x_1	x_2	x_3	x_4	x_5	x_1	x_2	x_3	x_4	x_5	x_1	x_2	x_3	x_4	x_5
1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0		1
2	0	0	0	1	0															
3	0	0	0	1	1	0	0	0	1		0	0	0	1		0	0	0	1	
4	0	0	1	0	0	0	0	1	0		0	0	1	0		0	0	1	0	
5	0	0	1	0	1	0	0	1		1	0	0	1		1	0	0	1		1
6	0	0	1	1	1															
7	0	1	0	0	1															
8	0	1	0	1	1	0	1	0		1	0	1	0		1	0	1	0		1
9	0	1	1	0	0															
10	0	1	1	0	1	0	1	1	0		0	1	1	0		0	1	1		
11	0	1	1	1	0															
12	0	1	1	1	1	0	1	1	1											
13	1	0	0	0	0															
14	1	0	0	0	1	1	0	0	0		1	0	0	0		1	0	0	0	
15	1	0	0	1	0	1	0		1	0	1	0		1	0	1	0		1	0
16	1	0	1	0	0	1	0	1		0	1	0	1		0	1	0	1		0
17	1	0	1	1	0															
18	1	1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1		1	0
19	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0	1	1	1		0
20	1	1	1	1	0															
21	1	1	1	1	1	1	1	1	1			1	1	1			1	1	1	

$=$

Algebraic transformations of the 1st matrix (the result of the transformation is written in the second matrix):

– gluing of variables of 2 and 3 blocks:

$$\bar{x}_1\bar{x}_2\bar{x}_3x_4\bar{x}_5+\bar{x}_1\bar{x}_2\bar{x}_3x_4x_5=$$
$$=\bar{x}_1\bar{x}_2\bar{x}_3x_4(\bar{x}_5+x_5)=\bar{x}_1\bar{x}_2\bar{x}_3x_4;$$

– gluing and replacing of variables of 4, 5 and 6 blocks:

$$\bar{x}_1\bar{x}_2x_3\bar{x}_4\bar{x}_5+\bar{x}_1\bar{x}_2x_3\bar{x}_4x_5+\bar{x}_1\bar{x}_2x_3x_4x_5=$$
$$=\bar{x}_1\bar{x}_2x_3(\bar{x}_4\bar{x}_5+\bar{x}_4x_5+x_4x_5)=$$
$$=\bar{x}_1\bar{x}_2x_3(\bar{x}_4\bar{x}_5+\bar{x}_4x_5+\bar{x}_4+x_4x_5)=$$
$$=\bar{x}_1\bar{x}_2x_3(\bar{x}_4\bar{x}_5+\bar{x}_4x_5+\bar{x}_4+x_5)=$$
$$=\bar{x}_1\bar{x}_2x_3(\bar{x}_4(\bar{x}_5+x_5)+\bar{x}_4+x_5)=$$
$$=\bar{x}_1\bar{x}_2x_3(\bar{x}_4+\bar{x}_4+x_5)=$$
$$=\bar{x}_1\bar{x}_2x_3(\bar{x}_4+x_5)=\bar{x}_1\bar{x}_2x_3\bar{x}_4+\bar{x}_1\bar{x}_2x_3x_5;$$

– gluing of variables of 7 and 8 blocks:

$$\bar{x}_1x_2\bar{x}_3\bar{x}_4x_5+\bar{x}_1x_2\bar{x}_3x_4x_5=$$
$$=\bar{x}_1x_2\bar{x}_3x_5(\bar{x}_4+x_4)=\bar{x}_1x_2\bar{x}_3x_5;$$

– gluing of variables of 9 and 10 blocks:

$$\bar{x}_1x_2x_3\bar{x}_4\bar{x}_5+\bar{x}_1x_2x_3\bar{x}_4x_5=$$
$$=\bar{x}_1x_2x_3\bar{x}_4(\bar{x}_5+x_5)=\bar{x}_1x_2x_3\bar{x}_4;$$

– gluing of variables of 11 and 12 blocks:

$$\bar{x}_1x_2x_3x_4\bar{x}_5+\bar{x}_1x_2x_3x_4x_5=$$
$$=\bar{x}_1x_2x_3x_4(\bar{x}_5+x_5)=\bar{x}_1x_2x_3x_4;$$

– gluing of variables of 13 and 14 blocks:

$$x_1\bar{x}_2\bar{x}_3\bar{x}_4\bar{x}_5+x_1\bar{x}_2\bar{x}_3\bar{x}_4x_5=$$
$$=x_1\bar{x}_2\bar{x}_3\bar{x}_4(\bar{x}_5+x_5)=x_1\bar{x}_2\bar{x}_3\bar{x}_4;$$

– gluing and replacing of variables of 15, 16 and 17 blocks:

$$x_1\bar{x}_2\bar{x}_3x_4\bar{x}_5+x_1\bar{x}_2x_3\bar{x}_4\bar{x}_5+x_1\bar{x}_2x_3x_4\bar{x}_5=x_1\bar{x}_2\bar{x}_5\times$$
$$\times(\bar{x}_3x_4+x_3\bar{x}_4+x_3x_4)=x_1\bar{x}_2\bar{x}_5(\bar{x}_3x_4+x_3x_4+x_3\bar{x}_4)=$$
$$=x_1\bar{x}_2\bar{x}_5(\bar{x}_3x_4+x_3x_4+x_4+x_3\bar{x}_4)=$$
$$=x_1\bar{x}_2\bar{x}_5(x_4(\bar{x}_3+x_3)+x_4+x_3)=x_1\bar{x}_2\bar{x}_5(x_4+x_4+x_3)=$$
$$=x_1\bar{x}_2\bar{x}_5(x_4+x_3)=x_1\bar{x}_2x_3\bar{x}_5+x_1\bar{x}_2x_4\bar{x}_5;$$

– gluing of variables of 20 and 21 blocks:

$$x_1\bar{x}_2\bar{x}_3\bar{x}_4\bar{x}_5+x_1\bar{x}_2\bar{x}_3\bar{x}_4x_5=$$
$$=x_1\bar{x}_2\bar{x}_3\bar{x}_4(\bar{x}_5+x_5)=x_1\bar{x}_2\bar{x}_3\bar{x}_4.$$

At the second stage, the implicants are glued together and the substitutions of the variables are made.

