**Riznyk V.,
Solomko M.**

MINIMIZATION OF CONJUNCTIVE NORMAL FORMS OF BOOLEAN FUNCTIONS BY COMBINATORIAL METHOD

Об'єктом дослідження є комбінаторний метод мінімізації кон'юктивних нормальних форм (КНФ) булевих функцій з метою зменшення його алгоритмічної складності. Одним з найпроблемніших місць мінімізації КНФ булевих функцій є складність алгоритму мінімізації та гарантія отримання мінімальної функції.

У ході дослідження використовувався метод рівносильних образних перетворень, який ґрунтується на законах та аксіомах алгебри логіки, протоколи мінімізації КНФ булевих функцій.

Отримано зменшення обчислювальної складності процесу мінімізації КНФ булевих функцій комбінаторним методом за новими встановленими критеріями, завдяки використанню ряду особливостей алгоритму пошуку мінімальних диз'юнктивних нормальних форм (ДНФ) та КНФ логічних функцій, зокрема:

– застосування математичного апарату перетворення блок-схем з повторенням дає змогу збільшити інформаційну компоненту образного перетворення стосовно ортогональності, суміжності, однозначності блоків таблиці істинності;

– рівносильні образні перетворення дозволяють з ефектом замінити вербальні процедури алгебричних перетворень за рахунок більшої інформаційної ємності матричних образів;

– результат мінімізації оцінюється за ознакою мінімальної функції;

– мінімальні ДНФ або КНФ функції отримуються незалежно від нормальної форми заданої логічної функції;

– протоколи мінімізації КНФ булевих функцій складають бібліотеку протоколів для процесу мінімізації КНФ булевих функцій як стандартні процедури.

Завдяки вищевикладеному забезпечується можливість оптимального зменшення кількості змінних заданої функції без втрати її функціональності. Ефективність застосування образних перетворень демонструється прикладами мінімізації функцій, запозичених з інших методів з метою порівняння.

У порівнянні з аналогічними відомими методами мінімізації булевих функцій запропонований метод дозволяє:

– зменшити алгоритмічну складність мінімізації КНФ булевих функцій;

– збільшити наочність процесу мінімізації ДНФ або КНФ булевих функцій;

– забезпечити самодостатність комбінаторного методу мінімізації булевих функцій за рахунок впровадження ознаки мінімальної функції та мінімізації на повній таблиці ДНФ і КНФ.

Ключові слова: *мінімізація кон'юктивних нормальних форм, комбінаторний метод мінімізації булевих функцій, блок-схема з повторенням.*

1. Introduction

The problems and shortcomings of the known methods of minimizing conjunctive normal forms (CNF) of Boolean functions are associated with an increase in the amount of computation with an increase in the number of variable logical functions. The complexity of the minimization problem for Boolean functions of n variables in the CNF class with increasing n increases exponentially. The complexity of the algebraic method and the complexity of minimizing the CNF of a logical function by the Karnaugh map increases markedly with an increase in the number of variables by more than four or five; therefore, it is not advisable to use these methods with a large number of variables.

The task of Boolean functions minimization $F = F(x_1, x_2 \ldots x_n)$ in the CNF class is formulated as follows: for a Boolean function of n variables F, find the CNF of the functions by the minimum possible number of CNF factors or with the minimum possible number of input literals (MCNF).

In [1–3] the combinatorial method of Boolean functions minimization in the class of disjunctive normal forms (DNF) was considered. The features of the method consist in the greater informativeness of the process of minimization compared to the algebraic method of Boolean functions minimization, due to tabular organization and the introduction of the apparatus of figurative transformations.

This paper presents a combinatorial method for Boolean functions minimization in the CNF class of logic functions. The use of the method of figurative transformations to simplify the CNF of the functions gives new rules for the algebra of logic, establishes the sign of the minimal function.

The evolution of methods for simplifying logical functions is the result of relentless optimization, so research remains relevant, in particular, to the improvement of such factors as:

– methodology of minimization of logical functions in the class of DNF and CNF;

– establishing signs of minimal function;

– the cost of technology to minimize logical functions.

2. The object of research and its technological audit

The object of research is the figurative transformations of the combinatorial method, to minimize the CNF of Boolean functions, which are applied when there is a complete or

incomplete binary combinatorial systems with repetition in the structure of the truth table [1–3].

The figurative transformations of the combinatorial method make up a library of protocols for the process of minimizing CNF of Boolean functions as standard procedures, therefore the use of a separate such protocol is reduced to performing a single algebraic transformation. Equivalent figurative transformations of the combinatorial method by their properties have a greater information capacity, therefore they are capable of replacing verbal procedures of minimizing CNF with the Boolean functions of algebraic transformations with effect.

The effectiveness of the use of figurative transformations of the combinatorial method for minimizing CNF of Boolean functions is significantly reducing the complexity of the procedure for reducing logical functions. This allows to do without hardware and software tools to automate the process of reducing the Boolean functions to 10 variables.

Reducing the complexity of the process of simplifying the CNF of Boolean functions by the combinatorial method, in particular, adds efficiency in the practical establishment and operation of the sign of the minimal function.

The disadvantages of the use of figurative transformations of the combinatorial method while minimizing CNF of the Boolean functions are associated with a small amount of existing theoretical developments. The prospect of using the combinatorial method for minimization of a CNF of logical functions is based on practical chances of optimal minimization of a CNF of logical functions. With an increase in the number of variables (computation time), in order to minimize the CNF of the function by the combinatorial method, it is necessary to search for new protocols for minimizing CNF of the Boolean functions and expanding the library of the specified protocols.

3. The aim and objectives of research

The aim of research is simplification of the process of minimizing the CNF of the Boolean functions by the combinatorial method. To achieve this aim it is necessary to solve the following tasks:

1. To establish the adequacy of the application of the combinatorial method to minimize the CNF of the Boolean functions.

2. To determine equivalent figurative transformations of a disjunctive maxterms to minimize the CNF of Boolean functions.

3. To establish the feasibility of the use of figurative transformations to obtain the signs of the minimum logical function.

4. To establish the feasibility of using figurative transformations to minimize the two normal forms – DNF and CNF of a given Boolean function, using the full truth table.

4. Research of existing solutions of the problem

The relationship between the CNF representations of a given Boolean function and the essential sets of the implicant is studied in [4]. It is known that every representation of a CNF of the function and every essential set of implicant must intersect. Therefore, the maximum number of pairwise essential sets, do not intersect, give the lower limit of the size of any representation of the CNF of a logical function. In [4], the lower limit of the minimum size of a CNF of a given function is studied. The lower bound is given by the expression that indicates the number of pairwise essential implicant sets that do not overlap. Functions for which this lower bound corresponds to the minimum size of a CNF, called «covering functions». The polynomial complexity of solving this problem of minimizing CNF of Boolean functions is shown. This minimization problem has many practical applications. For example, for artificial intelligence, this problem is equivalent to finding the most compact representation of a given knowledge base. Such transformation of the knowledge base ensures the compression of knowledge, since the actual knowledge does not change, and the size of the presentation can be significantly reduced.

The problem of minimizing CNF of Boolean functions is considered in [5], where generalizations of a large class of CNF formulas and their minimization in polynomial time are presented.

The method of decomposition of Boolean functions, which can be applied in some cases to the CNF formula, when it is necessary to bring its minimum, is considered in [6], where examples of such an approach to minimizing CNF of a Boolean function are presented.

In [7], the minimization of CNF of Boolean functions in polynomial time is considered and the complexity of solving this problem is analyzed. It is known that the question of whether there is a shorter CNF for a function defined as a CNF has n^2 complexity for general formulas, although for certain classes of formulas the complexity of minimizing a CNF may be different.

The discussion on the role of autosymmetry degree of variables of a Boolean function and why it deserves attention to minimize the logical function is presented in [8]. The regularity of variables of a Boolean function can be expressed by the degree of autosymmetry, which ultimately gives a new tool for effective minimization.

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

A comprehensive survey of methods for minimizing logical functions is demonstrated in [10]. These methods are examined by their purpose, methodology, implementation, and benefits. A comparison of the reviewed approaches to the minimization of logical functions is presented.

A new technique of two-stage process of combinational logic optimization is described in [11]. Such technique can be applied to arbitrary combinational logic problems, and often gives an improvement even after optimization by standard methods. This optimization technique is used to improve software performance.

Boolean functions minimization using a triple tree, in which basic Boolean operations are applied, is considered in [12]. The method is designed to minimize Boolean functions with a large number of variables with the support of minimizing incompletely specified functions.

Minimization of conjunctive normal form of partial monotone Boolean functions for k-valued logic is considered in [13], and it is proved in [14] that conjunctive normal forms of partial monotone Boolean functions of 2-valued logic can be effectively minimized using partial monotone disjuncts. A Boolean function is called partially monotone

if it is monotone with respect to some of its arguments and anti-monotone with respect to its other arguments.

In [14], it is proved that conjunctive normal forms of partially monotonic Boolean functions of 2-valued logic can be minimized very effectively using only partially monotonic disjuncts. A Boolean function is called partially monotone if it is monotone with respect to some of its arguments and anti-monotone with respect to its other arguments.

Unlike the above-mentioned sources, in this work, the object of solving the problem is minimizing CNF of the Boolean functions by the combinatorial method if there is a complete or incomplete binary combinatorial system with repetition in the structure of the truth table. The mathematical apparatus of the block diagram with repetition makes it possible to obtain more information about the orthogonality, adjacency, uniqueness of the truth table blocks. Equivalent figurative transformations in the form of two-dimensional matrices by their properties have a large information capacity, therefore, they are able to effectively replace the verbal procedures of algebraic transformations.

