e-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2019. № 4
p-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2019. № 4

UDC 004.056

THE INVERSION METHOD OF FOUR-BIT BOOLEAN SAC CRYPTOTRANSFORMS

Fedotova-Piven I. M. – PhD, Assistant Professor, Assistant Professor of the Department of Information Security and Computer Engineering, Cherkassy State Technological University, Cherkassy, Ukraine.

Rudnytskyi V. M. – Dr. Sc., Professor, Head of the Department of Information Security and Computer Engineering, Cherkassy State Technological University, Cherkassy, Ukraine.

Piven O. B. – PhD, Assistant Professor, Professor of the Department of Information Security and Computer Engineering, Cherkassy State Technological University, Cherkassy, Ukraine.

Myroniuk T. V. – PhD, Assistant Professor of the Department of Information Security and Computer Engineering, Cherkassy State Technological University, Cherkassy, Ukraine.

ABSTRACT

Context. Nonlinear systems of Boolean functions play a prominent role in the protection of cryptosystems. The creation and use of new four-bit cryptographic transformations with nonlinear Boolean functions that have the property of strict avalanche criterion is an actual task for increasing the reliability of information protection systems.

Objective. The goal of the work is creating a method for obtaining inverse four-bit cryptographic transformations with the strict avalanche criterion property, which contain balanced Boolean functions only with the operations of inversion and addition modulo two.

Method. A method is proposed for obtaining inverse four-bit cryptographic transformations with the strict avalanche criterion property, each of which contains balanced Boolean functions only with the operations of inversion and addition modulo two. The method simplifies the process of finding inverse cryptographic transformations by creating a class of thirty balanced basic Boolean functions with the required predefined limitations and properties and for finding, within this class, the basic Boolean functions that make up the inverse cryptographic transformation.

Results. The effectiveness of the method is shown for obtaining two inverse four-bit cryptographic transformations with the property of a strict avalanche criterion from two direct four-bit cryptographic transformations with the property of a strict avalanche criterion.

Conclusions. For the first time, there was proposed a method for obtaining inverse four-bit cryptographic transformations with the strict avalanche criterion property for balanced Boolean functions containing two logical operations (inversion and addition modulo two) to ensure reliable information protection. This method is a method of selecting the already existing basic Boolean functions from a predetermined set of balanced basic Boolean functions for direct and inverse cryptographic transformations, whereas the existing methods of searching for inverse cryptographic transformation are methods for calculating each element of the Boolean functions for the inverse cryptographic transformation. The method can be extended to a larger even number of arguments of the balanced Boolean functions of cryptographic transformations to increase the cryptographic resilience.

KEYWORDS: Boolean functions, inverse cryptographic transformation, balancedness, strict avalanche criterion, inversion, addition modulo 2.

ABBREVIATIONS

BBF is a basic Boolean function;
BF is a Boolean function;
CA is a cryptographic algorithm;
CT is a cryptographic transformation;
DCT is a direct cryptographic transformation;
FPBE is a Forward Problem of Boolean equations;
HW is a Hamming weight;
ICT is a inverse cryptographic transformation;
IPBE is a Inverse Problem of Boolean equations;
SAC is a strict avalanche criterion.

NOMENCLATURE

$\neg$ is a sign of the Boolean operation inversion (complementation);

$\oplus$ is a sign of the Boolean operation addition modulo 2 (XOR – exclusive OR);

$B_{n,n}$ is a set of all Boolean functions with n inputs and n outputs;

$f_i(x_1,...,x_n)$ is the i-th Boolean function with n inputs $x_1,...,x_n$ and 1 output;

F is a Boolean function with n inputs and n outputs;

F_{pi}^r is the values of the i-th basic Boolean function of the inverse cryptographic transformation F_1^r;

$f_j(x_1,...,x_4)$ is the j-th basic Boolean function from the set of basic Boolean functions (Table 1);

$wt(f)$ is a Hamming weight of the Boolean function $f(x_1,...,x_n)$.

INTRODUCTION

Nowadays, the number of users of the Internet and digital mobile networks (such as GSM) is more than 4 billions [1], the amount of data transmission is huge. Therefore, data security plays a crucially important role in this data transmission. One of the main ways to ensure the reliability and safety of information is effective methods of encryption/decryption of data [2] with high cryptographic resilience. Today, computationally resilient cryptosystems generally protect information in a satisfactory way, but quantum computers with computing power far beyond the computing power of any classical com-

DOI 10.15588/1607-3274-2019-4-19

puter [3–5] can solve a lot of cryptanalysis tasks that can not be solved by traditional computing systems. The issue of crypto security of information security systems has become extremely acute in connection with the advent of quantum computers.

BFs play a prominent role in the security of cryptosystems [6]. Their most important cryptographic applications include the analysis and design of S-boxes in block ciphers and the construction of filter/combining functions in stream ciphers [7]. Constructing optimal S-boxes has been a prominent topic of interest for security experts [8]. Also, each reversible BF can be implemented as a reversible circuit [9], whereas reversible circuits are indispensable in error correction [10, 11].

Cryptoresistance of a broad class of CAs is determined by their correspondence to some special criteria of bit transform BFs being implemented in these algorithms [12]. One of such criteria is a SAC [12], that is whenever a single input bit is complemented, each of the output bits changes with a probability of one half [13]. This is essential to diminish any correlation between input and output combinations and fails to leak information [14]. This also means that there are no functions with fewer bits, that is a good approximation to the given function and the use of which would significantly reduce the amount of work required to decode the message [15]. That is why the design problem of the Boolean SAC-functions is actually and practically important [16].

The object of study is the process of constructing DCT and ICT of BFs defined by systems for the number of arguments 4 and more.

The subject of study is the methods of constructing ICTs of BFs by given DCTs of BFs that have the property of a SAC and contain only the operations of inversion and addition modulo two.

The purpose of the work is creating a method for obtaining inverse four-bit CTs with the SAC property, which contain balanced BFs only with the operations of inversion and addition modulo two for increasing the reliability of information protection systems. The method must have an applicability property on a larger even number of bits.

1 PROBLEM STATEMENT

It is important to study four-bit, eight-bit BFs in public key cryptography [17, 18]. The formalized procedure for construction of four-bit Boolean SAC-functions with the operations of inversion is proposed in [16]. But CTs with four-bit Boolean SAC-functions with the operations of inversion and addition modulo two are insufficiently investigated and remains relevant. Mathematical statement: we have Boolean multiple-output function $F^d \in B_{4,4}$ with 4 inputs and 4 outputs with the SAC property and bijection property (there are no two or more different sets of input values of $F^d \in B_{4,4}$ that corresponds to the same set of the output values). The function $F^d \in B_{4,4}$ contain balanced BFs only with the operations of inversion and addition modulo two. The function $F^d \in B_{4,4}$ forms a direct four-bit CT. It is required to receive a Boolean multiple-output function $F^r \in B_{4,4}$ with 4 inputs and 4 outputs (an ICT)). All the sets of input values of the function $F^d \in B_{4,4}$ and all the sets of the output values of the function $F^r \in B_{4,4}$ must be the same, and all the sets of the output values of the function $F^d \in B_{4,4}$ and all the sets of the input values of the function $F^r \in B_{4,4}$ also must be the same.

2 REVIEW OF THE LITERATURE

The problem of finding the roots of a system of nonlinear BFs is analytically intractable and therefore provides the basis for many CAs [12]. The FPBE consists of finding all solutions of a system of Boolean equations, whereas the IPBE aims at reconstructing the mathematical formulae of the system of Boolean equations for given the set of solutions. The FPBE has been extensively treated in the literature [19–22] while the inverse problem seems to have received no or little attention [23].

In papers [23–26], various methods for obtaining inverse Boolean functions with *n* inputs and 1 output are described for given direct Boolean functions with *n* inputs and 1 output, but inverse Boolean functions with *n* inputs and *n* outputs for given direct Boolean functions with *n* inputs and *n* outputs in these papers are not considered.

The paper [30] describes invertible Boolean functions of three variables. The papers [27, 28] describe the properties of the Boolean function with *n* inputs and *n* outputs, but the concrete method or algorithm for obtaining the inverse Boolean functions with *n* inputs and *n* outputs for given direct Boolean functions with *n* inputs and *n* outputs is not given. Other publications containing a concrete algorithm for obtaining an ICT using a given DCT containing four or more Boolean functions with four or more Boolean variables and two or more different Boolean operations are unknown to the authors of this article.

The existing methods [23–26] of searching for an inverse Boolean functions are methods for calculating each element of the BFs of the inverse Boolean functions for given direct Boolean functions and this situation needs the development of more effective methods for obtaining ICT for given DCT.

3 MATERIALS AND METHODS

We assume that everywhere in the article $n \geq 1, n \in N$ and $i \in \{1,...,n\}$.

Let $B \overset{def}{=} \{0;1\}$ denote the Boolean values and $B_{n,n}$ [27] denote the set of all BFs with *n* inputs and *n* outputs, where

$$B_{n,n} \overset{def}{=} \{F \mid F : B^n \to B^n\}. \quad (1)$$

DOI 10.15588/1607-3274-2019-4-19

We write $B_n \overset{def}{=} B_{n,1}$ for each n and assume that each $f_i(x_1,...,x_n) \in B_n$ for each i is represented by a propositional formula over the variables $\{x_1,...,x_n\}$ [27]. Conversely, any n-tuple t of BFs over variables $\{x_1,...,x_n\}$ corresponds to a unique BF $F_t \in B_{n,n}$ [27]. We assume that each function $F \in B_{n,n}$ is represented as a tuple $F = (f_1(x_1,...,x_n),..., f_n(x_1,...,x_n))$, where $f_i(x_1,...,x_n) \in B_n$ for each i and hence for each $F(\vec{x}) = (f_1(\vec{x}),..., f_n(\vec{x}))$ for each $\vec{x} \in B^n$ [27].

As known [28], function $F: B^n \to B^n$ is called reversible iff F is bijective, i.e., if each input pattern uniquely maps to an output pattern, and vice versa. Otherwise, it is called irreversible.