Algebraic transformations of the 2nd matrix (the result of the transformation is written in the 3rd matrix):

– gluing of variables of 12 and 21 blocks:

$$\bar{x}_1x_2x_3x_4+x_1x_2x_3x_4=x_2x_3x_4(\bar{x}_1+x_1)=x_2x_3x_4.$$

Algebraic transformations of the 3rd matrix (the result of the transformation is written in the 4th matrix):

– substitution of variables for 10, 18, 19 and 21 blocks:

$$\bar{x}_1x_2x_3\bar{x}_4+x_1x_2\bar{x}_3x_4\bar{x}_5+x_1x_2x_3\bar{x}_4\bar{x}_5+x_2x_3x_4=$$
$$=x_2x_3x_4+\bar{x}_1x_2x_3\bar{x}_4+x_1x_2x_3\bar{x}_4\bar{x}_5+x_2x_3x_4+$$
$$+x_1x_2\bar{x}_3x_4\bar{x}_5=x_2x_3(x_4+\bar{x}_1\bar{x}_4+x_1\bar{x}_4\bar{x}_5)+$$
$$+x_2x_4(x_3+x_1\bar{x}_3\bar{x}_5)=x_2x_3(x_4+\bar{x}_1+x_1\bar{x}_5)+$$
$$+x_2x_4(x_3+x_1\bar{x}_5)=x_2x_3(x_4+\bar{x}_1+x_1\bar{x}_5)+$$
$$+x_2x_4(x_3+x_1\bar{x}_5)=x_2x_3x_4+\bar{x}_1x_2x_3+$$
$$+x_1x_2x_3\bar{x}_5+x_2x_3x_4+x_1x_2x_4\bar{x}_5=x_2x_3x_4+$$
$$+\bar{x}_1x_2x_3+x_1x_2x_3\bar{x}_5+x_2x_3x_4+x_1x_2x_4\bar{x}_5=$$
$$=x_2x_3x_4+\bar{x}_1x_2x_3+x_1x_2x_3\bar{x}_5+x_1x_2x_4\bar{x}_5;$$

– substitution of variables for 1 and 3 blocks:

$$\bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4x_5+\bar{x}_1\bar{x}_2\bar{x}_3x_4=\bar{x}_1\bar{x}_2\bar{x}_3(\bar{x}_4x_5+x_4)=$$
$$=\bar{x}_1\bar{x}_2\bar{x}_3(x_5+x_4)=\bar{x}_1\bar{x}_2\bar{x}_3x_5+\bar{x}_1\bar{x}_2\bar{x}_3x_4.$$

Algebraic transformations of the 4th matrix (the result of the transformation is written in the 5th matrix):

– gluing of variables of 15 and 18 blocks:

$$x_1\bar{x}_2x_4\bar{x}_5+x_1x_2x_4\bar{x}_5=x_1x_4\bar{x}_5(\bar{x}_2+x_2)=x_1x_4\bar{x}_5;$$

– gluing of variables of 16 and 19 blocks:

$$x_1\bar{x}_2x_3\bar{x}_5+x_1x_2x_3\bar{x}_5=x_1x_3\bar{x}_5(\bar{x}_2+x_2)=x_1x_3\bar{x}_5;$$

– substitution of variables of 8 and 10 blocks:

$$\bar{x}_1x_2\bar{x}_3x_5+\bar{x}_1x_2x_3=\bar{x}_1x_2(\bar{x}_3x_5+x_3)=$$
$$=\bar{x}_1x_2(x_5+x_3)=\bar{x}_1x_2x_5+\bar{x}_1x_2x_3;$$

 ISSN 2226-3780

– substitution of variables of 4 and 10 blocks:

$$\overline{x_1}\,\overline{x_2}x_3\overline{x_4}+\overline{x_1}x_2x_3=\overline{x_1}x_3(\overline{x_2}\,\overline{x_4}+x_2)=$$
$$=\overline{x_1}x_3(\overline{x_4}+x_2)=\overline{x_1}x_3\overline{x_4}+\overline{x_1}x_2x_3;$$

– gluing of variables of 1 and 5 blocks:

$$\overline{x_1}\,\overline{x_2}\,\overline{x_3}x_5+\overline{x_1}\,\overline{x_2}x_3x_5=\overline{x_1}\,\overline{x_2}x_5(\overline{x_3}+x_3)=\overline{x_1}\,\overline{x_2}x_5.$$