5. Research results

5.1. Equivalent transformations of CNF of Boolean functions. The rules for simplifying the CNF of logical functions are based on the associative (1), (2), commutative (3), (4) and distributive (5), (6) laws of the algebra of logic.

Associative laws:

$$x_1 + x_2 + x_3 = x_1 + (x_2 + x_3) = x_2 + (x_1 + x_3) =$$
$$= x_3 + (x_1 + x_2), \tag{1}$$

$$x_1 x_2 x_3 = x_1 \cdot (x_2 \cdot x_3) = x_2 \cdot (x_1 \cdot x_3) = x_3 \cdot (x_1 \cdot x_2). \tag{2}$$

Commutative laws:

$$x_1 + x_2 = x_2 + x_1, \tag{3}$$

$$x_1 x_2 = x_2 \cdot x_1. \tag{4}$$

Commutative laws are valid for disjunctions and conjunctions of any number of variables.

Distributive laws:

a) distributivity of the conjunction with respect to disjunction (distributive law of the 1st kind):

$$x_1(x_2 + x_3) = x_1 \cdot x_2 + x_1 \cdot x_3; \tag{5}$$

b) distributivity of disjunction with respect to conjunction (distributive law of the 2nd kind):

$$x_1 + (x_2 \cdot x_3) = (x_1 + x_2)(x_1 + x_3). \tag{6}$$

It is also easy to verify the validity of de Morgan's laws (laws of inversion):

$$\overline{x_1 \cdot x_2 \cdot \ldots \cdot x_m} = \overline{x_1} + \overline{x_2} + \ldots + \overline{x_m};$$
$$\overline{x_1 + x_2 + \ldots + x_m} = \overline{x_1} \cdot \overline{x_2} \cdot \ldots \cdot \overline{x_m}.$$

When performing logical operations in expressions, it is necessary to observe the following rules:

1) if only identical operations are present in the expression, they must be performed in the order in which they are written;

2) if in the expression there are various operations, then it is necessary first to carry out operations of inversion, then – conjunction and, finally, – disjunction.

Brackets are used to change the priorities of the logical operations.

In the general case, when minimizing the CNF of the Boolean functions by the combinatorial method, the following rules of the algebra of logic are used:

The rule of gluing of variables for a CNF of logical expression.

A conjunction of two adjacent elementary clauses of some rank p is replaced by one elementary clause of rank $p-1$, and is a common part of the initial operands of the conjunction. This rule is a consequence of the distribution law of the 2nd kind:

$$(x_1 + x_2 + \overline{x_3} + \overline{x_4})(x_1 + x_2 + \overline{x_3} + x_4) = x_1 + x_2 + \overline{x_3}; \tag{7}$$

$$(\overline{x_1} + x_2)(x_1 + x_2) = x_2. \tag{8}$$

Equivalent figurative transformations of the combinatorial method for the gluing rule of CNF of a logical expression (7), (8) have an illustration of the images (9), (10) respectively:

$$\begin{vmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{vmatrix} = |1 \quad 1 \quad 0 \quad \sim|; \tag{9}$$

$$\begin{vmatrix} 0 & 1 \\ 1 & 1 \end{vmatrix} = |\sim \quad 1|. \tag{10}$$

The semi-gluing rule of the variables for a CNF of logical expression.

$$(x_1 + x_2)(x_1 + \overline{x_2} + x_3) = (x_1 + x_2)(x_1 + x_3). \tag{11}$$

For proof, open the brackets:

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

The result will again be presented to CNF:

$$x_1 + x_2 x_3 = x_1(x_2 + x_3) = x_1 x_2 + x_1 x_3 = (x_1 + x_2)(x_1 + x_3).$$

So

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

Equivalent figurative transformations of the combinatorial method for the semi-gluing rule of CNF of logical expression (11) have an illustration of the image:

$$\begin{vmatrix} 1 & 1 & \\ 1 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 1 & \\ 1 & & 1 \end{vmatrix}. \tag{12}$$

The rule of generalized gluing of variables for a CNF of logical expression:

$$(x_1 + x_3)(x_2 + \overline{x_3})(x_1 + x_2) = (x_1 + x_3)(x_2 + \overline{x_3}); \tag{13}$$

$$(x_1 + x_3)(x_2 + \overline{x_3}) = (x_1 + x_3)(x_2 + \overline{x_3})(x_1 + x_2). \tag{14}$$

Equivalent figurative transformations of the combinatorial method for the rule of the generalized gluing of a CNF of a logical expression (13), (14) have an illustration of the images (15), (16) respectively:

$$\begin{vmatrix} 1 & & 1 \\ & 1 & 0 \\ 1 & 1 & \end{vmatrix} = \begin{vmatrix} 1 & & 1 \\ & 1 & 0 \end{vmatrix}; \tag{15}$$

$$\begin{vmatrix} 1 & & 1 \\ & 1 & 0 \end{vmatrix} = \begin{vmatrix} 1 & & 1 \\ & 1 & 0 \\ 1 & 1 & \end{vmatrix}. \tag{16}$$

The variable absorption rule for a CNF of logical expression.
A conjunction of two elementary disjunctions of various ranks, one of which is its own part of the other, is replaced by a disjunction that has a lower rank. This rule is a consequence of the distribution law of the 2nd kind:

$$(x_1 + x_2 + \overline{x_3} + x_4)(\overline{x_3} + x_4) = \overline{x_3} + x_4; \tag{17}$$

$$\overline{x_2} \cdot (x_1 + \overline{x_2}) = \overline{x_2}. \tag{18}$$

Equivalent figurative transformations of the combinatorial method for the absorption rule of a CNF of a logical expression (17), (18) have an illustration of the images (19), (20), respectively:

$$\begin{vmatrix} 1 & 1 & 0 & 1 \\ & & 0 & 1 \end{vmatrix} = |\sim \ \sim \ 0 \ 1|; \tag{19}$$

$$\begin{vmatrix} & 0 \\ 1 & 0 \end{vmatrix} = |\sim \ 0|. \tag{20}$$

The idempotence rule for variables for a CNF of logical expression:

$$(x_1 + x_2)(x_1 + x_2) = x_1 + x_2. \tag{21}$$

For proof, open the brackets:

$$(x_1 + x_2)(x_1 + x_2) = x_1 + x_1 x_2 + x_1 x_2 + x_2 = x_1 + x_2.$$

The equivalent figurative transformations of the combinatorial method for the idempotence rule of the CNF of logical expression (21) have an illustration of the image:

$$\begin{vmatrix} 1 & 1 \\ 1 & 1 \end{vmatrix} = |1 \ 1|. \tag{22}$$

If in conjunctive normal form (CNF) of logical function:

$$F = (\overline{x_1} + \overline{x_2} + \overline{x_3})(\overline{x_1} + \overline{x_2} + x_3)(\overline{x_1} + x_2 + x_3) \times$$
$$\times (x_1 + \overline{x_2} + \overline{x_3})(x_1 + \overline{x_2} + x_3)(x_1 + x_2 + x_3), \tag{23}$$

variables with inversion replace with «0», and variables without inversion replace with «1», then obtain the binary equivalent of the expression of a logical function:

$$F = (0_1 + 0_2 + 0_3)(0_1 + 0_2 + 1_3)(0_1 + 1_2 + 1_3) \times$$
$$\times (1_1 + 0_2 + 0_3)(1_1 + 0_2 + 1)(1_1 + 1_2 + 1_3). \tag{24}$$

The expression (24) is represented by the matrix:

$$F = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{vmatrix}. \tag{25}$$

Disjunctive normal form (DNF) of logic function:

$$F = \overline{x_1} \ \overline{x_2} \ \overline{x_3} + \overline{x_1} \ \overline{x_2} x_3 + \overline{x_1} x_2 x_3 +$$
$$+ x_1 \overline{x_2} \ \overline{x_3} + x_1 \overline{x_2} x_3 + x_1 x_2 x_3, \tag{26}$$

can be represented by binary codes:

$$F = 000 + 001 + 011 + 100 + 101 + 111, \tag{27}$$

or matrix:

$$F = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{vmatrix}. \tag{28}$$

Considering (25) and (28), it can be seen that the CNF and DNF of logical functions are represented by matrices with the same combinatorial structures. The difference between these matrices lies in the hermeneutics of logical operations. The matrix (25), which reflects the CNF of a logical function, provides the disjunctive maxterm of the function and conjunction operations for them. The matrix (28), which reflects the DNF of a logical function, provides the minterm of the function and disjunction operations for them.

Since the matrix images provide more information about the orthogonality, adjacency, uniqueness of the combinatorial system blocks (25) and (28), which are the truth tables of the given logical functions, using them to search for objects equivalent transformations is effective in the process of simplifying logical functions.

5.2. Simplification of Boolean functions by combinatorial method. Equivalent transformations of the structure of Boolean functions, changing only its form, not its value, allow to obtain a simplified scheme of a combinational device. The transformation of the structure of a Boolean function in order to simplify the combinational device is called its *minimization*.

Example 1. Simplify expression:

$$f = ad + ab + \overline{a}c + \overline{b}cd. \tag{29}$$

According to the law of generalized gluing:

$$ad + \overline{a}c = ad + \overline{a}c + cd,$$

so

$$f = ad + ab + \overline{a}c + cd + \overline{b}cd,$$

whence

$$f = ad + ab + \bar{a}c + cd(1 + \bar{b}) = ad + ab + \bar{a}c + cd.$$

The generalized gluing law is again applicable:

$$ad + \bar{a}c + cd = ad + \bar{a}c,$$

and finally:

$$f = ad + ab + \bar{a}c.$$

Simplification of algebraic expressions (29) by figurative transformations looks like this:

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

Simplified form of the function:

$$f = ad + ab + \bar{a}c.$$

During the simplification of the expression (29), the identity (highlighted in red) and the law of absorption are applied by figurative transformations.