Let a DCT and ICT are Boolean multiple-output functions $F^d \in B_{n,n}$ and $F^r \in B_{n,n}$ respectively with n inputs and n outputs. Then not every DCT that satisfies the SAC, has a pertinent ICT. For example, for a DCT $F^d \in B_{n,n}$ that given by the formula (2) and satisfies the SAC, there is no ICT because two different sets (for example, a direct set $x_1 = x_2 = x_3 = x_4 = 0$ and an inverse set $x_1 = x_2 = x_3 = x_4 = 1$) of the input values

$$F^d = \begin{bmatrix} x_1 \oplus x_2 \\ x_1 \oplus x_3 \\ \neg(x_1 \oplus x_2) \\ \neg(x_2 \oplus x_4) \end{bmatrix}. \quad (2)$$

of the DCT corresponds to the same set of the output values – the results of the operation (2):

$$F^d = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

and therefore, the property of one-to-one correspondence (bijection) of the ICT $F^r \in B_{n,n}$ is lost.

As known [29], the HW $wt(f)$ of the BF $f(x_1,...,x_n) \in B^n$ is the number of the nonzero terms in the truth table of the BF:

$$wt(f) = \sum_i^n x_i.$$

As known [29], the BF $f(x_1,...,x_n) \in B^n$ is balanced if its HW $wt(f) = 2^{n-1}$, i.e. the output column of this BF in the truth table contains equal number of 0's and 1's.

To date, the problem of the total number of balanced Boolean SAC-functions determination for n variables remains open [16]. The search area of the roots of systems of Boolean equations may be decreased significantly by application of different expedients based on taking into account the special features of BFs constructing the system of nonlinear Boolean equations [12].

We will consider four-bit CTs that satisfy a SAC and are constructed using only Boolean operations of the inversion and addition modulo 2. To synthesize both DCT and ICT, we create a set of BBFs $f_j(x_1,...,x_4) \in B^4$, $j \in \{1,...,30\}$ with such restrictions:

1) all the BBFs from Table 1 are balanced, because the HW of each of them is $2^3 = 8$, that is, a half of the number 16 - the length of the vector of values of each BBF;

2) each BBF from Table 1 contains from one to four variables $x_1,..., x_4$, and the same variable is included only once in each BBF of the CT;

3) all the BBFs from Table 1 must have non-coinciding sets of values (see Table 2).

We will assume that the BBFs of an ICT will be selected from the same set of BBFs from Table 1. Functions $f_1(x_1,...,x_4),..., f_{30}(x_1,...,x_4)$ are a superposition of variables and operations of inversion and addition modulo 2. Functions $f_{31}(x_1,...,x_4) = 0$ and $f_{32}(x_1,...,x_4) = 1$ for any values of $x_1,..., x_4$ are not listed in Table 1, because they do not contain operations symbols over variables, that is, there is no explicitly indicated mathematical form of the function.

To construct ICTs that satisfy the SAC, we apply the following method that defines the BBFs of the ICT over

Table 1 – Basic Boolean functions $f_1(x_1,...,x_4),..., f_{30}(x_1,...,x_4)$

1) $f_1 = x_1$;	11) $f_{11} = x_1 \oplus x_4$;	21) $f_{21} = x_1 \oplus x_2 \oplus x_3$;
2) $f_2 = x_2$;	12) $f_{12} = x_2 \oplus x_3$;	22) $f_{22} = x_1 \oplus x_2 \oplus x_4$;
3) $f_3 = x_3$;	13) $f_{13} = x_2 \oplus x_4$;	23) $f_{23} = x_1 \oplus x_3 \oplus x_4$;
4) $f_4 = x_4$;	14) $f_{14} = x_3 \oplus x_4$;	24) $f_{24} = x_2 \oplus x_3 \oplus x_4$;
5) $f_5 = \neg(x_1)$;	15) $f_{15} = \neg(x_1 \oplus x_2)$;	25) $f_{25} = x_1 \oplus x_2 \oplus x_3 \oplus x_4$;
6) $f_6 = \neg(x_2)$;	16) $f_{16} = \neg(x_1 \oplus x_3)$;	26) $f_{26} = \neg(x_1 \oplus x_2 \oplus x_3)$;
7) $f_7 = \neg(x_3)$;	17) $f_{17} = \neg(x_1 \oplus x_4)$;	27) $f_{27} = \neg(x_1 \oplus x_2 \oplus x_4)$;
8) $f_8 = \neg(x_4)$;	18) $f_{18} = \neg(x_2 \oplus x_3)$;	28) $f_{28} = \neg(x_1 \oplus x_3 \oplus x_4)$;
9) $f_9 = x_1 \oplus x_2$;	19) $f_{19} = \neg(x_2 \oplus x_4)$;	29) $f_{29} = \neg(x_2 \oplus x_3 \oplus x_4)$;
10) $f_{10} = x_1 \oplus x_3$;	20) $f_{20} = \neg(x_3 \oplus x_4)$;	30) $f_{30} = \neg(x_1 \oplus x_2 \oplus x_3 \oplus x_4)$;

DOI 10.15588/1607-3274-2019-4-19

the whole set of values at the input and output of the BBFs of DCT.

1. Let's create Table 2 (truth table) of the values of the BBFs from Table 1 for all possible sets of values of the variables $x_1,...,x_4$.

2. Let's create a Table 3 which contains only those BBFs that give the value of 0 for a given set of values $x_1,...,x_4$.

3. Let's create a Table 4 which contains only those BBFs that give the value of 1 for a given set of values $x_1,...,x_4$.

Table 2 – The values of the BBFs from Table 1 for all possible sets of values of the variables $x_1,...,x_4$

x_1 $=f_1$	x_2 $=f_2$	x_3 $=f_3$	x_4 $=f_4$	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}	f_{27}	f_{28}	f_{29}	f_{30}
0	0	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	1	1	1	1	1
0	0	0	1	1	1	1	0	0	0	1	0	1	1	1	1	0	1	0	0	0	1	1	1	1	1	0	0	0	0
0	0	1	0	1	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	1	1	0	1	0	0	0
0	1	0	0	1	0	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	0	1	1	0	0	1	0	0
1	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	0	1	0	0	0	1	0
0	0	1	1	1	1	0	0	0	1	1	1	1	0	1	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
0	1	0	1	1	0	1	0	1	0	1	1	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	1	1
1	0	0	1	0	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	1	0	0	1	1	0	1
0	1	1	0	1	0	0	1	1	1	0	0	1	1	0	0	1	1	0	0	0	1	1	0	0	1	0	0	1	1
1	0	1	0	0	1	0	1	1	0	1	1	0	1	0	1	0	0	1	0	0	1	0	1	0	1	0	1	0	1
1	1	0	0	0	0	1	1	0	1	1	1	1	0	1	0	0	0	0	1	0	0	1	1	0	1	1	0	0	1
0	1	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	1	1	1	1	1	0	0
1	0	1	1	0	1	0	0	1	0	0	1	1	0	0	1	1	0	0	1	0	0	1	0	1	1	1	0	1	0
1	1	0	1	0	0	1	0	0	1	0	1	0	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	0
1	1	1	0	0	0	0	1	0	0	1	0	1	1	1	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1

Table 3 – The BBFs from the Table 1, that give the value of 0 for a given set of values $x_1,...,x_4$ for each row

x_1 $=f_1$	x_2 $=f_2$	x_3 $=f_3$	x_4 $=f_4$	The value of function	The functions from the set $\{f_1, f_2, ... f_{30}\}$ that gives 0 as a result for the specified values $x_1, ...,x_4$ for each row													
0	0	0	0	0	f_1	f_2	f_3	f_4	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}
0	0	0	1		f_1	f_2	f_3	f_8	f_9	f_{10}	f_{12}	f_{17}	f_{19}	f_{20}	f_{21}	f_{27}	f_{28}	f_{29}	f_{30}
0	0	1	0		f_1	f_2	f_4	f_7	f_9	f_{11}	f_{13}	f_{16}	f_{18}	f_{20}	f_{22}	f_{26}	f_{28}	f_{29}	f_{30}
0	1	0	0		f_1	f_3	f_4	f_6	f_{10}	f_{11}	f_{14}	f_{15}	f_{18}	f_{19}	f_{23}	f_{26}	f_{27}	f_{29}	f_{30}
1	0	0	0		f_2	f_3	f_4	f_5	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{24}	f_{26}	f_{27}	f_{28}	f_{30}
0	0	1	1		f_1	f_2	f_7	f_8	f_9	f_{14}	f_{16}	f_{17}	f_{18}	f_{19}	f_{23}	f_{24}	f_{25}	f_{26}	f_{27}
0	1	0	1		f_1	f_3	f_6	f_8	f_{10}	f_{13}	f_{15}	f_{17}	f_{18}	f_{20}	f_{22}	f_{24}	f_{25}	f_{26}	f_{28}
1	0	0	1		f_2	f_3	f_5	f_8	f_{11}	f_{12}	f_{15}	f_{16}	f_{19}	f_{20}	f_{22}	f_{23}	f_{25}	f_{26}	f_{29}
0	1	1	0		f_1	f_4	f_6	f_7	f_{11}	f_{12}	f_{15}	f_{16}	f_{19}	f_{20}	f_{21}	f_{24}	f_{25}	f_{27}	f_{28}
1	0	1	0		f_2	f_4	f_5	f_7	f_{10}	f_{13}	f_{15}	f_{17}	f_{18}	f_{20}	f_{21}	f_{23}	f_{25}	f_{27}	f_{29}
1	1	0	0		f_3	f_4	f_5	f_6	f_9	f_{14}	f_{16}	f_{17}	f_{18}	f_{19}	f_{21}	f_{22}	f_{25}	f_{28}	f_{29}
0	1	1	1		f_1	f_6	f_7	f_8	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{21}	f_{22}	f_{23}	f_{29}	f_{30}
1	0	1	1		f_2	f_5	f_7	f_8	f_{10}	f_{11}	f_{14}	f_{15}	f_{18}	f_{19}	f_{21}	f_{22}	f_{24}	f_{28}	f_{30}
1	1	0	1		f_3	f_5	f_6	f_8	f_9	f_{11}	f_{13}	f_{16}	f_{18}	f_{20}	f_{21}	f_{23}	f_{24}	f_{27}	f_{30}
1	1	1	0		f_4	f_5	f_6	f_7	f_9	f_{10}	f_{12}	f_{17}	f_{19}	f_{20}	f_{22}	f_{23}	f_{24}	f_{26}	f_{30}
1	1	1	1		f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{25}	f_{26}	f_{27}	f_{28}	f_{29}