=

No.	5					6					7					8				
	x_1	x_2	x_3	x_4	x_5	x_1	x_2	x_3	x_4	x_5	x_1	x_2	x_3	x_4	x_5	x_1	x_2	x_3	x_4	x_5
1																				
2																				
3	0	0	0	1		0	0	0	1		0	0	0	1		0	0	0	1	
4	0		1	0		0		1	0		0		1	0						
5	0	0			1															
6																				
7																				
8	0	1			1	0				1	0				1	0				1
9																				
10	0	1	1			0	1	1												
11																				
12																				
13																				
14	1	0	0	0		1	0	0	0		1	0	0	0		1	0	0	0	
15																				
16													1	0	0			1	0	0
17																				
18	1			1	0	1			1	0	1			1	0	1			1	0
19	1		1		0	1		1		0	1		1		0					
20																				
21		1	1	1			1	1	1			1	1	1			1	1	1	

Algebraic transformations of the 5th matrix (the result of the transformation is written in the 6th matrix):

– gluing of variables of 5 and 8 blocks:

$$\overline{x_1}\,\overline{x_2}x_5+\overline{x_1}x_2x_5=\overline{x_1}x_5(\overline{x_2}+x_2)=\overline{x_1}x_5.$$

Algebraic transformations of the 6th matrix (the result of the transformation is written in the 7th matrix):

– generalized substitution of variables for 4, 8 and 16 blocks:

$$\overline{x_1}x_3\overline{x_4}+\overline{x_1}x_2x_3+x_2x_3x_4=\overline{x_1}x_3\overline{x_4}+$$
$$+x_2x_3x_4+\overline{x_1}x_2x_3=\overline{x_1}x_3\overline{x_4}+x_2x_3x_4;$$

– generalized substitution of variables for 4 and 19 blocks:

$$\overline{x_1}x_3\overline{x_4}+x_1x_3\overline{x_5}=\overline{x_1}x_3\overline{x_4}+x_1x_3\overline{x_5}+$$
$$+x_3x_3\overline{x_4}\,\overline{x_5}=\overline{x_1}x_3\overline{x_4}+x_1x_3\overline{x_5}+x_3\overline{x_4}\,\overline{x_5}.$$

Algebraic transformations of the 7th matrix (the result of the transformation is written in the 8th matrix):

– generalized substitution of variables for 4, 10 and 21 blocks:

$$\overline{x_1}x_5+x_3\overline{x_4}\,\overline{x_5}+\overline{x_1}x_3\overline{x_4}=\overline{x_1}x_5+x_3\overline{x_4}\,\overline{x_5};$$

– generalized substitution of variables for 16, 18 and 19 blocks:

$$x_3\overline{x_4}\,\overline{x_5}+x_1x_4\overline{x_5}=x_3\overline{x_4}\,\overline{x_5}+x_1x_4\overline{x_5}+x_1x_3\overline{x_5}\,\overline{x_5}=$$
$$=x_3\overline{x_4}\,\overline{x_5}+x_1x_4\overline{x_5}+x_1x_3\overline{x_5}=x_3\overline{x_4}\,\overline{x_5}+x_1x_4\overline{x_5}.$$

Attempts at further application of algebraic transformation operations do not yield a result. So the minimal form of the logical function is found:

$$F=\overline{x_1}x_5+\overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4+x_2x_3x_4+$$
$$+x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4}+x_1x_4\overline{x_5}+x_3\overline{x_4}\,\overline{x_5}.$$

Example 5. Minimize the logical function $F(x_1, x_2, x_3, x_4, x_5)$ given by the truth table Σ(1, 2, 3, 4, 5, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18, 20, 22, 26, 28, 30, 31) by the combinatorial method, using the minimization protocols:

$$F=\begin{vmatrix} 1 & 0\;0\;0\;0\;1 \\ 2 & 0\;0\;0\;1\;0 \\ 3 & 0\;0\;0\;1\;1 \\ 4 & 0\;0\;1\;0\;0 \\ 5 & 0\;0\;1\;0\;1 \\ 7 & 0\;0\;1\;1\;1 \\ 9 & 0\;1\;0\;0\;1 \\ 11 & 0\;1\;0\;1\;1 \\ 12 & 0\;1\;1\;0\;0 \\ 13 & 0\;1\;1\;0\;1 \\ 14 & 0\;1\;1\;1\;0 \\ 15 & 0\;1\;1\;1\;1 \\ 16 & 1\;0\;0\;0\;0 \\ 17 & 1\;0\;0\;0\;1 \\ 18 & 1\;0\;0\;1\;0 \\ 20 & 1\;0\;1\;0\;0 \\ 22 & 1\;0\;1\;1\;0 \\ 26 & 1\;1\;0\;1\;0 \\ 28 & 1\;1\;1\;0\;0 \\ 30 & 1\;1\;1\;1\;0 \\ 31 & 1\;1\;1\;1\;1 \end{vmatrix}=\begin{vmatrix} 0\;0\;0\;0\;1 \\ 0\;0\;0\;1\;0 \\ 0\;0\;0\;1\;1 \\ 0\;0\;1\;0\;1 \\ 0\;0\;1\;1\;1 \\ 0\;1\;0\;0\;1 \\ 0\;1\;0\;1\;1 \\ 0\;1\;1\;0\;1 \\ 0\;1\;1\;1\;0 \\ 0\;1\;1\;1\;1 \\ 1\;0\;0\;0\;0 \\ 1\;0\;0\;0\;1 \\ 1\;0\;0\;1\;0 \\ 1\;0\;1\;1\;0 \\ 1\;1\;0\;1\;0 \\ 1\;1\;1\;1\;0 \\ 1\;1\;1\;1\;1 \\ \sim\;\sim\;1\;0\;0 \end{vmatrix}=\begin{vmatrix} 0\;\sim\;\sim\;\sim\;1 \\ 0\;0\;0\;1\;\sim \\ 0\;1\;1\;1\;\sim \\ 1\;0\;0\;0\;\sim \\ 1\;\sim\;\sim\;1\;0 \\ 1\;1\;1\;1\;\sim \\ \sim\;\sim\;1\;0\;0 \end{vmatrix}=\begin{vmatrix} 0\;\sim\;\sim\;\sim\;1 \\ 0\;0\;0\;1\;\sim \\ \sim\;1\;1\;1\;\sim \\ 1\;0\;0\;0\;\sim \\ 1\;\sim\;\sim\;1\;0 \\ \sim\;\sim\;1\;0\;0 \end{vmatrix}.$$