Example 2. Simplify expression:

$$
\begin{aligned}
y &= x_1 x_2 x_3 + \overline{x_1} x_2 x_3 + \overline{x_1}\,\overline{x_2} x_3 + \overline{x_1} x_2 \overline{x_3} + \overline{x_1}\,\overline{x_2}\,\overline{x_3} = \\
&= x_1 x_2 x_3 + \overline{x_1} x_2 (x_3 + \overline{x_3}) + \overline{x_1}\,\overline{x_2}(x_3 + \overline{x_3}) = \\
&= x_1 x_2 x_3 + \overline{x_1} x_2 + \overline{x_1}\,\overline{x_2} = \\
&= x_1 x_2 x_3 + \overline{x_1}(x_2 + \overline{x_2}) = \\
&= x_1 x_2 x_3 + \overline{x_1} = \\
&= x_2 x_3 + \overline{x_1}.
\end{aligned}
\tag{30}
$$

Simplification of algebraic expressions (30) by figurative transformations has the form:

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

Simplified form of the function:

$$y = \overline{x_1} + x_2 x_3.$$

During the simplification of expression (30), the law of super-gluing of variables [2] (highlighted in red) and semi-gluing of variables is applied by figurative transformations.

Example 3. Simplify expression:

$$
\begin{aligned}
y &= \overline{x_1}\,\overline{x_2}\,\overline{x_3} + \overline{x_1}\,\overline{x_2} x_3 + \overline{x_1} x_2 x_3 + \\
&+ x_1 \overline{x_2}\,\overline{x_3} + x_1 \overline{x_2} x_3 + x_1 x_2 x_3.
\end{aligned}
\tag{31}
$$

Let's glue the first and fourth, second and fifth, third and sixth minterms:

$$y = \overline{x_2}\,\overline{x_3} + \overline{x_2} x_3 + x_2 x_3.$$

Let's continue to simplify the expression. Apply the next axiom to the second minterm:

$$\overline{x_2} x_3 = \overline{x_2} x_3 + \overline{x_2} x_3,$$
$$\tag{32}$$

and substitute in the result, then:

$$
\begin{aligned}
y &= \left(\overline{x_2}\,\overline{x_3} + \overline{x_2} x_3\right) + \left(\overline{x_2} x_3 + x_2 x_3\right) = \\
&= \overline{x_2}\left(\overline{x_3} + x_3\right) + x_3\left(\overline{x_2} + x_2\right) = \overline{x_2} + x_3.
\end{aligned}
$$

This is the desired simplification of the expression (31). To obtain it again the law of gluing is applied.

Simplification of algebraic expressions (31) by figurative transformations has the form:

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

Simplified form of the function:

$$y = \overline{x_2} + x_3.$$

During the simplification of expression (31), the law of super-gluing of variables [2] (highlighted in red), simple gluing (highlighted in blue), and the semi-gluing of variables are applied by figurative transformations.

Another way to simplify the expression (31) by figurative transformations is:

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

The result of the simplification of the expression (31) in two variants is the same.

The results of simplifying the Boolean functions in examples 1–3 by figurative transformations coincide with the result of simplification obtained using the algebraic method, but the process of simplifying functions by figurative transformations is simple.

5.3. Nelson method. The method allows to obtain a reduced DNF of the Boolean function *F* with its arbitrary CNF. The algorithm of the method is reduced to the opening of the brackets of an arbitrary CNF of the Boolean function *F* followed by all acquisitions. The result will be a shortened DNF of the Boolean function *F*.

Example 4. Find the simplified DNF of the function *F* in CNF by the Nelson method:

$$F = (x_1 + \overline{x_2})(\overline{x_1} + x_3)(x_1 + x_2 + \overline{x_3}).$$
$$\tag{33}$$

After opening the brackets:

$$
\begin{aligned}
F &= (x_1 x_3 + \overline{x_1}\,\overline{x_2} + \overline{x_2} x_3)(x_1 + x_2 + \overline{x_3}) = \\
&= x_1 x_3 + x_1 x_2 x_3 + \overline{x_1}\,\overline{x_2}\,\overline{x_3} + x_1 \overline{x_2} x_3).
\end{aligned}
\tag{34}
$$

After all the acquisitions, let's obtain the simplified DNF of the function F:

$$F = x_1 x_3 + \overline{x_1}\,\overline{x_2}\,\overline{x_3}. \tag{35}$$

Let's note that to simplify the Boolean function (34), it is possible to apply figurative transformations:

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

Simplified form of the function:

$$F = x_1 x_3 + \overline{x_1}\,\overline{x_2}\,\overline{x_3}. \tag{36}$$

In order to simplify the function (34), absorption is applied twice through figurative transformations. The result of simplification (36) obtained by figurative transformations coincides with the result (35) obtained by the Nelson method.

6. Research results

When the CNF is minimized by the algebraic method, it is often enough (but not always!) it is possible to get the best results if increase the given CNF using the idempotence property of disjunction $xx = x$.

Example 5. Minimize the CNF of a function given by perfect conjunctive normal form (PCNF):

$$F(x_1,x_2,x_3) = (\overline{x_1}+x_2+x_3)(\overline{x_1}+\overline{x_2}+x_3)(\overline{x_1}+\overline{x_2}+\overline{x_3}). \tag{37}$$

First, let's minimize the function $F(x_1,x_2,x_3)$ by applying the gluing laws. Choose one of the possible options for gluing of variables, for example (disjunctions that can be glued underlined):

$$F(x_1,x_2,x_3) = \underline{(\overline{x_1}+x_2+x_3)(\overline{x_1}+\overline{x_2}+x_3)}(\overline{x_1}+\overline{x_2}+\overline{x_3}),$$

and minimize CNF:

$$F(x_1,x_2,x_3) = (\overline{x_1}+x_3)(\overline{x_1}+\overline{x_2}+\overline{x_3}). \tag{38}$$

Add the second disjunction again in expression (37). This does not change the Boolean function itself, but as a result of this «building up» of the function, let's obtain the minimal CNF with a shorter representation than (38) [15]:

$$F(x_1,x_2,x_3) = (\overline{x_1}+x_2+x_3)(\overline{x_1}+\overline{x_2}+x_3) \times$$
$$\times (\overline{x_1}+\overline{x_2}+x_3)(\overline{x_1}+\overline{x_2}+\overline{x_3}) = (\overline{x_1}+x_3)(\overline{x_1}+\overline{x_2}). \tag{39}$$

The method of figurative transformations of minimizing CNF of Boolean functions allows obtaining the result of minimization (39) without additional «building up» a given function (37):

$$F(x_1,x_2,x_3) = \begin{vmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{vmatrix} = \begin{vmatrix} 0 & & 1 \\ 0 & 0 & 0 \end{vmatrix} = \begin{vmatrix} 0 & & 1 \\ 0 & 0 & \end{vmatrix}.$$

Minimized CNF of the function $F(x_1,x_2,x_3)$:

$$F(x_1,x_2,x_3) = (\overline{x_1}+x_3)(\overline{x_1}+\overline{x_2}). \tag{40}$$

In the first matrix, the operation of gluing of variables (highlighted in red) is applied, in the second matrix, the semi-gluing of variables is carried out. The results of minimization (39) and (40) coincide. In contrast to the algebraic method, figurative transformations have wider possibilities for simplifying the minimization of a CNF of Boolean functions.

Example 6. Minimize the CNF of the Boolean function $F(x_1,x_2,x_3,x_4)$ by the Nelson method, given by the following truth table:

$$F(x_1,x_2,x_3,x_4) = \Pi(4,\ 6,\ 7,\ 9,\ 11).$$

Note: the values in Π are maxterms for rows when the function $F(x_1,x_2,x_3,x_4)$ returns «0» on output (Table 1).

Table 1

Function truth table $F(x_1,x_2,x_3,x_4)$

No.	X_1	X_2	X_3	X_4	F
4	0	1	0	0	0
6	0	1	1	0	0
7	0	1	1	1	0
9	1	0	0	1	0
11	1	0	1	1	0

Based on the truth table (Table 1), according to the Nelson method, let's perform the inversion of variables in the blocks of the Table 1, which thus gives the CNF a given Boolean function. Let's minimize the function $F(x_1,x_2,x_3,x_4)$ obtained by the CNF in two ways.

Option 1. Minimizing the CNF of the Boolean function $F(x_1,x_2,x_3,x_4)$ given by Table 1, by algebraic method:

$$F(x_1,x_2,x_3,x_4) = (x_1+\overline{x_2}+x_3+x_4)(x_1+\overline{x_2}+\overline{x_3}+x_4) \times$$
$$\times (x_1+\overline{x_2}+\overline{x_3}+\overline{x_4})(\overline{x_1}+x_2+x_3+\overline{x_4})(\overline{x_1}+x_2+\overline{x_3}+\overline{x_4}); \tag{41}$$

$$(x_1+\overline{x_2}+x_3+x_4)(x_1+\overline{x_2}+\overline{x_3}+x_4) =$$
$$= x_1 + x_1\overline{x_2} + x_1\overline{x_3} + x_1 x_4 + x_1\overline{x_2} + \overline{x_2} +$$
$$+ \overline{x_2}\,\overline{x_3} + \overline{x_2}x_4 + x_1 x_3 + \overline{x_2}x_3 + x_3 x_4 +$$
$$+ x_1 x_4 + \overline{x_2}x_4 + \overline{x_3}x_4 + x_4 = x_1 + \overline{x_2} + x_4; \tag{42}$$

$$(\overline{x_1}+x_2+x_3+\overline{x_4})(\overline{x_1}+x_2+\overline{x_3}+\overline{x_4}) =$$
$$= \overline{x_1} + \overline{x_1}x_2 + \overline{x_1}\,\overline{x_3} + \overline{x_1}\,\overline{x_4} + \overline{x_1}x_2 + x_2 +$$
$$+ x_2\overline{x_3} + x_2\overline{x_4} + \overline{x_1}x_3 + x_2 x_3 + x_3\overline{x_4} +$$
$$+ \overline{x_1}\,\overline{x_4} + x_2\overline{x_4} + \overline{x_3}\,\overline{x_4} + \overline{x_4} = \overline{x_1} + x_2 + \overline{x_4}. \tag{43}$$

Considering the results (42) and (43), let's rewrite the function (41):

$$F(x_1,x_2,x_3,x_4) =$$
$$= (x_1+\overline{x_2}+x_4)(x_1+\overline{x_2}+\overline{x_3}+\overline{x_4})(\overline{x_1}+x_2+\overline{x_4}). \tag{44}$$

Function (44) allows two options for continuing to minimize it.