Table 4 – The BBFs from the Table 1, that give the value of 1 for a given set of values $x_1,...,x_4$ for each row

x_1 $=f_1$	x_2 $=f_2$	x_3 $=f_3$	x_4 $=f_4$	The value of function	The functions from the set $\{f_1, f_2, ... f_{30}\}$ that gives 1 as a result for the specified values $x_1, ..., x_4$ for each row													
0	0	0	0	1	f_5	f_6	f_7	f_8	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{26}	f_{27}	f_{28}	f_{29}	f_{30}
0	0	0	1		f_4	f_5	f_6	f_7	f_{11}	f_{13}	f_{14}	f_{15}	f_{16}	f_{18}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}
0	0	1	0		f_3	f_5	f_6	f_8	f_{10}	f_{12}	f_{14}	f_{15}	f_{17}	f_{19}	f_{21}	f_{23}	f_{24}	f_{25}	f_{27}
0	1	0	0		f_2	f_5	f_7	f_8	f_9	f_{12}	f_{13}	f_{16}	f_{17}	f_{20}	f_{21}	f_{22}	f_{24}	f_{25}	f_{28}
1	0	0	0		f_1	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{25}	f_{29}
0	0	1	1		f_3	f_4	f_5	f_6	f_{10}	f_{11}	f_{12}	f_{13}	f_{15}	f_{20}	f_{21}	f_{22}	f_{28}	f_{29}	f_{30}
0	1	0	1		f_2	f_4	f_5	f_7	f_9	f_{11}	f_{12}	f_{14}	f_{16}	f_{19}	f_{21}	f_{23}	f_{27}	f_{29}	f_{30}
1	0	0	1		f_1	f_4	f_6	f_7	f_9	f_{10}	f_{13}	f_{14}	f_{17}	f_{18}	f_{21}	f_{24}	f_{27}	f_{28}	f_{30}
0	1	1	0		f_2	f_3	f_5	f_8	f_9	f_{10}	f_{13}	f_{14}	f_{17}	f_{18}	f_{22}	f_{23}	f_{26}	f_{29}	f_{30}
1	0	1	0		f_1	f_3	f_6	f_8	f_9	f_{11}	f_{12}	f_{14}	f_{16}	f_{19}	f_{22}	f_{24}	f_{26}	f_{28}	f_{30}
1	1	0	0		f_1	f_2	f_7	f_8	f_{10}	f_{11}	f_{12}	f_{13}	f_{15}	f_{20}	f_{23}	f_{24}	f_{26}	f_{27}	f_{30}
0	1	1	1		f_2	f_3	f_4	f_5	f_9	f_{10}	f_{11}	f_{18}	f_{19}	f_{20}	f_{24}	f_{25}	f_{26}	f_{27}	f_{28}
1	0	1	1		f_1	f_3	f_4	f_6	f_9	f_{12}	f_{13}	f_{16}	f_{17}	f_{20}	f_{23}	f_{25}	f_{26}	f_{27}	f_{29}
1	1	0	1		f_1	f_2	f_4	f_7	f_{10}	f_{12}	f_{14}	f_{15}	f_{17}	f_{19}	f_{22}	f_{25}	f_{26}	f_{28}	f_{29}
1	1	1	0		f_1	f_2	f_3	f_8	f_{11}	f_{13}	f_{14}	f_{15}	f_{16}	f_{18}	f_{21}	f_{25}	f_{27}	f_{28}	f_{29}
1	1	1	1		f_1	f_2	f_3	f_4	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{30}

DOI 10.15588/1607-3274-2019-4-19

4. Let's take a DCT with the property of the SAC, for example

$$F_1^d = \begin{bmatrix} \neg(x_1 \oplus x_2 \oplus x_4) \\ \neg(x_1 \oplus x_2 \oplus x_3) \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_{27} \\ f_{26} \\ f_1 \\ f_2 \end{bmatrix} \quad (3)$$

and we will find ICT F_1^r for this CT. To do this, create a Table 5 from Table 2. In the left part of Table 5, we write the input values of the variables $x_1,...,x_4$, and in the right part of Table 5 the values of the four BBFs of the DCT. We take functions and their values from Table 2.

Table 5 – Sets of values of variables and their pertinent values of the BBFs of the DCT

The values of variables				The values of the BBFs of a DCT F_1^d			
$x_1=f_1$	$x_2=f_2$	$x_3=f_3$	$x_4=f_4$	f_{27}	f_{26}	$x_1=f_1$	$x_2=f_2$
0	0	0	0	1	1	0	0
0	0	0	1	0	1	0	0
0	0	1	0	1	0	0	0
0	1	0	0	0	0	0	1
1	0	0	0	0	0	1	0
0	0	1	1	0	0	0	0
0	1	0	1	1	0	0	1
1	0	0	1	1	0	1	0
0	1	1	0	0	1	0	1
1	0	1	0	0	1	1	0
1	1	0	0	1	1	1	1
0	1	1	1	1	1	0	1
1	0	1	1	1	1	1	0
1	1	0	1	0	1	1	1
1	1	1	0	1	0	1	1
1	1	1	1	0	0	1	1

5. Since the input values of any DCT are the output values of the pertinent ICT (if it exists), and the output values of the DCT are the input values of the ICT, we will change the places of the left and the right side of Table 5, that is, the set of input and output values of the DCT ($F_{p1}^r, F_{p2}^r, F_{p3}^r, F_{p4}^r$ – known sets of values of the pertinent four unknown BBFs of the ICP). The F_{pi}^r is the *i*-th part of the $F_1^r \in B_{n,n}$. As a result, we obtain Table 6.

Table 6 – Sets of values of variables and their pertinent values of the BBFs of the ICT

The values of variables				The values of unknown BBFs of the ICT F_1^r			
$x_1=f_1$	$x_2=f_2$	$x_3=f_3$	$x_4=f_4$	F_{p1}^r	F_{p2}^r	F_{p3}^r	F_{p4}^r
1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1
1	0	0	0	0	0	1	0
0	0	0	1	0	1	0	0
0	0	1	0	1	0	0	0
0	0	0	0	0	0	1	1
1	0	0	1	0	1	0	1
1	0	1	0	1	0	0	1
0	1	0	1	0	1	1	0
0	1	1	0	1	0	1	0
1	1	1	1	1	1	0	0
1	1	0	1	0	1	1	1
1	1	1	0	1	0	1	1
0	1	1	1	1	1	0	1
1	0	1	1	1	1	1	0
0	0	1	1	1	1	1	1

6. Let's create a Table 7 where we indicate those BBFs, whose calculation result is equal to the value F_{p1}^r of the first BBF of the ICT F_1^r for each set of values $x_1,...,x_4$.

Table 7 – All possible BBFs that give the values of the first BBF F_{p1}^r of the ICT F_1^r

The values of variables				F_{p1}^r (the values of the first BBF of the ICT F_1^r)	The BBFs from the Table 1 that give the value equal to the value of the first BBF F_{p1}^r of the ICT F_1^r for the specified values $x_1,...,x_4$ for each row														
$x_1=f_1$	$x_2=f_2$	$x_3=f_3$	$x_4=f_4$																
1	1	0	0	0	$\boldsymbol{f_3}$	f_4	f_5	f_6	f_9	f_{14}	f_{16}	f_{17}	f_{18}	f_{19}	f_{21}	f_{22}	f_{25}	f_{28}	f_{29}
0	1	0	0	0	f_1	$\boldsymbol{f_3}$	f_4	f_6	f_{10}	f_{11}	f_{14}	f_{15}	f_{18}	f_{19}	f_{23}	f_{26}	f_{27}	f_{29}	f_{30}
1	0	0	0	0	f_2	$\boldsymbol{f_3}$	f_4	f_5	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{24}	f_{26}	f_{27}	f_{28}	f_{30}
0	0	0	1	0	f_1	f_2	$\boldsymbol{f_3}$	f_8	f_9	f_{10}	f_{12}	f_{17}	f_{19}	f_{20}	f_{21}	f_{27}	f_{28}	f_{29}	f_{30}
0	0	1	0	1	$\boldsymbol{f_3}$	f_5	f_6	f_8	f_{10}	f_{12}	f_{14}	f_{15}	f_{17}	f_{19}	f_{21}	f_{23}	f_{24}	f_{25}	f_{27}
0	0	0	0	0	f_1	f_2	$\boldsymbol{f_3}$	f_4	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}
1	0	0	1	0	f_2	$\boldsymbol{f_3}$	f_5	f_8	f_{11}	f_{12}	f_{15}	f_{16}	f_{19}	f_{20}	f_{22}	f_{23}	f_{25}	f_{26}	f_{29}
1	0	1	0	1	f_1	$\boldsymbol{f_3}$	f_6	f_8	f_9	f_{11}	f_{12}	f_{14}	f_{16}	f_{19}	f_{22}	f_{24}	f_{26}	f_{28}	f_{30}
0	1	0	1	0	f_1	$\boldsymbol{f_3}$	f_6	f_8	f_{10}	f_{13}	f_{15}	f_{17}	f_{18}	$\boldsymbol{f_{20}}$	f_{22}	f_{24}	f_{25}	f_{26}	f_{28}
0	1	1	0	1	f_2	$\boldsymbol{f_3}$	f_5	f_8	f_9	f_{10}	f_{13}	f_{14}	f_{17}	f_{18}	f_{22}	f_{23}	f_{26}	f_{29}	f_{30}
1	1	1	1	1	f_1	f_2	$\boldsymbol{f_3}$	f_4	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{30}
1	1	0	1	0	$\boldsymbol{f_3}$	f_5	f_6	f_8	f_9	f_{11}	f_{13}	f_{16}	f_{18}	f_{20}	f_{21}	f_{23}	f_{24}	f_{27}	f_{30}
1	1	1	0	1	f_1	f_2	$\boldsymbol{f_3}$	f_8	f_{11}	f_{13}	f_{14}	f_{15}	f_{16}	f_{18}	f_{21}	f_{25}	f_{27}	f_{28}	f_{29}
0	1	1	1	1	f_2	$\boldsymbol{f_3}$	f_4	f_5	f_9	f_{10}	f_{11}	f_{18}	f_{19}	f_{20}	f_{24}	f_{25}	f_{26}	f_{27}	f_{28}
1	0	1	1	1	f_1	$\boldsymbol{f_3}$	f_4	f_6	f_9	f_{12}	f_{13}	f_{16}	f_{17}	f_{20}	f_{23}	f_{25}	f_{26}	f_{27}	f_{29}
0	0	1	1	1	$\boldsymbol{f_3}$	f_4	f_5	f_6	f_{10}	f_{11}	f_{12}	f_{13}	f_{15}	f_{20}	f_{21}	f_{22}	f_{28}	f_{29}	f_{30}

DOI 10.15588/1607-3274-2019-4-19

It can be seen from Table 7 that only the BBF f_3 from the set of functions of the Table 1 satisfies all the values F_{p1}^r from the Table 7. Therefore, $f_3 = F_{p1}^r$ is the first BBF of the ICT F_1^r.