The operation of super-gluing of variables in the first matrix is carried out for blocks 4, 12, 20, 28, which are highlighted in red. Minimization of blocks in the second matrix, highlighted in blue, was carried out according to the protocol (17). The blocks highlighted in green are minimized by the protocol (18).

Minimized function:

$$F=\overline{x_1}x_5+\overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4+x_2x_3x_4+$$
$$+x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4}+x_1x_4\overline{x_5}+x_3\overline{x_4}\,\overline{x_5}.$$

Protocols (16), (19) give another way of minimizing a given function:

$$F=\left|\begin{array}{r|c|}1&0\,0\,0\,0\,1\\2&0\,0\,0\,1\,0\\3&0\,0\,0\,1\,1\\4&0\,0\,1\,0\,0\\5&0\,0\,1\,0\,1\\7&0\,0\,1\,1\,1\\9&0\,1\,0\,0\,1\\11&0\,1\,0\,1\,1\\12&0\,1\,1\,0\,0\\13&0\,1\,1\,0\,1\\14&0\,1\,1\,1\,0\\15&0\,1\,1\,1\,1\\16&1\,0\,0\,0\,0\\17&1\,0\,0\,0\,1\\18&1\,0\,0\,1\,0\\20&1\,0\,1\,0\,0\\22&1\,0\,1\,1\,0\\26&1\,1\,0\,1\,0\\28&1\,1\,1\,0\,0\\30&1\,1\,1\,1\,0\\31&1\,1\,1\,1\,1\end{array}\right.=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\0\sim1\,0\sim\\0\,1\,1\sim\sim\\1\,0\,0\,0\sim\\1\,0\sim\sim0\\1\sim\sim1\,0\\1\sim1\sim0\\1\,1\,1\,1\sim\end{array}\right|=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\0\sim1\,0\sim\\0\,1\,1\sim\sim\\1\,0\,0\,0\sim\\1\,0\sim\sim0\\1\sim\sim1\,0\\1\sim1\sim0\\\sim1\,1\,1\sim\end{array}\right|=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\0\sim1\,0\sim\\1\,0\,0\,0\sim\\1\,0\sim\sim0\\1\sim\sim1\,0\\1\sim1\sim0\\\sim1\,1\,1\sim\end{array}\right|=$$

$$=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\0\sim1\,0\sim\\1\,0\,0\,0\sim\\1\,0\sim\sim0\\1\sim\sim1\,0\\\sim\sim1\,0\,0\\1\sim1\sim0\\\sim1\,1\,1\sim\end{array}\right|=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\1\,0\,0\,0\sim\\1\,0\sim\sim0\\1\sim\sim1\,0\\\sim\sim1\,0\,0\\\sim1\,1\,1\sim\end{array}\right|=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\1\,0\,0\,0\sim\\1\,0\sim0\,0\\1\,0\sim1\,0\\1\sim\sim1\,0\\\sim\sim1\,0\,0\\\sim1\,1\,1\sim\end{array}\right|=\left|\begin{array}{c}0\sim\sim\sim1\\0\,0\,0\,1\sim\\1\,0\,0\,0\sim\\1\sim\sim1\,0\\\sim\sim1\,0\,0\\\sim1\,1\,1\sim\end{array}\right|.$$

Minimization of the blocks 1, 2, 3, 4, 5, 7, 9, 11, 12, 13, 14, 15 in the first matrix is carried out according to the protocol (16), and the blocks 16, 17, 18, 20, 22, 26, 28 30 31 – according to the protocol (19).

However, minimization in the second variant of the decoupling with the use of minimization protocols is complicated in comparison with the first option. It follows that in order to reduce the complexity of the Boolean function minimization by combinatorial method, it is necessary to give preference to protocols with the operation of super-gluing variables when choosing the minimization protocol.

Comparing the solutions of Examples 4 and 5, it is possible to see that the minimization procedure using the minimization protocols for 5-bit Boolean functions (Example 5) is much simpler.

Example 6. Minimize the logic function $F(x_1, x_2, x_3, x_4, x_5)$ by the combinatorial method given by the following truth table Σ(2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15, 18, 19, 20, 21, 22, 23, 16, 17, 28, 29, 30, 31):

$$F=\left|\begin{array}{r|c|}2&0\,0\,0\,1\,0\\3&0\,0\,0\,1\,1\\4&0\,0\,1\,0\,0\\5&0\,0\,1\,0\,1\\6&0\,0\,1\,1\,0\\7&0\,0\,1\,1\,1\\10&0\,1\,0\,1\,0\\11&0\,1\,0\,1\,1\\12&0\,1\,1\,0\,0\\13&0\,1\,1\,0\,1\\14&0\,1\,1\,1\,0\\15&0\,1\,1\,1\,1\\18&1\,0\,0\,1\,0\\19&1\,0\,0\,1\,1\\20&1\,0\,1\,0\,0\\21&1\,0\,1\,0\,1\\22&1\,0\,1\,1\,0\\23&1\,0\,1\,1\,1\\26&1\,1\,0\,1\,0\\27&1\,1\,0\,1\,1\\28&1\,1\,1\,0\,0\\29&1\,1\,1\,0\,1\\30&1\,1\,1\,1\,0\\31&1\,1\,1\,1\,1\end{array}\right.=\left|\begin{array}{c}\sim\sim0\,1\sim\\0\,0\,1\sim\sim\\0\,1\,1\sim\sim\\1\,0\,1\sim\sim\\1\,1\,1\sim\sim\end{array}\right|=\left|\begin{array}{c}\sim\sim0\,1\sim\\\sim\sim1\sim\sim\end{array}\right|=\left|\begin{array}{c}\sim\sim\sim1\sim\\\sim\sim1\sim\sim\end{array}\right|.$$