In the first option, let's continue to use the Nelson method, for which will open the brackets of the expression (44):

$$(x_1 + \overline{x_2} + x_4)(x_1 + \overline{x_2} + \overline{x_3} + \overline{x_4}) =$$
$$= x_1 + x_1\overline{x_2} + x_1\overline{x_3} + x_1\overline{x_4} + x_1\overline{x_2} + \overline{x_2} +$$
$$+ \overline{x_2}\,\overline{x_3} + \overline{x_2}\,\overline{x_4} + x_1x_4 + \overline{x_2}x_4 + \overline{x_3}x_4 =$$
$$= x_1 + \overline{x_2} + \overline{x_3}x_4;$$

$$(x_1 + \overline{x_2} + \overline{x_3}x_4)(\overline{x_1} + x_2 + \overline{x_4}) =$$
$$= x_1x_2 + x_1\overline{x_4} + \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_4} + \overline{x_1}\,\overline{x_3}x_4 + x_2\overline{x_3}x_4.$$

All possible conversions, reducing the specified function, are exhausted. Thus, the obtained reduced DNF of a given function $F(x_1, x_2, x_3, x_4)$ by the algebraic method:

$$F(x_1, x_2, x_3, x_4) =$$
$$= x_1x_2 + x_1\overline{x_4} + \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_4} + \overline{x_1}\,\overline{x_3}x_4 + x_2\overline{x_3}x_4. \quad (45)$$

In the second option, to minimize the function (44), let's apply the figurative transformations:

$$F(x_1, x_2, x_3, x_4) = \begin{vmatrix} x_1 & \overline{x_1} & & x_4 \\ x_1 & \overline{x_2} & \overline{x_3} & \overline{x_4} \\ \overline{x_1} & x_2 & & \overline{x_4} \end{vmatrix} = \begin{vmatrix} x_1 & \overline{x_1} & & x_4 \\ x_1 & \overline{x_2} & \overline{x_3} & \\ \overline{x_1} & x_2 & & \overline{x_4} \end{vmatrix}.$$

Minimized CNF of the function $F(x_1, x_2, x_3, x_4)$:

$$F(x_1, x_2, x_3, x_4) =$$
$$= (x_1 + \overline{x_2} + x_4)(x_1 + \overline{x_2} + \overline{x_3})(\overline{x_1} + x_2 + \overline{x_4}). \quad (46)$$

In the first matrix, the operation of semi-gluing of variables is performed. Compared to (44), function (46) is simple. Expand the brackets in expression (46):

$$(x_1 + \overline{x_2} + x_4)(x_1 + \overline{x_2} + \overline{x_3}) =$$
$$= x_1 + x_1\overline{x_2} + x_1\overline{x_3} + x_1\overline{x_2} + \overline{x_2} + \overline{x_2}\,\overline{x_3} +$$
$$+ x_1x_4 + \overline{x_2}x_4 + \overline{x_3}x_4 = x_1 + \overline{x_2} + \overline{x_3}x_4;$$

$$(x_1 + \overline{x_2} + \overline{x_3}x_4)(\overline{x_1} + x_2 + \overline{x_4}) =$$
$$= x_1x_2 + x_1\overline{x_4} + \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_4} + \overline{x_1}\,\overline{x_3}x_4 + x_2\overline{x_3}x_4.$$

The resulting reduced DNF of the function $F(x_1, x_2, x_3, x_4)$ in Table 1:

$$F(x_1, x_2, x_3, x_4) =$$
$$= x_1x_2 + x_1\overline{x_4} + \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_4} + \overline{x_1}\,\overline{x_3}x_4 + x_2\overline{x_3}x_4. \quad (47)$$

The results of simplification (45) and (47) coincide.

Option 2. Minimizing the CNF of the Boolean function $F(x_1, x_2, x_3, x_4)$ given by Table 1, by figurative transformations:

$$F(x_1, x_2, x_3, x_4) = \begin{vmatrix} 4 & 1 & 0 & 1 & 1 \\ 6 & 1 & 0 & 0 & 1 \\ 7 & 1 & 0 & 0 & 0 \\ 9 & {\color{red}0} & {\color{red}1} & {\color{red}1} & {\color{red}0} \\ 11 & {\color{red}0} & {\color{red}1} & {\color{red}0} & {\color{red}0} \end{vmatrix} = \begin{vmatrix} 1 & 0 & & 1 \\ 1 & 0 & 0 & \\ {\color{red}0} & {\color{red}1} & & {\color{red}0} \end{vmatrix}.$$

All possible conversions, reducing the specified function, are exhausted. The resulting reduced CNF of the function $F(x_1, x_2, x_3, x_4)$ by figurative transformations:

$$F(x_1, x_2, x_3, x_4) =$$
$$= (x_1 + \overline{x_2} + x_4)(x_1 + \overline{x_2} + \overline{x_3})(\overline{x_1} + x_2 + \overline{x_4}). \quad (48)$$

According to the Nelson method, open the brackets in expression (48) and represent it in the DNF:

$$(x_1 + \overline{x_2} + x_4)(x_1 + \overline{x_2} + \overline{x_3}) = x_1 + x_1\overline{x_2} + x_1\overline{x_2} +$$
$$+ x_1\overline{x_2} + \overline{x_2} + \overline{x_2}\,\overline{x_3} + x_1x_4 + \overline{x_2}x_4 + \overline{x_3}x_4 = x_1 + \overline{x_2} + \overline{x_3}x_4;$$

$$(x_1 + \overline{x_2} + \overline{x_3}x_4)(\overline{x_1} + x_2 + \overline{x_4}) =$$
$$= x_1x_2 + x_1\overline{x_4} + \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_4} + \overline{x_1}\,\overline{x_3}x_4 + x_2\overline{x_3}x_4.$$

Further reductions are no longer possible. The obtained simplified DNF of the function $F(x_1, x_2, x_3, x_4)$ given in Table 1:

$$F(x_1, x_2, x_3, x_4) =$$
$$= x_1x_2 + x_1\overline{x_4} + \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_4} + \overline{x_1}\,\overline{x_3}x_4 + x_2\overline{x_3}x_4. \quad (49)$$

The results of simplification (45) and (49) are the same, but the process of simplifying the CNF of the function in the second version (figurative transformations) is simple.

Note also that the CNF of the function $F(x_1, x_2, x_3, x_4)$ (48), compared with the DNF of the function $F(x_1, x_2, x_3, x_4)$ (49), contains a smaller number of logical operations and inversions. Thus, with the same functionality of expressions (49) and (48), the latter is a simple structure (Fig. 1, *a*).
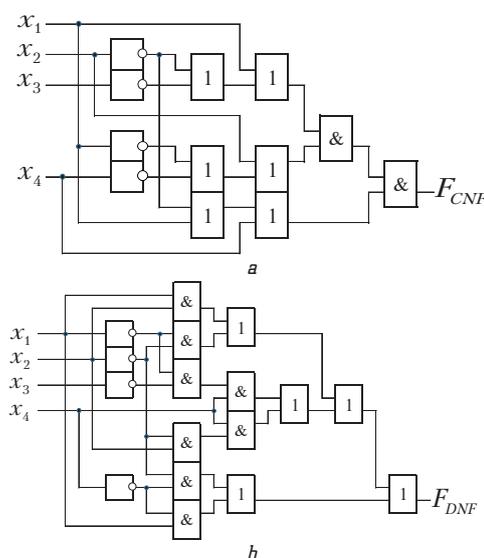
Fig. 1. The implementation of the minimum:
a – conjunctive normal form; *b* – disjunctive normal form of the Boolean function $F(x_1, x_2, x_3, x_4)$ on typical 2-inputs of logic elements

Looking at Fig. 1 it is easy to see that the implementation of the structure of the minimal CNF of the function by logical 2-inputs is simple compared to the implementation of the minimum structure of a DNF of 2 inputs by logical elements, both in complexity and in depth of the circuit.

Table 2 shows the functionality of minimized CNF and DNF of the functions given in Table 1.

Table 2

Function truth table

$$F_{CNF}(x_1, x_2, x_3, x_4) = (x_1 + \overline{x_2} + x_4)(x_1 + \overline{x_2} + \overline{x_3})(\overline{x_1} + x_2 + \overline{x_4}),$$

$$F_{DNF}(x_1, x_2, x_3, x_4) = x_1 x_2 + x_1 \overline{x_4} + \overline{x_1}\ \overline{x_2} + \overline{x_2}\ \overline{x_4} + x_1\ \overline{x_3} x_4 + x_2 \overline{x_3}\ x_4$$

No.	X_1	X_2	X_3	X_4	F_{CNF}	F_{DNF}
0	0	0	0	0	1	1
1	0	0	0	1	1	1
2	0	0	1	0	1	1
3	0	0	1	1	1	1
5	0	1	0	1	1	1
8	1	0	0	0	1	1
10	1	0	1	0	1	1
12	1	1	0	0	1	1
13	1	1	0	1	1	1
14	1	1	1	0	1	1
15	1	1	1	1	1	1
No.	X_1	X_2	X_3	X_4	F_{CNF}	F_{DNF}
4	0	1	0	0	0	0
6	0	1	1	0	0	0
7	0	1	1	1	0	0
9	1	0	0	1	0	0
11	1	0	1	1	0	0

Given the Table 2 it is possible to see that the minimal CNF and DNF of the functions have the same functionality, but the CNF of the minimal function has a simple structure (Fig. 1, *a*).