7. Similarly, we create Table 8 for the second BBF F_{p2}^r of the ICT F_1^r.

It can be seen from Table 8 that only the BBF f_4 from the set of functions of the Table 1 satisfies all the values F_{p2}^r from the Table 8. Therefore, $f_4 = F_{p2}^r$ is the second BBF of the ICT F_1^r.

8. Similarly, we create Table 9 for the third BBF F_{p3}^r of the ICT F_1^r.

It can be seen from Table 9 that only the BBF f_{29} from the set of functions of the Table 1 satisfies all the values F_{p3}^r from the Table 9. Therefore, $f_{29} = F_{p3}^r$ is the third BBF of the ICT F_1^r.

9. Similarly, we create Table 10 for the forth BBF F_{p4}^r of the ICT F_1^r.

Table 8 – All possible BBFs that give the values of the second BBF F_{p2}^r of the ICT F_1^r

The values of variables				F_{p2}^r (the values of the second BBF of the ICT F_1^r)	The BBFs from the Table 1 that give the value equal to the value of the second BBF F_{p2}^r of the ICT F_1^r for the specified values $x_1,\ldots,x_4$ for each row													
x_1 =f_1	x_2 =f_2	x_3 =f_3	x_4 =f_4															
1	1	0	0	0	f_3	**f_4**	f_5	f_6	f_9	f_{14}	f_{16}	f_{17}	f_{18}	f_{19}	f_{21}	f_{22}	f_{25}	f_{28}	f_{29}
0	1	0	0	0	f_1	f_3	**f_4**	f_6	f_{10}	f_{11}	f_{14}	f_{15}	f_{18}	f_{19}	f_{23}	f_{26}	f_{27}	f_{29}	f_{30}
1	0	0	0	0	f_2	f_3	**f_4**	f_5	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{24}	f_{26}	f_{27}	f_{28}	f_{30}
0	0	0	1	1	**f_4**	f_5	f_6	f_7	f_{11}	f_{13}	f_{14}	f_{15}	f_{16}	f_{18}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}
0	0	1	0	0	f_1	f_2	**f_4**	f_7	f_9	f_{11}	f_{13}	f_{16}	f_{18}	f_{20}	f_{22}	f_{26}	f_{28}	f_{29}	f_{30}
0	0	0	0	0	f_1	f_2	f_3	**f_4**	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}
1	0	0	1	1	f_1	**f_4**	f_6	f_7	f_9	f_{10}	f_{13}	f_{14}	f_{17}	f_{18}	f_{21}	f_{24}	f_{27}	f_{28}	f_{30}
1	0	1	0	0	f_2	**f_4**	f_5	f_7	f_{10}	f_{13}	f_{15}	f_{17}	f_{18}	f_{20}	f_{21}	f_{23}	f_{25}	f_{27}	f_{29}
0	1	0	1	1	f_2	**f_4**	f_5	f_7	f_9	f_{11}	f_{12}	f_{14}	f_{16}	f_{19}	f_{21}	f_{23}	f_{27}	f_{29}	f_{30}
0	1	1	0	0	f_1	**f_4**	f_6	f_7	f_{11}	f_{12}	f_{15}	f_{16}	f_{19}	f_{20}	f_{21}	f_{24}	f_{25}	f_{27}	f_{28}
1	1	1	1	1	f_1	f_2	f_3	**f_4**	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{30}
1	1	0	1	1	f_1	f_2	**f_4**	f_7	f_{10}	f_{12}	f_{14}	f_{15}	f_{17}	f_{19}	f_{22}	f_{25}	f_{26}	f_{28}	f_{29}
1	1	1	0	0	**f_4**	f_5	f_6	f_7	f_9	f_{10}	f_{12}	f_{17}	f_{19}	f_{20}	f_{22}	f_{23}	f_{24}	f_{26}	f_{30}
0	1	1	1	1	f_2	f_3	**f_4**	f_5	f_9	f_{10}	f_{11}	f_{18}	f_{19}	f_{20}	f_{24}	f_{25}	f_{26}	f_{27}	f_{28}
1	0	1	1	1	f_1	f_3	**f_4**	f_6	f_9	f_{12}	f_{13}	f_{16}	f_{17}	f_{20}	f_{23}	f_{25}	f_{26}	f_{27}	f_{29}
0	0	1	1	1	f_3	**f_4**	f_5	f_6	f_{10}	f_{11}	f_{12}	f_{13}	f_{15}	f_{20}	f_{21}	f_{22}	f_{28}	f_{29}	f_{30}

Table 9 – All possible BBFs that give the value of the third BBF F_{p3}^r of the ICT F_1^r

The values of variables				F_{p3}^r (the values of the third BBF of the ICT F_1^r)	The BBFs from the Table 1 that give the value equal to the value of the third BBF F_{p3}^r of the ICT F_1^r for the specified values $x_1,\ldots,x_4$ for each row													
x_1 =f_1	x_2 =f_2	x_3 =f_3	x_4 =f_4															
1	1	0	0	0	f_3	f_4	f_5	f_6	f_9	f_{14}	f_{16}	f_{17}	f_{18}	f_{19}	f_{21}	f_{22}	f_{25}	f_{28}	**f_{29}**
0	1	0	0	0	f_1	f_3	f_4	f_6	f_{10}	f_{11}	f_{14}	f_{15}	f_{18}	f_{19}	f_{23}	f_{26}	f_{27}	**f_{29}**	f_{30}
1	0	0	0	1	f_1	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{25}	**f_{29}**
0	0	0	1	0	f_1	f_2	f_3	f_8	f_9	f_{10}	f_{12}	f_{17}	f_{19}	f_{20}	f_{21}	f_{27}	f_{28}	**f_{29}**	f_{30}
0	0	1	0	0	f_1	f_2	f_4	f_7	f_9	f_{11}	f_{13}	f_{16}	f_{18}	f_{20}	f_{22}	f_{26}	f_{28}	**f_{29}**	f_{30}
0	0	0	0	1	f_5	f_6	f_7	f_8	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{26}	f_{27}	f_{28}	**f_{29}**	f_{30}
1	0	0	1	0	f_2	f_3	f_5	f_8	f_{11}	f_{12}	f_{15}	f_{16}	f_{19}	f_{20}	f_{22}	f_{23}	f_{25}	f_{26}	**f_{29}**
1	0	1	0	0	f_2	f_4	f_5	f_7	f_{10}	f_{13}	f_{15}	f_{17}	f_{18}	f_{20}	f_{21}	f_{23}	f_{25}	f_{27}	**f_{29}**
0	1	0	1	1	f_2	f_4	f_5	f_7	f_9	f_{11}	f_{12}	f_{14}	f_{16}	f_{19}	f_{21}	f_{23}	f_{27}	**f_{29}**	f_{30}
0	1	1	0	1	f_2	f_3	f_5	f_8	f_9	f_{10}	f_{13}	f_{14}	f_{17}	f_{18}	f_{22}	f_{23}	f_{26}	**f_{29}**	f_{30}
1	1	1	1	0	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{25}	f_{26}	f_{27}	f_{28}	**f_{29}**
1	1	0	1	1	f_1	f_2	f_4	f_7	f_{10}	f_{12}	f_{14}	f_{15}	f_{17}	f_{19}	f_{22}	f_{25}	f_{26}	f_{28}	**f_{29}**
1	1	1	0	1	f_1	f_2	f_3	f_8	f_{11}	f_{13}	f_{14}	f_{15}	f_{16}	f_{18}	f_{21}	f_{25}	f_{27}	f_{28}	**f_{29}**
0	1	1	1	0	f_1	f_6	f_7	f_8	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{21}	f_{22}	f_{23}	**f_{29}**	f_{30}
1	0	1	1	1	f_1	f_3	f_4	f_6	f_9	f_{12}	f_{13}	f_{16}	f_{17}	f_{20}	f_{23}	f_{25}	f_{26}	f_{27}	**f_{29}**
0	0	1	1	1	f_3	f_4	f_5	f_6	f_{10}	f_{11}	f_{12}	f_{13}	f_{15}	f_{20}	f_{21}	f_{22}	f_{28}	**f_{29}**	f_{30}

DOI 10.15588/1607-3274-2019-4-19

Table 10 – All possible BBFs that give the value of the forth BBF F_{p4}^r of the ICT F_1^r