Minimized function:

$$F=x_3+x_4.$$

The operation of super-gluing of variables in the first matrix is carried out for blocks:
- 4–7 (highlighted in blue);
- 12–15 (highlighted in green);
- 20–23 (highlighted in brown color);
- 28–31 (highlighted in purple).

Minimization of blocks 2, 3, 10, 11, 18, 19, 26, 27 in the first matrix is carried out according to the protocol (7).

Example 7. Minimize the logical function $F(x_1, x_2, x_3, x_4, x_5)$ by the combinatorial method given by the following truth table Σ(0, 4, 5, 13, 16, 21, 22, 23, 24, 25, 28, 29, 30, 31) [5]:

$$F=\left|\begin{array}{r|c|}0&0\,0\,0\,0\,0\\4&0\,0\,1\,0\,0\\5&0\,0\,1\,0\,1\\13&0\,1\,1\,0\,1\\16&1\,0\,0\,0\,0\\21&1\,0\,1\,0\,1\\22&1\,0\,1\,1\,0\\23&1\,0\,1\,1\,1\\24&1\,1\,0\,0\,0\\25&1\,1\,0\,0\,1\\28&1\,1\,1\,0\,0\\29&1\,1\,1\,0\,1\\30&1\,1\,1\,1\,0\\31&1\,1\,1\,1\,1\end{array}\right.=\left|\begin{array}{c}\sim0\,0\,0\,0\\0\,0\,1\,0\,0\\\sim\sim1\,0\,1\\1\sim1\,1\sim\\1\,1\,0\,0\sim\\1\,1\,1\,0\,0\end{array}\right|=\left|\begin{array}{c}\sim0\,0\,0\,0\\0\,0\,1\,0\sim\\\sim\sim1\,0\,1\\1\sim1\,1\sim\\1\,1\,0\,0\sim\\1\,1\,1\,0\sim\end{array}\right|=\left|\begin{array}{c}\sim0\,0\,0\,0\\0\,0\,1\,0\sim\\\sim\sim1\,0\,1\\1\sim1\,1\sim\\1\,1\sim0\sim\end{array}\right|.$$

 ISSN 2226-3780

Minimized function:

$$F=\overline{x_2}\,\overline{x_3}\,\overline{x_4}\,\overline{x_5}+\overline{x_1}\,\overline{x_2}x_3\overline{x_4}+x_3\overline{x_4}x_5+x_1x_3x_4+x_1x_2\overline{x_4}.$$

The operation of super-gluing of variables in the first matrix is carried out for blocks 5, 13, 21, 29, highlighted in blue. The minimization of blocks 22, 23, 30, 31 in the first matrix is carried out according to the protocol, similar to protocol (6).

Table 2 presents the results of minimizing the function $F(x_1,x_2,x_3,x_4,x_5)$ by reducing the truth table [5].

Table 2

The result of minimizing the function $F(x_1,x_2,x_3,x_4,x_5)$

Minimization by the truth table reduction method
$F(x_1,x_2,x_3,x_4,x_5)=x_1x_3x_4+x_1x_2\overline{x_4}+\overline{x_1}\,\overline{x_2}\,\overline{x_4}\,\overline{x_5}+\overline{x_2}x_3\overline{x_4}x_5+x_1x_3\overline{x_4}\,\overline{x_5}$
Minimization by combinatorial method
$F(x_1,x_2,x_3,x_4,x_5)=x_1x_3x_4+x_1x_2\overline{x_4}+\overline{x_2}\,\overline{x_3}\,\overline{x_4}\,\overline{x_5}+\overline{x_1}\,\overline{x_2}x_3\overline{x_4}+x_3\overline{x_4}x_5$

In view of Table 2, we see that the combinatorial method in the last minter of the minimized the function gives one input variable less.

Example 8. Minimize the logic function by the combinatorial method given by the following truth table Σ(0, 2, 3, 5, 7, 9, 11, 13, 14, 16, 18, 24, 26, 28, 30) [15]:

$$F=\left|\begin{array}{c|ccccc} 0 & 0&0&0&0&0\\ 2 & 0&0&0&1&0\\ 3 & 0&0&0&1&1\\ 5 & 0&0&1&0&1\\ 7 & 0&0&1&1&1\\ 9 & 0&1&0&0&1\\ 11 & 0&1&0&1&1\\ 13 & 0&1&1&0&1\\ 14 & 0&1&1&1&0\\ 16 & 1&0&0&0&0\\ 18 & 1&0&0&1&0\\ 24 & 1&1&0&0&0\\ 26 & 1&1&0&1&0\\ 28 & 1&1&1&0&0\\ 30 & 1&1&1&1&0 \end{array}\right|=\left|\begin{array}{ccccc} \sim&0&0&\sim&0\\ 0&0&\sim&1&1\\ 0&\sim&1&0&1\\ 0&1&0&\sim&1\\ 0&1&1&1&0\\ 1&1&\sim&\sim&0 \end{array}\right|=\left|\begin{array}{ccccc} \sim&0&0&\sim&0\\ 0&0&\sim&1&1\\ 0&\sim&1&0&1\\ 0&1&0&\sim&1\\ \sim&1&1&1&0\\ 1&1&\sim&\sim&0 \end{array}\right|.$$