Example 7. Minimize the CNF of the function given by the PCNF:

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

This function returns zero in such sets: $(0,0,0,0,0)$, $(0,0,0,0,1)$, $(0,0,1,0,0)$, $(0,0,1,1,0)$, $(0,1,1,0,0)$, $(0,1,1,1,0)$, $(1,0,0,0,0)$, $(1,0,0,0,1)$, $(1,1,1,0,0)$, $(1,0,1,1,1)$.

Minimize the CNF of a given function $F(x_1, x_2, x_3, x_4, x_5)$ by figurative transformations:

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

Minimized CNF of the function $F(x_1, x_2, x_3, x_4, x_5)$:

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

The operation of super-gluing of variables in the first matrix is carried out for the blocks that are highlighted in red and blue. In the second matrix, the semi-gluing operation is performed on the variables.

6.1. The use of figurative transformations to establish the characteristics of the minimum logical function. *Establishing the signs of the minimum logical function is reduced to the minimization of a function from sets of truth tables, for which the function returns «1» at the output and for sets of truth tables for which the function returns «0» at the output. With error-free calculations of the minimal function in two cases, the result of minimization will be the same. For this comparison, it is necessary to take into account the fact that a given logical function can have several minimal functions. In this regard, in some cases, the results of minimizing the logical function in DNF and CNF may differ, for example, in one variable, however, both minimized functions will be minimal.*

Example 8. Minimize the logical function $F(x_1, x_2, x_3, x_4, x_5)$ by figurative transformations. Function is given by the following truth table:

$$F = \Sigma(1, 2, 3, 4, 5, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18, 20,$$
$$22, 26, 28, 30, 31).$$

Note: the value in Σ is the minterms for rows when the function $F(x_1, x_2, x_3, x_4, x_5)$ returns «1» on output.

Let's carry out the minimization procedure for a Boolean function $F(x_1, x_2, x_3, x_4, x_5)$ until all the transformations that minimize this function have been exhausted:

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

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

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. The minimization of blocks in the second matrix, highlighted in blue and green, is carried out according to the protocols of minimizing 5-bit Boolean functions [3].

Attempts to further apply the operations of algebraic transformations do not give an improved result. So, the obtained minimum form of the logical function takes the form:

$$F_{DNF} = \overline{x}_1\overline{x}_5 + \overline{x}_1\,\overline{x}_2\,\overline{x}_3 x_4 + x_2 x_3 x_4 +$$
$$+ x_1\overline{x}_2\,\overline{x}_3\,\overline{x}_4 + x_1 x_4\overline{x}_5 + x_3\overline{x}_4\,\overline{x}_5. \tag{50}$$

To minimize the function from the truth table sets, in which the function returns «0» on the output, the Nelson method is used. This implies the minimization of the function in the CNF with the corresponding inversions of the variables in the blocks and with the subsequent transformation of the result of the minimization in the DNF of the function. Truth table sets for which a function $F(x_1, x_2, x_3, x_4, x_5)$ returns «0» on output are determined by the following truth table:

$$F(x_1, x_2, x_3, x_4, x_5) =$$
$$= \Pi(0,\ 6,\ 8,\ 10,\ 19,\ 21,\ 23,\ 24,\ 25,\ 27,\ 29). \tag{51}$$

Let's perform an inversion of variables in blocks of the truth table (51), after which minimize the CNF of the function by figurative transformations:

$$F_{CNF} = \begin{vmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 6 & 1 & 1 & 0 & 0 & 1 \\ 8 & 1 & 0 & 1 & 1 & 1 \\ 10 & 1 & 0 & 1 & 0 & 1 \\ 19 & 0 & 1 & 1 & 0 & 0 \\ 21 & 0 & 1 & 0 & 1 & 0 \\ 23 & 0 & 1 & 0 & 0 & 0 \\ 24 & 0 & 0 & 1 & 1 & 1 \\ 25 & 0 & 0 & 1 & 1 & 0 \\ 27 & 0 & 0 & 1 & 0 & 0 \\ 29 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & & 1 & 0 & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & & 1 & 0 \end{vmatrix} = \begin{vmatrix} 1 & & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & & 1 \\ 0 & & 1 & 0 & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & & 1 & 0 \end{vmatrix} \begin{matrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{matrix}. \tag{52}$$

The blocks of the first matrix 19, 21, 23, 25, 27, 29 (highlighted in blue) are minimized according to the minimization protocol of 5-bit Boolean functions [3]:

$$\begin{vmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{vmatrix} = \begin{vmatrix} 0 & & 1 & 0 & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & & 1 & 0 \end{vmatrix}. \tag{53}$$

According to the Nelson method, the result of minimization (the last matrix) of (52) is written to the minimal function of the CNF:

$$F_{CNF} = y_1 y_2 y_3 y_4 y_5 y_6 y_7,$$

after which open brackets and turn it into a DNF of the minimal Boolean function.

For further algebraic calculations in Example 8, let's make the following change of variables:
– x_n replace on 1_n;
– $\overline{x}_n$ replace on 0_n,
where n – the index that determines the width of the symbol-variable «1» or «0» in the minterm of the logic function.

The advantage of this replacement is that the symbol variable «0» does not require an additional inversion symbol, which simplifies further calculations.

After the above change of variables and by the Nelson method, let's multiply the variables in the CNF blocks of the minimized Boolean function (52):

$$y_{1,2} = (1_1 + 1_3 + 1_4 + 1_5)(1_1 + 1_2 + 0_3 + 0_4 + 1_5) =$$
$$= 1_1 + 1_1 1_2 + 1_1 0_3 + 1_1 0_4 + 1_1 1_5 + 1_1 1_3 + 1_2 1_3 + 1_3 0_4 +$$
$$+ 1_3 1_5 + 1_1 1_4 + 1_2 1_4 + 0_3 1_4 + 1_4 1_5 + 1_1 1_5 + 1_2 1_5 +$$
$$+ 0_3 1_5 + 0_4 1_5 + 1_5 = 1_1 + 1_2 1_3 + 1_3 0_4 + 1_2 1_4 + 0_3 1_4 + 1_5. \tag{54}$$

Multiplying the variables of the first row y_1, the resulting matrix (52), and its second row y_2 is carried out according to the rules of the algebra of logic:

$$1_1 \cdot 1_1 \rightarrow 1_1,\ \ 1_1 \cdot 1_2 \rightarrow 1_1 1_2,\ \ 1_1 \cdot 0_3 \rightarrow 1_1 0_3,$$

and so on, where the lower numeric indices determine the character depth of the character-variable «1» or «0» in the minterm of the logic function.

The main task of minimizing a function in DNF and CNF is the search for terms suitable for one or another algebraic operation, mainly for gluing variables with subsequent absorption. However, with an increase in the number of variables of algebraic expressions, such a search can be quite complicated. When simplifying logical formulas, it is not always obvious which of the laws of the algebra of logic must be applied at a particular stage.

In turn, the figurative transformations of the combinatorial method, due to their intrinsic clarity, allow to solve this problem to a certain extent. In some cases, the apparatus of figurative transformations is the only means to continue the optimal simplification of a logical expression.

Since the obtained logical expression (54) has received that difficulty, when it is not obvious which of the laws of the algebra of logic should be used, let's apply the visual apparatus of the figurative transformations of the combinatorial method:

$$y_{1,2} = \begin{vmatrix} 1 & & & & \\ & 1 & 1 & & \\ & & 1 & 0 & \\ & 1 & & 1 & \\ & & 0 & 1 & \\ & & & & 1 \end{vmatrix} = \begin{vmatrix} 1 & & & & \\ & 1 & 1 & & \\ & & 1 & 0 & \\ & & 0 & 1 & \\ & & & & 1 \end{vmatrix}.$$

Obviously, the application of generalized gluing of variables (one of the options for applying generalized gluing of variables is highlighted in red). As a result, let's obtain a simplified logical expression:

$$y_{1,2} = 1_1 + 1_2 1_3 + 1_3 0_4 + 0_3 1_4 + 1_5. \tag{55}$$

$$y_{3,4} = (0_2 + 1_3 + 1_4 + 1_5)(1_1 + 0_2 + 1_3 + 1_5) = 1_1 0_2 + 0_2 +$$
$$+ 0_2 1_3 + 0_2 1_5 + 1_1 1_3 + 0_2 1_3 + 1_3 + 1_3 1_5 + 1_1 1_4 + 0_2 1_4 +$$
$$+ 1_3 1_4 + 1_4 1_5 + 1_1 1_5 + 0_2 1_5 + 1_3 1_5 + 1_5 = 0_2 + 1_3 + 1_1 1_4 + 1_5.$$

$y_{5,6} = (0_1 + 1_3 + 0_4 + 0_5)(0_1 + 1_2 + 0_3 + 0_5) =$
$= 0_1 + 0_1 1_2 + 0_1 0_3 + 0_1 0_5 + 0_1 1_3 + 1_2 1_3 + 1_3 0_5 +$
$+ 0_1 0_4 + 1_2 0_4 + 0_3 0_4 + 0_4 0_5 + 0_1 0_5 + 1_2 0_5 + 0_3 0_5 + 0_5 =$
$= 0_1 + 1_2 1_3 + 1_2 0_4 + 0_3 0_4 + 0_5. \quad (56)$