The values of variables				F_{p4}^r (the values of the forth BBF of the ICT F_1^r)	The BBFs from the Table 1 that give the value equal to the value of the forth BBF F_{p4}^r of the ICT F_1^r for the specified values $x_1,...,x_4$ for each row													
$x_1=f_1$	$x_2=f_2$	$x_3=f_3$	$x_4=f_4$															
1	1	0	0	0	f_3	f_4	f_5	f_6	f_9	f_{14}	f_{16}	f_{17}	f_{18}	f_{19}	f_{21}	f_{22}	f_{25}	$\boldsymbol{f_{28}}$	f_{29}
0	1	0	0	1	f_2	f_5	f_7	f_8	f_9	f_{12}	f_{13}	f_{16}	f_{17}	f_{20}	f_{21}	f_{22}	f_{24}	f_{25}	$\boldsymbol{f_{28}}$
1	0	0	0	0	f_2	f_3	f_4	f_5	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{24}	f_{26}	f_{27}	$\boldsymbol{f_{28}}$	f_{30}
0	0	0	1	0	f_1	f_2	f_3	f_8	f_9	f_{10}	f_{12}	f_{17}	f_{19}	f_{20}	f_{21}	f_{27}	$\boldsymbol{f_{28}}$	f_{29}	f_{30}
0	0	1	0	0	f_1	f_2	f_4	f_7	f_9	f_{11}	f_{13}	f_{16}	f_{18}	f_{20}	f_{22}	f_{26}	$\boldsymbol{f_{28}}$	f_{29}	f_{30}
0	0	0	0	1	f_5	f_6	f_7	f_8	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{26}	f_{27}	$\boldsymbol{f_{28}}$	f_{29}	f_{30}
1	0	0	1	1	f_1	f_4	f_6	f_7	f_9	f_{10}	f_{13}	f_{14}	f_{17}	f_{18}	f_{21}	f_{24}	f_{27}	$\boldsymbol{f_{28}}$	f_{30}
1	0	1	0	1	f_1	f_3	f_6	f_8	f_9	f_{11}	f_{12}	f_{14}	f_{16}	f_{19}	f_{22}	f_{24}	f_{26}	$\boldsymbol{f_{28}}$	f_{30}
0	1	0	1	0	f_1	f_3	f_6	f_8	f_{10}	f_{13}	f_{15}	f_{17}	f_{18}	f_{20}	f_{22}	f_{24}	f_{25}	f_{26}	$\boldsymbol{f_{28}}$
0	1	1	0	0	f_1	f_4	f_6	f_7	f_{11}	f_{12}	f_{15}	f_{16}	f_{19}	f_{20}	f_{21}	f_{24}	f_{25}	f_{27}	$\boldsymbol{f_{28}}$
1	1	1	1	0	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{25}	f_{26}	f_{27}	$\boldsymbol{f_{28}}$	f_{29}
1	1	0	1	1	f_1	f_2	f_4	f_7	f_{10}	f_{12}	f_{14}	f_{15}	f_{17}	f_{19}	f_{22}	f_{25}	f_{26}	$\boldsymbol{f_{28}}$	f_{29}
1	1	1	0	1	f_1	f_2	f_3	f_8	f_{11}	f_{13}	f_{14}	f_{15}	f_{16}	f_{18}	f_{21}	f_{25}	f_{27}	$\boldsymbol{f_{28}}$	f_{29}
0	1	1	1	1	f_2	f_3	f_4	f_5	f_9	f_{10}	f_{11}	f_{18}	f_{19}	f_{20}	f_{24}	f_{25}	f_{26}	f_{27}	$\boldsymbol{f_{28}}$
1	0	1	1	0	f_2	f_5	f_7	f_8	f_{10}	f_{11}	f_{14}	f_{15}	f_{18}	f_{19}	f_{21}	f_{22}	f_{24}	$\boldsymbol{f_{28}}$	f_{30}
0	0	1	1	1	f_3	f_4	f_5	f_6	f_{10}	f_{11}	f_{12}	f_{13}	f_{15}	f_{20}	f_{21}	f_{22}	$\boldsymbol{f_{28}}$	f_{29}	f_{30}

It can be seen from Table 10 that only the BBF f_{28} from the set of functions of the Table 1 satisfies all the values F_{p4}^r from the Table 10. Therefore, $f_{28} = F_{p4}^r$ is the forth BBF of the ICT F_1^r.

As a result, the ICT for the DCT (3) has a view:

$$F_1^r = \begin{bmatrix} f_3 \\ f_4 \\ f_{29} \\ f_{28} \end{bmatrix} = \begin{bmatrix} x_3 \\ x_4 \\ \neg(x_2 \oplus x_3 \oplus x_4) \\ \neg(x_1 \oplus x_3 \oplus x_4) \end{bmatrix}. \quad (4)$$

This method provides the construction of ICTs for four variables and two logical operations (inversion and addition modulo 2), but can be extended to a larger even number of variables.

4 EXPERIMENTS

We prove that the resulting CT (4) is indeed the inverse of the CT (3).

As known, the composition $f \circ g$ of BFs $f(x_1,...,x_n) \in B^n$ and $g(x_1,...,x_n) \in B^n$ is a function defined by $(f \circ g)(x_1,...,x_n) = f(g(x_1,...,x_n))$.

As known, the BF with n inputs and n outputs $F(x_1,...,x_n) = (f_1(x_1,...,x_n),..., f_n(x_1,...,x_n))$ has the inverse BF with n inputs and n outputs $F^{-1}(x_1,...,x_n) = (f_1^{-1}(x_1,...,x_n),..., f_n^{-1}(x_1,...,x_n))$, if the following equalities hold: $F^{-1}(F(x_1,...,x_n)) = F(F^{-1}(x_1,...,x_n)) = (x_1,...,x_n)$.

For DCT F_1^d from Table 11 we have $F(x_1,...,x_4) = (\neg(x_1 \oplus x_2 \oplus x_4), \neg(x_1 \oplus x_2 \oplus x_3), x_1, x_2)$ and for ICT F_1^r from Table 11 we have

$F^{-1}(x_1,..., x_4) = (x_3, x_4, \neg(x_2 \oplus x_3 \oplus x_4), \neg(x_1 \oplus x_3 \oplus x_4))$.

Indeed,

$f_1^{-1}(f_1(x_1,...,x_4)) = x_1; \quad f_2^{-1}(f_2(x_1,...,x_4)) = x_2;$

$f_3^{-1}(f_3(x_1,...,x_4)) = \neg(\neg(x_1 \oplus x_2 \oplus x_3) \oplus x_1 \oplus x_2))$

$= x_1 \oplus x_2 \oplus x_3 \oplus \neg x_1 \oplus \neg x_2 = 1 \oplus 1 \oplus x_3 = x_3;$

$f_4^{-1}(f_4(x_1,...,x_4)) = \neg(\neg(x_1 \oplus x_2 \oplus x_4) \oplus (x_1 \oplus x_2)$

$= x_1 \oplus x_2 \oplus x_4 \oplus \neg x_1 \oplus \neg x_2 = 1 \oplus 1 \oplus x_4 = x_4;$

Thus, $F^{-1}(F(x_1,...,x_4)) = (x_1, x_2, x_3, x_4)$.

Conversely,

$f_1(f_1^{-1}(x_1,...,x_4)) = \neg(x_3 \oplus x_4 \oplus \neg(x_1 \oplus x_3 \oplus x_4))$

$= \neg x_3 \oplus \neg x_4 \oplus x_1 \oplus x_3 \oplus x_4 = 1 \oplus 1 \oplus x_1 = x_1;$

$f_2(f_2^{-1}(x_1,...,x_4)) = \neg(x_3 \oplus x_4 \oplus \neg(x_2 \oplus x_3 \oplus x_4))$

$= \neg x_3 \oplus \neg x_4 \oplus x_2 \oplus x_3 \oplus x_4 = 1 \oplus 1 \oplus x_2 = x_2;$

$f_3(f_3^{-1}(x_1,...,x_4)) = x_3; \quad f_4(f_4^{-1}(x_1,...,x_4)) = x_4;$

Thus, $F(F^{-1}(x_1,...,x_4)) = (x_1, x_2, x_3, x_4)$.

Consequently, DCT (3) has ICT (4).

We prove that the ICT F_2^r

$$F_2^r = \begin{bmatrix} x_1 \oplus x_2 \oplus x_4 \\ x_1 \oplus x_3 \oplus x_4 \\ \neg(x_2 \oplus x_3 \oplus x_4) \\ \neg(x_1 \oplus x_2 \oplus x_3) \end{bmatrix}. \quad (5)$$

is indeed the inverse of the DCT F_2^d

DOI 10.15588/1607-3274-2019-4-19

$$F_2^d = \begin{bmatrix} \neg(x_1 \oplus x_2 \oplus x_4) \\ x_1 \oplus x_3 \oplus x_4 \\ x_2 \oplus x_3 \oplus x_4 \\ \neg(x_1 \oplus x_2 \oplus x_3) \end{bmatrix}. \quad (6)$$

For DCT F_2^d we have

$F(x_1,...,x_4) = (\neg(x_1 \oplus x_2 \oplus x_4), (x_1 \oplus x_3 \oplus x_4),$
$(x_2 \oplus x_3 \oplus x_4), \neg(x_1 \oplus x_2 \oplus x_3));$

and for ICT F_2^r we have

$F^{-1}(x_1,...,x_4) = ((x_1 \oplus x_2 \oplus x_4), (x_1 \oplus x_3 \oplus x_4),$
$\neg(x_2 \oplus x_3 \oplus x_4), \neg(x_1 \oplus x_2 \oplus x_3)).$

Indeed,

$f_1^{-1}(f_1(x_1,...,x_4)) = \neg(x_1 \oplus x_2 \oplus x_4)$
$\oplus (x_1 \oplus x_3 \oplus x_4) \oplus \neg(x_1 \oplus x_2 \oplus x_3) = \neg(x_1 \oplus x_2)$
$\oplus \neg(x_1 \oplus x_2) \oplus \neg x_4 \oplus \neg x_3 \oplus (x_4 \oplus x_3 \oplus x_1) =$
$\neg x_4 \oplus \neg x_3 \oplus x_4 \oplus x_3 \oplus x_1 = 1 \oplus 1 \oplus x_1 = x_1;$

$f_2^{-1}(f_2(x_1,...,x_4)) = \neg(x_1 \oplus x_2 \oplus x_4) \oplus (x_2 \oplus x_3 \oplus x_4)$
$\oplus \neg(x_1 \oplus x_2 \oplus x_3) = \neg(x_1 \oplus x_2) \oplus \neg(x_1 \oplus x_2) \oplus \neg x_4$
$\oplus \neg x_3 \oplus (x_4 \oplus x_3 \oplus x_2) = \neg x_4 \oplus \neg x_3 \oplus x_4 \oplus x_3$
$\oplus x_2 = 1 \oplus 1 \oplus x_2 = x_2;$

$f_3^{-1}(f_3(x_1,...,x_4)) = \neg((x_1 \oplus x_3 \oplus x_4) \oplus (x_2 \oplus x_3 \oplus x_4)$
$\oplus \neg(x_1 \oplus x_2 \oplus x_3)) = \neg(x_1 \oplus x_3 \oplus x_4) \oplus \neg(x_2 \oplus x_3$
$\oplus x_4) \oplus (x_1 \oplus x_2 \oplus x_3) = \neg x_1 \oplus \neg x_2 \oplus x_1 \oplus x_2 \oplus x_3 =$
$1 \oplus 1 \oplus x_3 = x_3;$

$f_4^{-1}(f_4(x_1,...,x_4)) = \neg(\neg(x_1 \oplus x_2 \oplus x_4) \oplus (x_1 \oplus x_3 \oplus x_4)$
$\oplus \neg(x_2 \oplus x_3 \oplus x_4)) = x_1 \oplus x_2 \oplus x_4 \oplus \neg x_1 \oplus \neg x_2 =$
$1 \oplus 1 \oplus x_4 = x_4.$

Thus, $F^{-1}(F(x_1,...,x_4)) = (x_1, x_2, x_3, x_4)$.