Minimized function:

$$F=\overline{x_2}\,\overline{x_3}\,\overline{x_5}+\overline{x_1}\,\overline{x_2}x_4x_5+\overline{x_1}x_3\overline{x_4}x_5+$$
$$+\overline{x_1}x_2\overline{x_3}x_5+x_2x_3x_4\overline{x_5}+x_1x_2\overline{x_5}.$$

The result of minimizing the function $F(x_1, x_2, x_3, x_4, x_5)$ with the Carnot map [15] is:

$$F=B'C'E+ABE'+A'C'DE+A'B'CE+A'BD'E+BCDE'. \quad (22)$$

However, the function (22) does not pass verification. For example, when testing the code 00000 function (22) does not give a unit. Thus, an error will be made in the expression for the minimal function (22) [15].

Example 9. Minimize the logical function $F(x_1, x_2, x_3, x_4, x_5)$ by the combinatorial method given by the following truth table: Σ(0, 2, 5, 7, 9, 11, 13, 15, 16, 18, 21, 23, 25, 27, 29, 31) [15]:

$$F=\left|\begin{array}{c|ccccc} 0 & 0&0&0&0&0\\ 2 & 0&0&0&1&0\\ 5 & 0&0&1&0&1\\ 7 & 0&0&1&1&1\\ 9 & 0&1&0&0&1\\ 11 & 0&1&0&1&1\\ 13 & 0&1&1&0&1\\ 15 & 0&1&1&1&1\\ 16 & 1&0&0&0&0\\ 18 & 1&0&0&1&0\\ 21 & 1&0&1&0&1\\ 23 & 1&0&1&1&1\\ 25 & 1&1&0&0&1\\ 27 & 1&1&0&1&1\\ 29 & 1&1&1&0&1\\ 31 & 1&1&1&1&1 \end{array}\right|=\left|\begin{array}{ccccc} \sim&0&0&\sim&0\\ \sim&\sim&1&\sim&1\\ \sim&1&0&\sim&1 \end{array}\right|=\left|\begin{array}{ccccc} \sim&0&0&\sim&0\\ \sim&\sim&1&\sim&1\\ \sim&1&\sim&\sim&1 \end{array}\right|.$$

Minimized function:

$$F=\overline{x_2}\,\overline{x_3}\,\overline{x_5}+x_3x_5+x_2x_5.$$

Minimization of blocks 5, 7, 13, 15, 21, 23, 29, 31 in the first matrix is carried out according to the protocol (7). Minimization of blocks 0, 2, 16, 18 and 9, 11, 25, 27 in the first matrix is carried out according to the protocol (10). The result of minimizing the given function by the combinatorial method and the Carnot map [15] is the same, but the process of minimizing the function $F(x_1, x_2, x_3, x_4, x_5)$ by a combinatorial method is simple.

Example 10. Minimize the logic function $F(x_1, x_2, x_3, x_4, x_5)$ by the combinatorial method given by the following truth table Σ(0, 5, 7, 11, 12, 13, 15, 16, 21, 22, 23, 24, 28, 29, 30, 31) [16]:

$$F=\left|\begin{array}{c|ccccc} 0 & 0&0&0&0&0\\ 5 & 0&0&1&0&1\\ 7 & 0&0&1&1&1\\ 11 & 0&1&0&1&1\\ 12 & 0&1&1&0&0\\ 13 & 0&1&1&0&1\\ 15 & 0&1&1&1&1\\ 16 & 1&0&0&0&0\\ 21 & 1&0&1&0&1\\ 22 & 1&0&1&1&0\\ 23 & 1&0&1&1&1\\ 24 & 1&1&0&0&0\\ 28 & 1&1&1&0&0\\ 29 & 1&1&1&0&1\\ 30 & 1&1&1&1&0\\ 31 & 1&1&1&1&1 \end{array}\right|=\left|\begin{array}{ccccc} \sim&0&0&0&0\\ 1&\sim&0&0&0\\ \sim&\sim&1&\sim&1\\ 0&1&0&1&1\\ \sim&1&1&0&0\\ 1&\sim&1&1&0 \end{array}\right|=\left|\begin{array}{ccccc} \sim&0&0&0&0\\ 1&\sim&0&0&0\\ \sim&\sim&1&\sim&1\\ 0&1&\sim&1&1\\ \sim&1&1&0&\sim\\ 1&\sim&1&1&\sim \end{array}\right|.$$

Minimization of blocks 5, 7, 13, 15, 21, 23, 29, 31 in the first matrix is carried out according to the protocol (6). Minimization of blocks 0, 16, 24 in the first matrix is carried out according to the protocol (21). The minimization of blocks 12, 28 and 22, 30 in the first matrix is carried out by simple gluing.