Expression (56) applies figurative transformations of the combinatorial method:

$$y_{5,6} = \begin{vmatrix} 0 & & & \\ & 1 & 1 & \\ & 1 & & 0 \\ & & 0 & 0 \\ & & & 0 \end{vmatrix} = \begin{vmatrix} 0 & & & \\ & 1 & 1 & \\ & & 0 & 0 \\ & & & 0 \end{vmatrix}.$$

After applying the generalized gluing of variables (high-lighted in red), let's obtain a simplified logical expression:

$$y_{5,6} = 0_1 + 1_2 1_3 + 0_3 0_4 + 0_5. \quad (57)$$

$y_{1,2,3,4} = y_{1,2} \cdot y_{3,4} = (1_1 + 1_2 1_3 + 1_3 0_4 + 0_3 1_4 + 1_5) \times$
$\times (0_2 + 1_3 + 1_1 1_4 + 1_5) = 1_1 0_2 + 1_1 1_3 + 1_1 1_4 + 1_1 1_5 +$
$+ 1_2 1_3 + 1_1 1_2 1_3 1_4 + 1_2 1_3 1_5 + 0_2 1_3 0_4 + 1_3 0_4 + 1_3 0_4 1_5 +$
$+ 0_2 0_3 1_4 + 1_1 0_3 1_4 + 0_3 1_4 1_5 + 0_2 1_5 + 1_3 1_5 + 1_1 1_4 1_5 + 1_5 =$
$= 1_1 0_2 + 1_1 1_3 + 1_1 1_4 + 1_2 1_3 + 1_3 0_4 + 0_2 0_3 1_4 + 1_5. \quad (58)$

To expression (58) let's apply the figurative transformations of the combinatorial method:

$$y_{1,2,3,4} = \begin{vmatrix} 1 & 0 & & \\ 1 & & 1 & \\ 1 & & & 1 \\ & 1 & 1 & \\ & & 1 & 0 \\ & 0 & 0 & 1 \\ & & & & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & & \\ 1 & & & 1 \\ & 1 & 1 & \\ & & 1 & 0 \\ & 0 & 0 & 1 \\ & & & & 1 \end{vmatrix}.$$

After applying the generalized gluing of variables (high-lighted in red), let's obtain the following simplified logical expression:

$$y_{1,2,3,4} = 1_1 0_2 + 1_1 1_4 + 1_2 1_3 + 1_3 0_4 + 0_2 0_3 1_4 + 1_5. \quad (59)$$

$y_{5,6,7} = y_{5,6} \cdot y_7 = (0_1 + 1_2 1_3 + 0_3 0_4 + 0_5)(0_1 + 0_2 + 1_4 + 0_5) =$
$= 0_1 + 0_1 0_2 + 0_1 1_4 + 0_1 0_5 + 0_1 1_2 1_3 + 1_2 1_3 1_4 +$
$+ 1_2 1_3 0_5 + 0_1 0_3 0_4 + 0_2 0_3 0_4 + 0_3 0_4 0_5 + 0_1 0_5 +$
$+ 0_2 0_5 + 1_4 0_5 + 0_5 = 0_1 + 1_2 1_3 1_4 + 0_2 0_3 0_4 + 0_5.$

$y_{1,2,3,4,5,6,7} = y_{1,2,3,4} \cdot y_{5,6,7} = F_{DNF} = (1_1 0_2 + 1_1 1_4 + 1_2 1_3 +$
$+ 1_3 0_4 + 0_2 0_3 1_4 + 1_5)(0_1 + 1_2 1_3 1_4 + 0_2 0_3 0_4 + 0_5) =$
$+ 1_1 0_2 0_3 0_4 + 1_1 0_2 0_5 + 1_1 1_2 1_3 1_4 + 1_1 1_4 0_5 + 0_1 1_2 1_3 +$
$+ 1_2 1_3 1_4 + 1_2 1_3 0_5 + 0_1 1_3 0_4 + 1_3 0_4 0_5 + 0_1 0_2 0_3 1_4 +$
$+ 0_2 0_3 1_4 0_5 + 0_1 1_5 + 1_2 1_3 1_4 1_5 + 0_2 0_3 0_4 1_5 =$
$+ 1_1 0_2 0_3 0_4 + 1_1 0_2 0_5 + 1_1 1_4 0_5 + 0_1 1_2 1_3 +$
$+ 1_2 1_3 1_4 + 1_2 1_3 0_5 + 0_1 1_3 0_4 + 1_3 0_4 0_5 +$
$+ 0_1 0_2 0_3 1_4 + 0_2 0_3 1_4 0_5 + 0_1 1_5 + 0_2 0_3 0_4 1_5. \quad (60)$

To expression (60), let's apply the figurative transformations of the combinatorial method:

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

In the first matrix, the generalized gluing of variables is applied twice. For the didactic convenience of figurative transformations, the matrix is rewritten to a new line, since the current simplification procedure uses a common block:

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

The same operation is performed again, since the current simplification procedure uses a common block:

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

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

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

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

Attempts to further apply the operations of figurative transformation do not give an improvement in the result. So, after the implementation of the minimization of CNF (52) by the Nelson method using the figurative transformations, the following minimal DNF of the functions are obtained:

$$F_{DNF} = \overline{x_1}x_5 + \overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4 + x_2x_3x_4 +$$
$$+ x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4} + x_1x_4\overline{x_5} + x_3\overline{x_4}\,\overline{x_5}. \tag{61}$$

Functions (50) and (61) coincide, which, according to the sign of the minimal function, indicates that the minimization procedure yields the minimal Boolean function. Since the DNF of the minimal function (50) is simple compared to the CNF of the minimal function (52), the Boolean function (50) should be considered optimal for use in digital technology.

From the above examples, it follows that with an increase in the bit width of the Boolean function, the relative efficiency of the use of figurative transformations to minimize CNF of the functions increases, thanks to the unification of the original procedures and the establishment of features of the minimum logical function.

6.2. The use of figurative transformations to minimize Boolean functions on the complete truth table. The minimization of a DNF or CNF of Boolean functions is performed on the corresponding sets of truth table variables. However, the results of minimizing the Boolean functions in examples 6 and 8 show that in order to obtain the optimal from the point of view of practical implementation of the above method in digital technology, it is advisable to minimize the two forms – DNF and CNF using the full truth table of this function. The complete truth table contains sets of variables in which the function returns «1» or «0» at the output. The minimal function should be chosen according to the results of minimization of two normal forms – DNF and CNF.

Example 9. Minimize the logical function $F(x_1,x_2,x_3,x_4)$ on the complete truth table by figurative transformations in two normal forms – DNF and CNF, which is given in canonical form [16]:

$$F(x_1,x_2,x_3,x_4) = \Sigma(0,1,6,8,11,14,15). \tag{62}$$

The minimal function is chosen according to the results of minimization of two normal forms – DNF and CNF.

Minimization of the DNF of a given function is illustrated by figurative transformations:

$$F_{DNF} = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 6 & 0 & 1 & 1 & 0 \\ 8 & 1 & 0 & 0 & 0 \\ 11 & 1 & 0 & 1 & 1 \\ 14 & 1 & 1 & 1 & 0 \\ 15 & 1 & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 & 1 \\ & 0 & 0 & 0 \\ & 1 & 1 & 0 \\ 1 & & 1 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 \\ & 0 & 0 & 0 \\ & 1 & 1 & 0 \\ 1 & & 1 & 1 \end{vmatrix}.$$

Minimized DNF of the function $F(x_1,x_2,x_3,x_4)$:

$$F_{DNF} = \overline{x_1}\,\overline{x_2}\,\overline{x_3} + \overline{x_2}\,\overline{x_3}\,\overline{x_4} + x_2x_3\overline{x_4} + x_1x_3x_4. \tag{63}$$

The results of the minimizing the DNF of the function $F(x_1,x_2,x_3,x_4)$ using parallel splitting of conjuncterms [16] and the method of figurative transformations are presented in Table 3

Table 3

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

By the method of parallel splitting of conjuncterms	By the method of figurative transformations
$\{(000\sim),(\sim 000),(\sim 110),(1\sim 11)\}$	$\begin{vmatrix} 0 & 0 & 0 \\ & 0 & 0 & 0 \\ & 1 & 1 & 0 \\ 1 & & 1 & 1 \end{vmatrix}$

From Table 3 it can be seen that the results of minimizing the two compared methods are the same. The minimization exponent coincides $k_\theta/k_l = 4/12$, where k_θ – the number of simple implicants, k_l – the number of input variables. However, the computational complexity of Boolean function minimization by figurative transformations is less.

Minimizing the CNF of a given function:

$$F_{CNF} = \begin{vmatrix} 2 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 1 & 1 \\ 4 & 0 & 1 & 0 & 0 \\ 5 & 0 & 1 & 0 & 1 \\ 7 & 0 & 1 & 1 & 1 \\ 9 & 1 & 0 & 0 & 1 \\ 10 & 1 & 0 & 1 & 0 \\ 12 & 1 & 1 & 0 & 0 \\ 13 & 1 & 1 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{vmatrix} = \begin{vmatrix} & 1 & 0 & 1 \\ 1 & & 0 & 0 \\ & 0 & 1 \\ 0 & 1 & 1 & 0 \end{vmatrix} = \begin{vmatrix} & 1 & 0 & 1 \\ 1 & & 0 & 0 \\ & 0 & 1 \\ 0 & & 1 & 0 \end{vmatrix}.$$

Minimized CNF of the function $F(x_1,x_2,x_3,x_4)$:

$$F_{CNF} = (x_1 + \overline{x_3} + \overline{x_4})(\overline{x_1} + x_3 + \overline{x_4}) \times$$
$$\times (x_2 + \overline{x_3} + x_4)(\overline{x_2} + x_3). \tag{64}$$

The minimal CNF of the function $F(x_1,x_2,x_3,x_4)$ (64) contains a smaller number of literals, compared with the minimal DNF of the function $F(x_1,x_2,x_3,x_4)$ (63). Consequently, with the same functionality of expressions (63) and (64) (Table 4), the latter corresponds to a simpler structure (Fig. 2, *a*).