Conversely,

$f_1(f_1^{-1}(x_1,...,x_4)) = \neg((x_1 \oplus x_2 \oplus x_4) \oplus (x_1 \oplus x_3$
$\oplus x_4) \oplus \neg(x_1 \oplus x_2 \oplus x_3)) = \neg(x_1 \oplus x_2 \oplus x_4) \oplus \neg(x_1$
$\oplus x_3 \oplus x_4) \oplus (x_1 \oplus x_2 \oplus x_3) = \neg(x_1 \oplus x_4) \oplus \neg(x_1 \oplus$
$x_4) \oplus \neg x_2 \oplus \neg x_3 \oplus (x_1 \oplus x_2 \oplus x_3) == \neg x_2 \oplus \neg x_3 \oplus$
$x_1 \oplus x_2 \oplus x_3 == 1 \oplus 1 \oplus x_1 = x_1;$

$f_2(f_2^{-1}(x_1,...,x_4)) = \neg((x_1 \oplus x_2 \oplus x_4) \oplus (x_2 \oplus x_3$
$\oplus x_4) \oplus \neg(x_1 \oplus x_2 \oplus x_3))$
$= \neg(x_1 \oplus x_2) \oplus \neg(x_1 \oplus x_2) \oplus \neg x_4 \oplus \neg x_3 \oplus (x_4 \oplus x_3$
$\oplus x_2) = \neg x_4 \oplus \neg x_3 \oplus x_4 \oplus x_3 \oplus x_2 = 1 \oplus 1 \oplus x_2 = x_2;$

$f_3(f_3^{-1}(x_1,...,x_4)) = (x_1 \oplus x_3 \oplus x_4) \oplus \neg(x_2 \oplus x_3$
$\oplus x_4) \oplus \neg(x_1 \oplus x_2 \oplus x_3))$
$= (x_1 \oplus x_3 \oplus x_4) \oplus \neg(x_2 \oplus x_3) \oplus (x_2 \oplus x_3) \oplus \neg x_4$
$\oplus \neg x_1 = x_1 \oplus x_3 \oplus x_4 \oplus \neg x_4 \oplus \neg x_1 = 1 \oplus 1 \oplus x_3 = x_3;$

$f_4(f_4^{-1}(x_1,...,x_4)) = \neg((x_1 \oplus x_2 \oplus x_4) \oplus (x_1 \oplus x_3$
$\oplus x_4) \oplus \neg(x_2 \oplus x_3 \oplus x_4)) = \neg(x_1 \oplus x_2 \oplus x_4) \oplus \neg(x_1$
$\oplus x_3 \oplus x_4) \oplus (x_2 \oplus x_3 \oplus x_4) = \neg(x_1 \oplus x_4) \oplus \neg(x_1 \oplus$
$x_4) \oplus \neg x_2 \oplus \neg x_3 \oplus x_2 \oplus x_3 \oplus x_4 = 1 \oplus 1 \oplus x_4 = x_4.$

Thus, $F(F^{-1}(x_1,...,x_4)) = (x_1, x_2, x_3, x_4)$.

Consequently, DCT F_2^d (6) has ICT F_2^r (5).

5 RESULTS

The results of the construction by this method of two CTs are given in Table 11.

Table 11 – The results of application of the method to selected four-bit CTs satisfying a SAC. Direct and inverse CTs satisfy the SAC

Direct CT	Inverse CT
$F_1^d = \begin{bmatrix} \neg(x_1 \oplus x_2 \oplus x_4) \\ \neg(x_1 \oplus x_2 \oplus x_3) \\ x_1 \\ x_2 \end{bmatrix}$	$F_1^r = \begin{bmatrix} x_3 \\ x_4 \\ \neg(x_2 \oplus x_3 \oplus x_4) \\ \neg(x_1 \oplus x_3 \oplus x_4) \end{bmatrix}$
$F_2^d = \begin{bmatrix} \neg(x_1 \oplus x_2 \oplus x_4) \\ x_1 \oplus x_3 \oplus x_4 \\ x_2 \oplus x_3 \oplus x_4 \\ \neg(x_1 \oplus x_2 \oplus x_3) \end{bmatrix}$	$F_2^r = \begin{bmatrix} x_1 \oplus x_2 \oplus x_4 \\ x_1 \oplus x_3 \oplus x_4 \\ \neg(x_2 \oplus x_3 \oplus x_4) \\ \neg(x_1 \oplus x_2 \oplus x_3) \end{bmatrix}$

6 DISCUSSION

The existing methods of searching for an ICT are methods for calculating each element of the BBFs of the ICT, whereas proposed by us method is a method of choosing existing BBFs from a predetermined set of BBFs for a DCT and an ICT. The method can be extended to a larger even number of bits.

This method can be used to obtain other ICTs, having DCTs that have the property of SAC and for which there is an ICT.

To date, in the general case, the total number of balanced BFs of any number of variables with different sets of logical operations on these variables and having the property of a SAC remains unknown [16]. Therefore, the problem of finding systems of balanced BFs with an even number of variables greater than four for different sets of logical operations and having the property of SAC is a separate important scientific problem that goes beyond the scope of this article.

The article [23] presents methods that handle the inverse problem for the main types of solutions of Boolean

DOI 10.15588/1607-3274-2019-4-19

equations of the form $f(X) = 0$, where $f(X)$: $B^n \to B$ and B is an arbitrary Boolean algebra. The methods [23] are a mixture of purely-algebraic methods and map methods that utilize the variable entered Karnaugh map: (a) Subsumptive general solutions, in which each of the variables is expressed as an interval by deriving successive conjunctive or disjunctive eliminants of the original function, (b) Parametric general solutions, in which each of the variables is expressed via arbitrary parameters which are freely chosen elements of the underlying Boolean algebra and (c) Particular solutions, each of which is an assignment from the underlying Boolean algebra to every pertinent variable that makes the Boolean equation an identity. But the application of these methods to Boolean functions of the form (1) was not considered in [23].

In the article [31] a mathematical formalism is developed, showing the connection of the inverse Boolean function of the form (1) with its corresponding direct Boolean function of the form (1). But the method of obtaining an inverse Boolean function from a direct Boolean function is not specified in [31], and the conditions for the existence of an inverse Boolean function for a given direct Boolean function are not indicated.

But the method developed in this article makes it possible to effectively find the ICT for any four-bit DCT of BFs containing only the operations of inversion and addition modulo two and satisfying the restrictions 1–3, described in section 3 of this article.

In further studies using the method described in this article, it is possible to increase an even number of variables, which will increase the nonlinearity and cryptographic resilience of CTs.

CONCLUSIONS

The urgent problem of obtaining the inversion method of four-bit Boolean SAC cryptotransforms is solved to ensure reliable information protection..

The scientific novelty of obtained results is that the method for obtaining inverse four-bit CTs with the SAC property for balanced BFs containing two logical operations (inversion and addition modulo two) is proposed for the first time.

The practical significance of obtained results is that this method is a method of selecting the already existing basic four-bit BFs from a predetermined set of balanced BBFs for direct and inverse CTs, whereas the existing methods of searching for ICT are methods for calculating each element of the BFs for the ICT.

Prospects for further research are the modifications of this method to the larger even numbers of arguments of the balanced BFs of CTs to increase the cryptographic resilience.

ACKNOWLEDGEMENTS

The authors would like to thank Vice-Rector for Research of Cherkasy State Technological University Dr. Sc., Faure Emil Vitaliiovych, the Associate Professor of the Department of Information Security and Computer Engineering of Cherkasy State Technological University Ph.D., Associate Professor Shvydkyi Valerii Vasylovych and Head of the Department of Statistics and Applied Mathematics of Cherkasy State Technological University Ph.D., Associate Professor Shcherba Anatolii Ivanovych for fruitful discussions.

REFERENCES

1. Kemp S. Digital 2019. Essential insights into how people around the world use the Internet, mobile devices, social media, and e-commerce. [Electronic resource]. Access mode: https: //wearesocial.com/global-digital-report-2019.
2. Lakhtaria K. I. Protecting Computer Network with Encryption Technique: A Study, *Communications in Computer and Information Science (UCMA 2011)*, 2011, Vol. 151, No. PART 2, pp. 381–390. DOI: 10.1007/978-3-642-20998-7_47.
3. Debnath S., Linke N. M., Figgatt C. et al. Demonstration of a small programmable quantum computer with atomic qubits, *Nature,* 2016, No. 536, pp. 63–66. DOI: 10.1038/nature18648
4. Harrow A. W., Montanaro A. Quantum computational supremacy, *Nature*, 2017, Vol. 549, pp. 203–209. DOI: 10.1038/nature23458
5. Smart S. E., Schuster D. I., Mazziotti D. A. Experimental data from a quantum computer verifies the generalized Pauli exclusion principle, *Communications Physics*, 2019, Vol. 2, No. 1, pp. 1–6. DOI: 10.1038/s42005-019-0110-3
6. Kolokotronis N., Limniotis K., Kalouptsidis N. Best Affine and Quadratic Approximations of Particular Classes of Boolean Functions, *IEEE transactions on information theory*, 2009, Vol. 55, No. 11, pp. 5211–5222.
7. Menezes A. J., Van Oorschot P. C., Vanstone S. A. Handbook of Applied Cryptography. USA, CRC Press, Inc. Boca Raton, 1996, 810 p.
8. Alzaidi A. A., Ahmad M., Doja M. N. at al. A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes, *IEEE Access*, 2018, Vol. 6, pp. 55405–55418. DOI: 10.1109/ACCESS.2018.2871557.
9. Steinbach B. Problems and New Solutions in the Boolean Domain, UK, Cambridge Scholars Publishing, Newcastle upon Tyne, 2016, 480 p. ISBN (10): 1–4438-8947-4 ISBN (13): 978-1-4438-8947-6.
10. Nielsen M., Chuang I. Quantum Computation and Quantum Information, UK, Cambridge University Press, 2000, 676 p. ISBN 978-1-107-00217-3.
11. Golubitsky O., Maslov D. A study of optimal 4–bit reversible toffoli circuits and their synthesis, *IEEE Transactions on Computers*, 2012, Vol. 61, No. 9, pp. 1341–1353. DOI: 10.1109/TC.2011.144.
12. Bardis E. G., Bardis N. G., Markovski A. P. at al. Design of Boolean Functions from a great number of variables satisfying strict avalanche criterion, *Proceedings of the IEEE/WSES/IMACS : 3rd World multiconference on circuits, systems, communications and computers, Athens, July 1999: proceedings*. Athens, World Scientific, 1999, pp. 3111–3116.
13. Tang D., Zhang W., Tang X. Construction of balanced Boolean functions with high nonlinearity and good autocorrelation properties, *Designs, Codes and Cryptography*, 2013, Vol. 67, No. 1, pp. 77–99 DOI: 10.1007/s10623-011-9587-9
14. Alzaidi A. A., Ahmad M., Doja M. N. at al. A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes, *IEEE Access*, 2018, Vol. 6, pp. 55405–55418. DOI: 10.1109/ACCESS.2018.2871557