The obtained minimal function (the last matrix) admits an increase in the number of variables with the value of unity:

$$\begin{vmatrix} \sim & 0 & 0 & 0 & 0 \\ 1 & \sim & 0 & 0 & 0 \\ \sim & \sim & 1 & \sim & 1 \\ 0 & 1 & \sim & 1 & 1 \\ \sim & 1 & 1 & 0 & \sim \\ 1 & \sim & 1 & 1 & \sim \end{vmatrix} = \begin{vmatrix} \sim & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ \sim & \sim & 1 & \sim & 1 \\ 0 & 1 & \sim & 1 & 1 \\ \sim & 1 & 1 & 0 & \sim \\ 1 & \sim & 1 & 1 & \sim \end{vmatrix} = \begin{vmatrix} \sim & 0 & 0 & 0 & 0 \\ 1 & 1 & \sim & 0 & 0 \\ \sim & \sim & 1 & \sim & 1 \\ 0 & 1 & \sim & 1 & 1 \\ \sim & 1 & 1 & 0 & \sim \\ 1 & \sim & 1 & 1 & \sim \end{vmatrix}.$$

Minimized function:

$$F = \overline{x_2}\,\overline{x_3}\,\overline{x_4}\,\overline{x_5} + x_1 x_2 \overline{x_4}\,\overline{x_5} + x_3 x_5 + \overline{x_1} x_2 x_4 x_5 + x_2 x_3 \overline{x_4} + x_1 x_3 x_4.$$

The result of minimizing the function $F(x_1,x_2,x_3,x_4,x_5)$ by the combinatorial method and the Carnot map [16] is the same.

7. SWOT analysis of research results

Strengths. Strengths of using protocols of minimization of 5-bit Boolean functions of the combinative method may be considered as decreasing the complexity of the process of function minimization that advantageously distinguishes the combinative method comparing with analogues by the following factors:

– increase of the productivity of intellectual labor (intellectual component) at minimization of 5-bit Boolean functions that favors the increase of the scientific level of studying protocols of minimization, improvement of the algorithm of minimization of logic functions, widening of control functions and deeper understanding of logic transformations;

– less cost of the development and introduction at the expanse of shortening the need in applying hardware and software means of automation.

In comparison with the algorithm for Boolean functions minimization in [1], in this paper the object of solving the problem is new protocols for 5-bit Boolean functions minimization by combinatorial method. The effectiveness of the application of the developed protocols is demonstrated by comparing examples 4 and 5 of the Boolean functions minimization without using minimization protocols and with minimization protocols in Section 6 «Research results». Taking into account these examples, it follows that the use of protocols 5-bit Boolean functions minimization greatly simplifies the procedure for Boolean functions minimization, without losing its functionality.

The algorithm for Boolean functions minimization by a combinatorial method is based on a block-diagram with a repetition, what is the truth table of this function. This allow to concentrate the minimization principle within the function calculation protocol (within the truth table of the function) and thus dispense with auxiliary objects like the Carnot map, the Weich diagram, the acyclic graph, the cubic representation, and so on.

Equivalent transformations by graphic images in the form of two-dimensional matrices have large information capacity due to tabular organization of the mathematical apparatus of the block-diagram with repetition. This allows to obtain more information about the orthogonality, contiguity, uniqueness of truth table blocks (combinatorial system). Therefore, equivalent transformations by graphic images can effectively replace, in particular, verbal procedures of algebraic transformations (example 1), minimization by the method of synthesis of the infimum disjunctive normal form (IDNF) of a logical function, using the perfect matrix arrangement of the n-dimensional cube (example 2). A comparison of the combinatorial method with other methods for Boolean functions minimization is presented in [1, 2].

Weaknesses. The weak side of the application of the minimization protocols for the 5-bit Boolean functions of the combinatorial method is related to the small practice of applying methods of minimizing various variants of 5-bit Boolean functions. Negative internal factors are inherent in the protocols of 5-bit Boolean functions minimization by combinatorial method consist in increasing the time of obtaining the minimum function with insufficient library of protocols for 5-bit Boolean functions minimization.

Opportunities. The opportunities of further research into the protocols for 5-bit Boolean functions minimization of the combinatorial method is the search for new protocols, the order of their application in order to improve the accuracy of solving the task of minimizing the function within a certain time. Additional features that the implementation of minimization protocols of the combinatorial method can bring are the creation and support of a library of protocols minimizing 5-bit Boolean functions as standard procedures in order to optimize the search for minimal logical functions.

Threats. The protocols for 5-bit Boolean functions minimization of the combinatorial method are independent of the protocols of other minimization methods, so there is no threat of negative impact on the object of research of external factors. An analogue of the combinatorial method for minimizing the 5-bit Boolean function is the algebraic method [17]. The algebraic method of 5-bit Boolean functions minimization is best because for it there are pre-established simplification laws, discovered properties, and algorithms for minimizing Boolean functions. However, the algebraic method is a verbal procedure of operational transformations, gives a smaller effect of the quality of minimization compared to the image transformations of the combinatorial method.

8. Conclusions

1. It is established that the implementation of the minimization protocols for the 5-bit Boolean functions of the combinatorial method makes it possible to simplify the procedure for minimizing the 5-bit Boolean function without losing its functionality.

2. It is shown that the minimization protocols for 5-bit Boolean functions support the function minimization if there is a complete binary combinatorial system with repetition or an incomplete binary combinatorial system with repetition in the truth table structure. The use of minimization protocols is most effective in the presence of a complete binary combinatorial system with repetition. The effectiveness of minimization protocols in the presence of incomplete binary combinatorial system with repetition decreases not significantly.

ISSN 2226-3780

3. It is established that the results of verification of minimized functions obtained using the minimization protocols of the 5-bit Boolean functions of the combinatorial method satisfy the original truth table of this function and, therefore, show an optimal decrease in the number of function variables without losing its functionality.