From Fig. 2 it is possible to see that the implementation of the combinational methods of the minimal CNF of the Boolean function (Fig. 2, *a*) is simple, because it contains a 2-input logical element OR, which are absent in the method that implements the minimal DNF of the Boolean function (Fig. 2, *b*).

Table 4 shows the functionality of minimized CNF and DNF of the functions given the canonical form (62).

Given the Table 4, it is possible to see that the minimal CNF and DNF of the functions have the same functionality, but the CNF of the minimal function has one less literal.

According to the results of minimization of two normal forms – DNF and CNF of a given function, let's choose the minimal function in CNF (64).
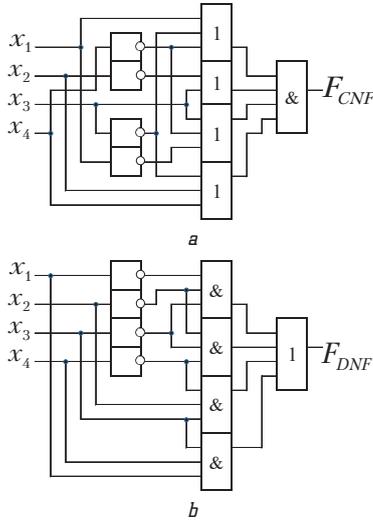
Fig. 2. The implementation of the minimum:
a – conjunctive normal form; *b* – disjunctive normal form of the Boolean
function $F(x_1, x_2, x_3, x_4)$ by combinational method

Table 4

Function truth table

$F_{CNF}(x_1, x_2, x_3, x_4) = (x_1 + \overline{x_3} + \overline{x_4})(\overline{x_1} + x_3 + \overline{x_4})(x_2 + \overline{x_3} + x_4)(\overline{x_2} + x_3)$,

$F_{CNF}(x_1, x_2, x_3, x_4) = \overline{x_1}\,\overline{x_2}\,\overline{x_3} + \overline{x_2}\,\overline{x_3}\,\overline{x_4} + x_2 x_3 x_4 + x_1 x_3 x_4$

No.	X_1	X_2	X_3	X_4	F_{CNF}	F_{DNF}
0	0	0	0	0	1	1
1	0	0	0	1	1	1
6	0	1	1	0	1	1
8	1	0	0	0	1	1
11	1	0	1	1	1	1
14	1	1	1	0	1	1
15	1	1	1	1	1	1
No.	X_1	X_2	X_3	X_4	F_{CNF}	F_{DNF}
2	0	0	1	0	0	0
3	0	0	1	1	0	0
4	0	1	0	0	0	0
5	0	1	0	1	0	0
7	0	1	1	1	0	0
9	1	0	0	1	0	0
10	1	0	1	0	0	0
12	1	1	0	0	0	0
13	1	1	0	1	0	0

Example 10. Minimize the logical function $F(x_1, x_2, x_3, x_4)$ by figurative transformations on the complete truth table in two normal forms – DNF and CNF, which is given by the following truth table:

$$F(x_1, x_2, x_3, x_4) = \Sigma(1, 3, 6, 7, 8, 10, 14, 15).$$

The minimal function is chosen according to the results of minimization of two normal forms – DNF and CNF.

Below is the minimization of the DNF of a given function by the method of figurative transformations:

$$F_{DNF} = \begin{vmatrix} 1 & 0\ 0\ 0\ 1 \\ 3 & 0\ 0\ 1\ 1 \\ 6 & 0\ 1\ 1\ 0 \\ 7 & 0\ 1\ 1\ 1 \\ 8 & 1\ 0\ 0\ 0 \\ 10 & 1\ 0\ 1\ 0 \\ 14 & 1\ 1\ 1\ 0 \\ 15 & 1\ 1\ 1\ 1 \end{vmatrix} = \begin{vmatrix} 0\ 0 & & 1 \\ & 1\ 1 & \\ 1\ 0 & & 0 \end{vmatrix}. \tag{65}$$

Minimized DNF of the function $F(x_1, x_2, x_3, x_4)$:

$$F_{DNF} = \overline{x_1}\,\overline{x_2} x_4 + x_2 x_3 + x_1 \overline{x_2}\,\overline{x_4}. \tag{66}$$

Minimized CNF of the function:

$$F_{CNF} = \begin{vmatrix} 0 & 0\ 0\ 0\ 0 \\ 2 & 0\ 0\ 1\ 0 \\ 4 & 0\ 1\ 0\ 0 \\ 5 & 0\ 1\ 0\ 1 \\ 9 & 1\ 0\ 0\ 1 \\ 11 & 1\ 0\ 1\ 1 \\ 12 & 1\ 1\ 0\ 0 \\ 13 & 1\ 1\ 0\ 1 \end{vmatrix} = \begin{vmatrix} 1\ 1\ 1\ 1 \\ 1\ 1\ 0\ 1 \\ 1\ 0\ 1\ 1 \\ 1\ 0\ 1\ 0 \\ 0\ 1\ 1\ 0 \\ 0\ 1\ 0\ 0 \\ 0\ 0\ 1\ 1 \\ 0\ 0\ 1\ 0 \end{vmatrix} = \begin{vmatrix} 1\ 1 & & 1 \\ & 0\ 1 & \\ 0\ 1 & & 0 \end{vmatrix}.$$

Minimized CNF of the function $F(x_1, x_2, x_3, x_4)$:

$$F_{CNF} = (x_1 + x_2 + x_4)(\overline{x_2} + x_3)(\overline{x_1} + x_2 + \overline{x_4}). \tag{67}$$

Let's transform the minimal CNF (67) into DNF:

$$(x_1 + x_2 + x_4)(\overline{x_2} + x_3) = x_1 \overline{x_2} + x_1 x_3 + x_2 x_3 + \overline{x_2} x_4 + x_3 x_4.$$

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

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

$$\begin{vmatrix} 1\ 0\ \ \ 0 \\ 0\ 1\ 1 \\ \ \ 1\ 1 \\ \ \ 1\ 1\ 0 \\ 0\ 0\ \ \ 1 \end{vmatrix} = \begin{vmatrix} 1\ 0\ \ \ 0 \\ \ \ 1\ 1 \\ 0\ 0\ \ \ 1 \end{vmatrix}. \tag{68}$$

The results of calculations (65) and (68) coincide, which corresponds to the procedure for obtaining the minimization of a function on the complete truth table. It can be seen that in the minimal CNF of the function (67) compared with the minimal DNF of the function (66), for the input variable is one less inversion. Therefore, the implementation of the CNF function of the combinational circuit will give one less connection (Fig. 3).
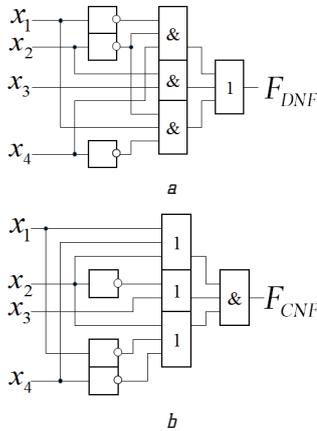
Fig. 3. The implementation of the minimum:
a – disjunctive normal form; *b* – conjunctive normal form of the Boolean
function $F(x_1, x_2, x_3, x_4)$ by combinatorial methods

From Fig. 3 it follows that the structure of the combinational scheme, which implements the minimal CNF of the function (Fig. 3, *b*), contains less wired connections, as compared with the implementation of the minimal DNF (Fig. 3, *a*). This allows to technologically simplify the manufacture of the scheme. Therefore, ceteris paribus, it is advisable to choose CNF as the minimal function in terms of the technological implementation of the scheme (67).

Example 11. Minimize the logical function $F(x_1, x_2, x_3, x_4)$ by figurative transformations on the complete table in two normal forms –DNF and CNF, which is given by the following truth table [17]:

$$F(x_1, x_2, x_3, x_4) = (0,\ 0,\ 1,\ 1,\ 1,\ 0,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 0,\ 0,\ 0,\ 1).$$

The minimal function is chosen according to the results of minimization of two normal forms – DNF and CNF.

To minimize the DNF of a given function, let's compile a truth table of a 4-bit Boolean function from blocks at which the function returns the value «1», that is, for sets: 2, 3, 4, 6, 7, 8, 9, 10, 11, 15. And let's perform minimization:

$$F_{DNF} = \begin{vmatrix} 2 & 0\ 0\ 1\ 0 \\ 3 & 0\ 0\ 1\ 1 \\ 4 & 0\ 1\ 0\ 0 \\ 6 & 0\ 1\ 1\ 0 \\ 7 & 0\ 1\ 1\ 1 \\ 8 & 1\ 0\ 0\ 0 \\ 9 & 1\ 0\ 0\ 1 \\ 10 & 1\ 0\ 1\ 0 \\ 11 & 1\ 0\ 1\ 1 \\ 15 & 1\ 1\ 1\ 1 \end{vmatrix} = \begin{vmatrix} 0\ 0\ 1 \\ 0\ 1 & 0 \\ 1\ 0 \\ & 1\ 1\ 1 \end{vmatrix} = \begin{vmatrix} & 0\ 1 \\ 0\ 1 & 0 \\ 1\ 0 \\ & 1\ 1\ 1 \end{vmatrix} = \begin{vmatrix} & 0\ 1 \\ 0\ 1 & 0 \\ 1\ 0 \\ & 1\ 1 \end{vmatrix}.$$

For blocks 8–11 (highlighted in red) of the first matrix a super-gluing protocol for variables used, since there is a combinatorial system 2 (2, 4)-design [2]. Simple gluing of variables is highlighted in colors. Incomplete gluing of variables is carried out in the last two matrices.