DOI 10.15588/1607-3274-2019-4-19

15. Lloyd S. Counting Functions Satisfying a Higher Order Strict Avalanche Criterion, *EUROCRYPT '89 : Workshop on the Theory and Application of Cryptographic Techniques. Advances in Cryptology, 10–13 April 1989: proceedings*. Berlin, Heidelberg, Springer, 1989, Vol. 434, pp. 63–67. DOI: 10.1007/3-540-46885-4_9
16. Bardis N. G. Combinatorial method for Boolean SAC functions designing, *WSEAS Transactions on Communications*, 2004, Vol. 3, No. 2, pp. 746–752.
17. Gupta Brij B., Dharma P. Agraval, Haoxiang Wang Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives. New York, CRC Press, Taylor & Francis Group, Boca Raton, 2019, 665 p. ISBN 9780815371335.
18. Dey S., Ghosh R. Cryptanalysis of 4-Bit Crypto S-Boxes in Smart Applications, *Security in Smart Cities: Models, Applications, and Challenges. Lecture Notes in Intelligent Transportation and Infrastructure*. Cham, Springer, 2019, P. 211–253. ISBN 978-3-030-01560-2. DOI: 10.1007/978-3-030-01560-2_10.
19. Woods S., Casinovi G. Efficient solution of systems of Boolean equations, *96 Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design, 10–14 November 1996: proceeding.* San Jose, California, ACM Press, 1996, pp. 542–546.
20. Rudeanu S. Boolean sets and most general solutions of Boolean equations, *Information Sciences*, 2010, Vol. 180, No. 12, pp. 2440–2447. DOI: 10.1016/j.ins.2010.01.029.
21. Baneres D., Cortadella J., Kishinevsky M. A Recursive Paradigm to Solve Boolean Relations, *IEEE Transactions on Computers*, 2009, Vol. 58(4), pp. 512–527. http://doi.ieeecomputersociety.org/10.1109/TC.2008.165
22. Rushdi Ali M. A., Motaz H. Amashah Using variable-entered Karnaugh maps to produce compact parametric general solutions of Boolean equations, *International Journal of Computer Mathematics,* 2011, Vol. 88, No. 15, pp. 3136–3149. DOI: 10.1080/00207160.2011.594505.
23. Rushdi A. M. A., H. M. Albarakati The Inverse Problem for Boolean Equations, *Journal of Computer Science*, 2012, Vol. 8, No. 12, pp. 2098–2105. DOI: 10.3844/jcssp.2012.2098.2105
24. Bibilo P. N. Decomposition of Boolean functions based on the solution of logic equations. I. II. III., *Izvestiya Rossijskoj akademii nauk. Teoriya i sistemy upravleniya,* 2002, No. 4, pp. 53–64; 2002, no.5, pp. 57–63.; 2003, no. 6. – P. 88–97.
25. Rudeanu S. On the Decomposition of Boolean Functions via Boolean Equations, *Journal of Universal Computer Science*, 2004, Vol. 10, No. 9, pp. 1294–1301.
26. Primenko É. A. Equivalence classes of invertible Boolean functions, *Cybernetics*, 1984, Vol. 20, No. 6, pp. 771–776. DOI: 10.1007/BF01072161
27. Soeken M., Wille R., Keszocze O. at al. Embedding of Large Boolean Functions for Reversible Logic, *Journal on Emerging Technologies in Computing Systems*, 2016, Vol. 12, № 4, Article No. 41, pp. 41:1–41:26. DOI: 10.1145/2786982.
28. Soeken M. Abdessaied N., De Micheli G. Enumeration of Reversible Functions and Its Application to Circuit Complexity, *Proceedings of the 8th Conference on Reversible Computation (RC 2016), 7–8 July 2016: proceedings*. Bologna: Cham, Springer, 2016, Vol 9720, pp. 255–270. ISBN: 978-3-319-40578-0. DOI: 10.1007/978-3-319-40578-0_19.
29. Kavut S., Maitra S., Tang D. Construction and search of balanced Boolean functions on even number of variables towards excellent autocorrelation profile, *Designs, Codes and Cryptography*, 2019, Vol. 87, No. 2–3, pp. 261–276. DOI: 10.1007/s10623-018-0522-1.
30. Lorens C. S. Invertible Boolean functions, *IEEE Transactions on Electronic Computers*, 1964, Vol. EC–13, No. 5, pp. 529–541. DOI:10.1109/pgec.1964.263724.
31. Varadharajan V., Wu C.-K. Public key cryptosystems based on boolean permutations and their applications, *International Journal of Computer Mathematics*, 2000, Vol. 74, No. 2, pp. 167–184. DOI: 10.1080/00207160008804932.

Received 07.05.2019.
Accepted 26.09.2019.

УДК 004.056

МЕТОД ЗНАХОДЖЕННЯ ОБЕРНЕНИХ ЧОТИРЬОХРОЗРЯДНИХ БУЛЕВИХ КРИПТОПЕРЕТВОРЕНЬ ЗІ СТРОГИМ ЛАВИННИМ КРИТЕРІЄМ

Федотова-Півень І. М. – канд. техн. наук, доцент, доцент кафедри інформаційної безпеки та комп'ютерної інженерії Черкаського державного технологічного університету, Черкаси, Україна.

Рудницький В. М. – д-р техн. наук, професор, завідуючий кафедри інформаційної безпеки та комп'ютерної інженерії Черкаського державного технологічного університету, Черкаси, Україна.

Півень О. Б. – канд. фіз.-мат. наук, доцент, професор кафедри інформаційної безпеки та комп'ютерної інженерії Черкаського державного технологічного університету, Черкаси, Україна.

Миронюк Т. В. – канд. техн. наук, доцент кафедри інформаційної безпеки та комп'ютерної інженерії, Черкаський державний технологічний університет, Черкаси, Україна.

АНОТАЦІЯ

Актуальність. Нелінійні системи булевих функцій грають важливу роль в захисті криптосистем. Створення і використання нових чотирьохрозрядних криптографічних перетворень з нелінійними булевими функціями, що володіють властивістю строгого лавинного критерію, є актуальним завданням підвищення надійності систем захисту інформації.

Метою роботи є створення методу отримання обернених чотирьохбітових криптографічних перетворень з властивістю строгого лавинного критерію, які містять збалансовані булеві функції лише з операціями інверсії і додавання за модулем два.

Метод. Запропоновано метод отримання обернених чотирьохбітових криптографічних перетворень з властивістю строгого лавинного критерію, кожне з яких містить збалансовані булеві функції тільки з операціями інверсії і додавання за модулем два. Метод спрощує процес пошуку обернених криптографічних перетворень шляхом створення класу з тридцяти збалансованих базових булевих функцій з необхідними наперед визначеними обмеженнями і властивостями, а також знаходження в цьому класі базових булевих функцій, що становлять обернене криптографічне перетворення.

DOI 10.15588/1607-3274-2019-4-19

Результати. Показана ефективність методу для отримання двох обернених чотирьохбітових криптографічних перетворень з властивістю строгого лавинного критерію з двох прямих чотирьохбітових криптографічних перетворень з властивістю строгого лавинного критерію.

Висновки. Вперше запропоновано метод отримання обернених чотирьохбітових криптографічних перетворень з властивістю строгого лавинного критерію для збалансованих булевих функцій, що містять дві логічні операції (інверсія і додавання за модулем два) для забезпечення надійного захисту інформації. Цей метод являє собою метод вибору вже існуючих базових булевих функцій з заздалегідь визначеного набору збалансованих базових булевих функцій для прямого і оберненого криптографічних перетворень, тоді як існуючі методи пошуку оберненого криптографічного перетворення є методами обчислення кожного елемента булевих функцій для оберненого криптографічного перетворення. Метод може бути розширений до більшого парного числа аргументів збалансованих булевих функцій криптографічних перетворень для підвищення криптографічного стійкості.

КЛЮЧОВІ СЛОВА: булеві функції, обернене криптографічне перетворення, збалансованість, строгий лавинний критерій, інверсія, додавання за модулем 2.

УДК 004.056

МЕТОД НАХОЖДЕНИЯ ОБРАТНЫХ ЧЕТЫРЕХРАЗРЯДНЫХ БУЛЕВЫХ КРИПТОПРЕОБРАЗОВАНИЙ СО СТРОГИМ ЛАВИННЫМ КРИТЕРИЕМ

Федотова-Пивень И. Н. – канд. техн. наук, доцент, доцент кафедры информационной безопасности и компьютерной инженерии Черкасского государственного технологического университета, Черкассы, Украина.

Рудницкий В. Н. – д-р техн. наук, профессор, заведующий кафедры информационной безопасности и компьютерной инженерии Черкасского государственного технологического университета, Черкассы, Украина.

Пивень О. Б. – канд. физ.-мат. наук, доцент, профессор кафедры информационной безопасности и компьютерной инженерии Черкасского государственного технологического университета, Черкассы, Украина.

Миронюк Т. В. – канд. техн. наук, доцент кафедры информационной безопасности и компьютерной инженерии, Черкасский государственный технологический университет, Черкассы, Украина.

АННОТАЦИЯ

Актуальность. Нелинейные системы булевых функций играют важную роль в защите криптосистем. Создание и использование новых четырехразрядных криптографических преобразований с нелинейными булевыми функциями, обладающими свойством строгого лавинного критерия, является актуальной задачей повышения надежности систем защиты информации. Целью работы является создание метода получения обратных четырехбитовых криптографических преобразований со свойством строгого лавинного критерия, которые содержат сбалансированные булевы функции только с операциями инверсии и сложения по модулю два.