4. The effectiveness of the application of minimization protocols for the 5-bit Boolean functions of the combinatorial method is demonstrated by examples of minimization of functions borrowed from the work of other authors for the purpose of comparison:

– example 7 [5];
– example 8 [15];
– example 9 [15];
– example 10 [16].

Taking into account the mentioned examples and examples 4, 5 and 6 of the application of the minimization protocols for the 5-bit Boolean functions of the combinatorial method, it is reasonable to apply them in the minimization of logical functions.

References

1. Riznyk V., Solomko M. Minimization of Boolean functions by combinatorial method // Technology Audit and Production Reserves. 2017. Vol. 4, Issue 2 (36). P. 49–64. doi: http://doi.org/10.15587/2312-8372.2017.108532
2. Riznyk V., Solomko M. Application of super-sticking algebraic operation of variables for Boolean functions minimization by combinatorial method // Technology Audit and Production Reserves. 2017. Vol. 6, Issue 2 (38). P. 60–76. doi: http://doi.org/10.15587/2312-8372.2017.118336
3. Manojlovic V. Minimization of Switching Functions using Quine-McCluskey Method // International Journal of Computer Applications. 2013. Vol. 82, Issue 4. P. 12–16. doi: http://doi.org/10.5120/14103-2127
4. Rytsar B. The Minimization Method of Boolean Functionns in Polynomial Set-theoretical Format: conf. At Rzeszow, 2015. P. 130–146. URL: http://ceur-ws.org/Vol-1492/Paper_37.pdf
5. Rathore T. S. Minimal Realizations of Logic Functions Using Truth Table Method with Distributed Simplification // IETE Journal of Education. 2014. Vol. 55, Issue 1. P. 26–32. doi: http://doi.org/10.1080/09747338.2014.921412
6. Dan R. Software for The Minimization of The Combinational Logic Functions // The Romanian Review Precision Mechanics, Optics & Mechatronics. 2010. Vol. 20, Issue 37. P. 95–99. URL: https://www.researchgate.net/publication/268270733_Software_for_The_Minimization_of_The_Combinational_Logic_Functions_SOFTWARE_FOR_THE_MINIMIZATION_OF_THE_COMBINATIONAL_LOGIC_FUNCTIONS
7. Zolfaghari B., Sheidaeian H. A New Case for Image Compression Using Logic Function Minimization // The International Journal of Multimedia & Its Applications. 2011. Vol. 3, Issue 2. P. 45–62. doi: http://doi.org/10.5121/ijma.2011.3204
8. Nosrati M., Karimi R., Nariri M. Minimization of boolean functions using genetic algorithm // Anale. Seria Informatica. 2012. Vol. X, Issue 1. P. 73–77. URL: http://anale-informatica.tibiscus.ro/download/lucrari/10-1-08-Nosrati.pdf
9. Nosrati M., Karimi R. An Algorithm for Minimizing of Boolean Functions Based on Graph DS // World Applied Programming. 2011. Vol. 1, Issue 3. P. 209–214.
10. Bernasconi A., Ciriani V. Three-Level Logic Minimization Based on Function Regularities // IEEE Transactions on computer-aided design of integrated circuits and systems. 2003. Vol. 22, Issue 8. P. 1005–1016. URL: http://cs.ecs.baylor.edu/~maurer/CSI5346/ThreeLevelMin.pdf
11. Solairaju A., Periyasamy R. Optimal Boolean Function Simplification through K-Map using Object-Oriented Algorithm // International Journal of Computer Applications. 2011. Vol. 15, Issue 7. P. 28–32. doi: http://doi.org/10.5120/1959-2621
12. Mohana Ranga Rao R. An Innovative procedure to minimize Boolean function // International journal of advanced engineering sciences and technologies. 2011. Vol. 3, Issue 1. P. 12–14.
13. Zakrevskiy A. D., Pottosin Yu. V., Cheremisinova L. D. Logicheskie osnovy proektirovaniya diskretnykh ustroystv. Matematika. Prikladnaya matematika. Moscow: Fizmatlit, 2007. 592 p. URL: https://www.twirpx.com/file/2197687/
14. Ivanov Yu. D., Zakharova O. S. Algoritm sinteza infimumnykh diz'yuktivnykh normal'nykh form logicheskikh funktsiy // Trudy Odesskogo politekhnicheskogo universiteta. 2004. Issue 2 (22). P. 1–7. URL: http://pratsi.opu.ua/app/webroot/articles/1313074117.pdf
15. Simplification and minimization of boolean functions. URL: https://shubh977.files.wordpress.com/2017/02/chap4.pdf (Last accessed: 15.07.2017)
16. Sudnitson A. Diskretnaya matematika. Minimizatsiya bulevykh funktsiy. Primer iz knigi A. D. Zakrevskogo. 2008. P. 11. URL: http://ati.ttu.ee/~alsu/DM%20_MinBF_2008_lecture.pdf (Last accessed: 15.07.2017)
17. Martyniuk O. M. Osnovy dyskretnoi matematyky. Konspekt lektsii. Odeskyi natsionalnyi politekhnichnyi universytet, Nauka i tekhnika, 2008. 300 p.

Riznyk Volodymyr, *Doctor of Technical Sciences, Professor, Department of Control Aided Systems, Lviv Polytechnic National University, Ukraine, e-mail: rvv@polynet.lviv.ua*

Solomko Mykhailo, *PhD, Associate Professor, Department of Computer Engineering, National University of Water and Environmental Engineering, Rivne, Ukraine, e-mail: doctrinas@ukr.net*