As a result, let's obtain the minimal DNF of the function:

$$F_{DNF} = x_1 \overline{x_2} + \overline{x_1} x_2 \overline{x_4} + \overline{x_2} x_3 + x_3 x_4. \tag{69}$$

Table 5 presents the results of minimization of a function $F(x_1, x_2, x_3, x_4)$ using parallel splitting of conjuncterms [17] and the method of figurative transformations.

Table 5
The result of minimizing the function $F(x_1, x_2, x_3, x_4)$

By the method of parallel splitting of conjuncterms	By the method of figurative transformations
$F_{DNF} = x_1 \overline{x_2} + \overline{x_1} x_2 \overline{x_4} + \overline{x_1} x_3 + x_3 x_4$	$F_{DNF} = x_1 \overline{x_2} + \overline{x_1} x_2 \overline{x_4} + \overline{x_2} x_3 + x_3 x_4$

Given the Table 5 it is easy to see that both functions have the same parameters and are verified, although differ in the composition of variables in the third implicant. Example 11 demonstrates the lower computational complexity of minimizing the DNF of a Boolean function by the combinatorial method.

Minimizing the CNF of a given function:

$$F_{CNF} = \begin{vmatrix} 0 & 0\ 0\ 0\ 0 \\ 1 & 0\ 0\ 0\ 1 \\ 5 & 0\ 1\ 0\ 1 \\ 12 & 1\ 1\ 0\ 0 \\ 13 & 1\ 1\ 0\ 1 \\ 14 & 1\ 1\ 1\ 0 \end{vmatrix} = \begin{vmatrix} 1\ 1\ 1\ 1 \\ 1\ 1\ 1\ 0 \\ 1\ 0\ 1\ 0 \\ 0\ 0\ 1\ 1 \\ 0\ 0\ 1\ 0 \\ 0\ 0\ 0\ 1 \end{vmatrix} = \begin{vmatrix} 1\ 1\ 1 \\ & 0\ 1\ 0 \\ 0\ 0 & 1 \end{vmatrix}.$$

Minimized CNF of the function $F(x_1, x_2, x_3, x_4)$:

$$F_{CNF} = (x_1 + x_2 + x_3)(\overline{x_2} + x_3 + \overline{x_4})(\overline{x_1} + \overline{x_2} + x_4). \tag{70}$$

The minimum CNF of the function $F(x_1, x_2, x_3, x_4)$ (70) compared to the minimum DNF of the function $F(x_1, x_2, x_3, x_4)$ (69) has the same number of literals, but fewer terms, which gives a technological simplification of the scheme development. In this connection, all other things being equal, it is advisable to choose CNF as the minimum function (70).

7. SWOT analysis of research results

Strengths. The strength of the combinatorial method of Boolean functions minimization can be attributed to the reduction in the complexity of the algorithm for minimizing the CNF of Boolean functions. This distinguishes the combinatorial method in comparison with analogues by the following factors:
– increase in the productivity of mental labor (intellectual component) while minimizing the CNF of the Boolean functions, contributes to the improvement of the algorithm for minimizing the CNF of logical functions, expanding the control functions of the combinatorial method and a deeper understanding of logical transformations;
– decrease in the amount of computation in the case of using the signs of the minimum function and a decrease in the computation volume in the case of Boolean functions minimization on the full truth table;
– lower cost of development and implementation by reducing the need for the use of hardware-software automation tools.

Weaknesses. The weak side of the combinatorial method for manual minimization of CNF of Boolean functions is associated with the small practice of applying the method of minimizing CNF of Boolean functions. Negative internal factors inherent in the process of minimization of CNF of Boolean functions by the combinatorial method consist in increasing the time for obtaining the minimum function with an insufficient library of protocols for minimizing CNF of Boolean functions.

Opportunities. The opportunities of further research of the combinatorial method may be the development of a protocol for computing minimal functions for symmetric Boolean functions.

Additional possibilities for the practical implementation of the combinatorial method of minimizing the CNF of Boolean functions consist in establishing new criteria for combinatorial optimization of Boolean functions, determined by the sign of the minimal function and Boolean functions minimization on the full truth table.

Threats. The process of CNF minimization of the Boolean functions by the combinatorial method is independent of the minimization processes by other methods, therefore there is no threat of negative impact on the object of study of external factors.

An analogue of the combinatorial method of minimizing CNF of Boolean functions is the algebraic method [18]. The algebraic method for minimizing CNF of Boolean functions is best in that for it the already predetermined laws of simplification, the discovered properties and algorithms for Boolean functions minimization are created. However, the algebraic method is a verbal procedure for operational transformations, which has a lesser effect on the quality of minimization compared with the figurative transformations of the combinatorial method.

8. Conclusions

1. It is established that the minimization of a CNF of the Boolean functions by the combinatorial method is based on a flowchart with a repetition. This is the truth table of this function. This allows to focus on the minimization principle within the protocol for calculating a logical function (within the truth table of a function) and, thus, dispense with auxiliary objects, such as the Karnaugh map, Veitch diagrams, acyclic graph, cubic representation, etc.

2. It is revealed that the tabular organization of the mathematical apparatus of the block diagram with repetition allows to obtain more information about the orthogonality, adjacency, uniqueness of the blocks of the combinatorial system, and, consequently, the blocks of the truth table of this function. Equivalent figurative transformations, by their properties, have a large information capacity, capable of replacing verbal procedures of algebraic transformations, in particular, using the library of protocols for minimizing CNF of Boolean functions.

3. It is revealed that figurative transformations simplify the procedure for establishing the signs of a minimum logical function (examples 8–10), which guarantees an optimal reduction in the number of variables of a logic function without losing its functionality.

4. It is revealed that to achieve the best result of Boolean functions minimization can be obtained in the DNF and in the CNF of the minimal function (examples 6, 8–11). It follows that the minimization of a given function should be carried out in two normal forms – DNF and CNF, using the full truth table, and the minimal function should be chosen according to the results of minimization of two normal forms – DNF and CNF.

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. Riznyk V., Solomko M. Research of 5-bit boolean functions minimization protocols by combinatorial method // Technology Audit and Production Reserves. 2018. Vol. 4, Issue 2 (42). P. 41–52. doi: http://doi.org/10.15587/2312-8372.2018.140351
4. Cepek O., Kucera P., Savicky P. Boolean functions with a simple certificate for CNF complexity // Discrete Applied Mathematics. 2012. Vol. 160, Issue 4-5. P. 365–382. doi: http://doi.org/10.1016/j.dam.2011.05.013
5. Hemaspaandra E., Schnoor H. Minimization for Generalized Boolean Formulas // Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence. 2012. P. 566–571.
6. Boros E., Cepek O., Kucera P. A decomposition method for CNF minimality proofs // Theoretical Computer Science. 2013. Vol. 510. P. 111–126. doi: http://doi.org/10.1016/j.tcs.2013.09.016
7. Gursk'y, S. Minimization of Matched Formulas // WDS'11 Proceedings of Contributed Papers. Part 1. 2011. P. 101–105.
8. Bernasconi A., Ciriani V., Luccio F., Pagli L. 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. doi: http://doi.org/10.1109/tcad.2003.814950
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. Valli M., Periyasamy Dr. R., Amudhavel J. A state of appraoches on minimization of boolean functions // Journal of Advanced Research in Dynamical and Control Systems. 2017. Issue 12. P. 1322–1341. URL: http://www.jardcs.org/abstract.php?archiveid=1323#
11. Boyar J., Peralta R. A New Combinational Logic Minimization Technique with Applications to Cryptology. Lecture Notes in Computer Science. Berlin: Springer, 2010. P. 178–189. doi: http://doi.org/10.1007/978-3-642-13193-6_16
12. Fiser P., Toman D. A Fast SOP Minimizer for Logic Funcions Described by Many Product Terms // 2009 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools. Patras, 2009. doi: http://doi.org/10.1109/dsd.2009.157
13. Pynko A. P. Minimal sequent calculi for monotonic chain finitely-valued logics // Bulletin of the Section of Logic. 2014. Vol. 43, Issue 1-2. P. 99–112.
14. Pyn'ko A. P. Minimizaciya KNF chastichno-monotonnykh bulevykh funkciy // Dopovidi Nacional'noi akademii nauk Ukraini. 2017. Issue 3. P. 18–21.
15. Bulevy funkcii. URL: http://any-book.org/download/88296.html
16. Rytsar B. Ye. New minimization method of logical functions in polynomial set-theoretical format. 1. Generalized rules of conjuncterms simplification // Upravlyayushhie sistemy i mashiny. 2015. Issue 2. P. 39–57. URL: http://dspace.nbuv.gov.ua/handle/123456789/87194
17. Rytsar B. Ye. Minimatsiia systemy lohichnykh funktsii metodom paralelnoho rozcheplennia koniunktermiv // Visnyk Natsionalnoho universytetu «Lvivska politekhnika». Radioelektronika ta telekomunikatsii. 2013. Issue 766. P. 18–27. URL: http://nbuv.gov.ua/UJRN/VNULPPT_2013_766_6
18. Martyniuk O. M. Osnovy dyskretnoi matematyky. Konspekt lektsii. Odesa: 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, ORCID: http://orcid.org/0000-0002-3880-4595

Solomko Mykhailo, PhD, Associate Professor, Department of Computer Engineering, National University of Water and Environmental Engineering, Rivne, Ukraine, e-mail: doctrinas@ukr.net, ORCID: http://orcid.org/0000-0003-0168-5657