Метод. Предложен метод получения обратных четырехбитовых криптографических преобразований со свойством строгого лавинного критерия, каждое из которых содержит сбалансированные булевы функции только с операциями инверсии и сложения по модулю два. Метод упрощает процесс поиска обратных криптографических преобразований путем создания класса из тридцати сбалансированных базовых булевых функций с требуемыми предопределенными ограничениями и свойствами, а также нахождения в этом классе базовых булевых функций, составляющих обратное криптографическое преобразование.

Результаты. Показана эффективность метода для получения двух обратных четырехбитовых криптографических преобразований со свойством строгого лавинного критерия из двух прямых четырехбитовых криптографических преобразований со свойством строгого лавинного критерия.

Выводы. Впервые был предложен метод получения обратных четырехбитовых криптографических преобразований со свойством строгого лавинного критерия для сбалансированных булевых функций, содержащих две логические операции (инверсия и сложение по модулю два) для обеспечения надежной защиты информации. Этот метод представляет собой метод выбора уже существующих базовых булевых функций из заранее определенного набора сбалансированных базовых булевых функций для прямого и обратного криптографических преобразований, тогда как существующие методы поиска обратного криптографического преобразования представляют собой методы для вычисления каждого элемента булевых функций для обратного криптографического преобразования. Метод может быть расширен до большего четного числа аргументов сбалансированных булевых функций криптографических преобразований для повышения криптографической стойкости.

КЛЮЧЕВЫЕ СЛОВА: булевы функции, обратное криптографическое преобразование, сбалансированность, строгий лавинный критерий, инверсия, сложение по модулю 2.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Kemp S. Digital 2019. Essential insights into how people around the world use the Internet, mobile devices, social media, and e-commerce. [Electronic resource] / S. Kemp. – Access mode: https: //wearesocial.com/global-digital-report-2019.
2. Lakhtaria K. I. Protecting Computer Network with Encryption Technique: A Study / K. I. Lakhtaria // Communications in Computer and Information Science (UCMA 2011). – 2011. – Vol. 151, No. PART 2. – P. 381–390. DOI: 10.1007/978-3-642-20998-7_47.
3. Demonstration of a small programmable quantum computer with atomic qubits / [S. Debnath, N. M. Linke, C. Figgatt et al.] // Nature. – 2016. – № 536. – P. 63–66. DOI: 10.1038/nature18648
4. Harrow A. W. Quantum computational supremacy / A. W. Harrow, A. Montanaro // Nature. – 2017. – Vol. 549. – P. 203–209. DOI: 10.1038/nature23458

DOI 10.15588/1607-3274-2019-4-19

5. Smart S. E. Experimental data from a quantum computer verifies the generalized Pauli exclusion principle / S. E. Smart, D. I. Schuster, D. A. Mazziotti // Communications Physics. – 2019. – Vol. 2, № 1. – P. 1–6. DOI: 10.1038/s42005-019-0110-3
6. Kolokotronis N. Best Affine and Quadratic Approximations of Particular Classes of Boolean Functions / N. Kolokotronis, K. Limniotis, N. Kalouptsidis // IEEE transactions on information theory. – 2009. – Vol. 55, № 11. – P. 5211–5222.
7. Menezes A. J. Handbook of Applied Cryptography / A. J. Menezes, P. C. Van Oorschot, S. A. Vanstone. – USA : CRC Press, Inc. Boca Raton, 1996. – 810 p.
8. A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes / [A. A. Alzaidi, M. Ahmad, M. N. Doja at al.] // IEEE Access. – 2018. – Vol. 6. – P. 55405–55418. DOI: 10.1109/ACCESS.2018.2871557.
9. Steinbach B. Problems and New Solutions in the Boolean Domain / B. Steinbach. – UK: Cambridge Scholars Publishing, Newcastle upon Tyne, 2016. – 480 p. ISBN (10): 1–4438-8947-4 ISBN (13): 978-1-4438-8947-6.
10. Nielsen M. Quantum Computation and Quantum Information / M. Nielsen, I. Chuang. – UK: Cambridge University Press, 2000. – 676 p. ISBN 978-1-107-00217-3.
11. Golubitsky O. A study of optimal 4-bit reversible toffoli circuits and their synthesis / O. Golubitsky, D. Maslov // IEEE Transactions on Computers. – 2012. – Vol. 61, № 9. – P. 1341–1353. DOI: 10.1109/TC.2011.144.
12. Design of Boolean Functions from a great number of variables satisfying strict avalanche criterion / [E. G. Bardis, N. G. Bardis, A. P Markovski at al.] // Proceedings of the IEEE/WSES/IMACS : 3rd World multiconference on circuits, systems, communications and computers, Athens, July 1999: proceedings. – Athens: World Scientific, 1999. – P. 3111–3116.
13. Tang D. Construction of balanced Boolean functions with high nonlinearity and good autocorrelation properties / D. Tang, W. Zhang, X. Tang // Designs, Codes and Cryptography. – 2013. – Vol. 67, № 1. – P. 77–99 DOI: 10.1007/s10623-011-9587-9
14. A New 1D Chaotic Map and β-Hill Climbing for Generating Substitution-Boxes / [A. A. Alzaidi, M. Ahmad, M. N. Doja at al.] // IEEE Access. – 2018. – Vol. 6. – P. 55405 – 55418. DOI: 10.1109/ACCESS.2018.2871557
15. Lloyd S. Counting Functions Satisfying a Higher Order Strict Avalanche Criterion / S. Lloyd // EUROCRYPT ’89 : Workshop on the Theory and Application of Cryptographic Techniques. Advances in Cryptology, 10–13 April 1989: proceedings. – Berlin, Heidelberg: Springer, 1989. – Vol. 434. – P. 63–67. DOI: 10.1007/3-540-46885-4_9
16. Bardis N. G. Combinatorial method for Boolean SAC functions designing / N. G. Bardis // WSEAS Transactions on Communications. – 2004. – Vol. 3, № 2. – P. 746–752.
17. Gupta Brij B. Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives / Brij B. Gupta, Dharma P. Agraval, Haoxiang Wang. – London, New York: CRC Press, Taylor & Francis Group, Boca Raton, 2019. – 665 p. ISBN 9780815371335.
18. Dey S. Cryptanalysis of 4-Bit Crypto S-Boxes in Smart Applications / S. Dey, R. Ghosh // Security in Smart Cities: Models, Applications, and Challenges. Lecture Notes in Intelligent Transportation and Infrastructure. – Cham: Springer, 2019. – P. 211–253. ISBN 978-3-030-01560-2. DOI: 10.1007/978-3-030-01560-2_10.
19. Woods S. Efficient solution of systems of Boolean equations / S. Woods, G. Casinovi // 96 Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design, 10–14 November 1996: proceeding. – San Jose, California : ACM Press, 1996. – P. 542–546.
20. Rudeanu S. Boolean sets and most general solutions of Boolean equations / S. Rudeanu // Information Sciences. – 2010. – Vol. 180, № 12. – P. 2440–2447. DOI: 10.1016/j.ins.2010.01.029.
21. Baneres D. A Recursive Paradigm to Solve Boolean Relations / D. Baneres, J. Cortadella, M. Kishinevsky // IEEE Transactions on Computers. – 2009. – Vol. 58(4). – P. 512–527.http://doi.ieeecomputersociety.org/10.1109/TC.2008.165
22. Rushdi Ali M. A. Using variable-entered Karnaugh maps to produce compact parametric general solutions of Boolean equations / Ali M. A. Rushdi, Motaz H. Amashah // International Journal of Computer Mathematics. – 2011. – Vol. 88, № 15. – P. 3136–3149. DOI: 10.1080/00207160.2011.594505.
23. Rushdi A. M. A. The Inverse Problem for Boolean Equations / A. M. A. Rushdi, H. M. Albarakati // Journal of Computer Science. – 2012. – Vol. 8, № 12. – P. 2098–2105. DOI: 10.3844/jcssp.2012.2098.2105
24. Bibilo P.N. Decomposition of Boolean functions based on the solution of logic equations. I. II. III. / P. N. Bibilo // Izvestiya Rossijskoj akademii nauk. Teoriya i sistemy upravleniya. – 2002. – No. 4. – P. 53–64; 2002. – No.5. – P. 57–63.; 2003. – No. 6. – P. 88–97.
25. Rudeanu S. On the Decomposition of Boolean Functions via Boolean Equations / S. Rudeanu // Journal of Universal Computer Science. – 2004. – Vol. 10, № 9. – P. 1294–1301.
26. Primenko É. A. Equivalence classes of invertible Boolean functions / É. A. Primenko // Cybernetics. – 1984. – Vol. 20, № 6. – P. 771–776. DOI: 10.1007/BF01072161
27. Soeken M. Embedding of Large Boolean Functions for Reversible Logic / [M. Soeken, R. Wille, O. Keszocze et al.] // Journal on Emerging Technologies in Computing Systems. – 2016. – Vol. 12, № 4, Article No. 41. – P. 41:1–41:26. DOI: 10.1145/2786982.
28. Soeken M. Enumeration of Reversible Functions and Its Application to Circuit Complexity / M. Soeken, N. Abdessaied, G. De Micheli // Proceedings of the 8th Conference on Reversible Computation (RC 2016), 7–8 July 2016: proceedings. – Bologna: Cham, Springer, 2016. – Vol. 9720. – P. 255–270. ISBN: 978-3-319-40578-0. DOI: 10.1007/978-3-319-40578-0 19.
29. Kavut S. Construction and search of balanced Boolean functions on even number of variables towards excellent autocorrelation profile / S. Kavut S. Maitra, D. Tang // Designs, Codes and Cryptography. – 2019. – Vol. 87, № 2–3. – P. 261–276. DOI: 10.1007/s10623-018-0522-1.
30. Lorens C. S. Invertible Boolean functions / C. S Lorens // IEEE Transactions on Electronic Computers – 1964 – Vol. EC–13, № 5. – P. 529–541. DOI:10.1109/pgec.1964.263724.
31. Varadharajan V. Public key cryptosystems based on boolean permutations and their applications / V. Varadharajan, C.-K. Wu // International Journal of Computer Mathematics. – 2000. – Vol. 74, № 2. – P. 167–184. DOI: 10.1080/00207160008804932.

DOI 10.15588/1607-3274-2019-4-19