Constructing Symmetric Operations of Cryptographic Information Encoding

Volodymyr Rudnytskyi[1], Vira Babenko[1], Nataliia Lada[1], Yaroslav Tarasenko[1], and Yuliia Rudnytska[1]

[1] *Cherkasy State Technological University, Shevchenko ave., 460, Cherkassy, 18006, Ukraine*

Abstract

This article was addressed to the problem of improving the quality of low-resource systems for cryptographic information transformation. The existing contradictions between the limits on software and hardware implementation and the strength of the cryptographic algorithms can be partially eliminated by applying a set of groups of symmetric operations of cryptographic encoding. Unfortunately, to date, the results of studying symmetric two-bit two-operand operations of cryptographic encoding have a limited non-systemic nature. The lack of the unified generalized method for synthesizing groups of symmetric multi-bit two-operand operations of cryptographic encoding makes it impossible to use the full potential of the practical application of these operations. Thus, the aim of this article is to create a unified method for synthesizing groups of symmetric multi-bit two-operand operations of cryptographic encoding. Achieving this aim will make it possible to significantly increase both the variability of lightweight low-resource cryptographic algorithms and the cryptographic strength directly related to it. A new concept of synthesizing groups of operations is proposed, which will make it possible to address the main shortcoming of the previous concept in which each method is based on its splitting into suboperands. Addressing this shortcoming will also make it possible to synthesize both symmetric two-bit two-operand operations of cryptographic encoding and symmetric two-operand operations of arbitrary bitness. Having been applied a new concept, it was synthesized a new previously unknown group of symmetric two-bit two-operand operations up to permutation. A symmetric group of matrix three-bit two-operand operations of cryptographic encoding was synthesized. The developed method for synthesizing symmetric multi-bit two-operand operations of cryptographic encoding, belonging to different mathematical groups, provides an increase in the variability and strength of cryptographic algorithms.

Keywords

information security, cryptography, information encoding, operations of cryptographic encoding, synthesis of operations of cryptographic encoding

1. The Relevance of the Research

Lightweight and low-resource cryptography are designed to implement cryptographic algorithms on devices with limited technical resources. The relevance of this direction and significant interest in its development are directly related to the expansion of the scope of using the cryptographic information protection [1–5]. Despite the fact that the limits on the hardware and software cases of implementing the cryptographic algorithms are different, in practice they are interrelated and most of them are overcome by the same means of addressing them.

The vast majority of recently developed lightweight ciphers are symmetric block ciphers [6–10]. As a rule, when constructing low-resource ciphers, simplifications of well-known cryptographic

CPITS-II-2021: Cybersecurity Providing in Information and Telecommunication Systems, October 26, 2021, Kyiv, Ukraine
EMAIL: rvn_2008@ukr.net (V. Rudnytskyi); verababenko84@gmail.com (V. Babenko); ladanatali256@gmail.com (N. Lada); yaroslav.tarasenko93@gmail.com (Y. Tarasenko); y.v.rudnitskaya@gmail.com (Y. Rudnytska)
ORCID: 0000-0003-3473-7433 (V. Rudnytskyi); 0000-0003-2039-2841 (V. Babenko); 0000-0002-7682-2970 (N. Lada); 0000-0002-5902-8628 (Y. Tarasenko); 0000-0001-6384-0523 (Y. Rudnytska)

algorithms are used [11–17]. These simplifications are associated with: decreasing the size of encryption blocks, decreasing the size of keys, simplifying the key schedule and encryption rounds [18]. As it can be seen, all the above simplifications are directly related to the decrease in the cryptographic strength of the algorithms.

One of the ways to partially overcome the conflict between the simplification of the cryptographic algorithm and its complexity can be considered increasing the algorithm's variability through the use of cryptographic information coding operations.

2. Overview of the Work-Related Publications

Symmetric two-operand operations of cryptographic encoding can be used to expand the variability of symmetric block ciphers. These operations can change both within the encryption round and when its shift.

In essence, a two-operand operation of cryptographic encoding is a model of an interconnected group of lookup table sets for the first operand. Determining the lookup table that implements the transformation is defined by the value of the second operand [19]. Asymmetric operations of cryptographic encoding do not allow swapping of operand values. In addition, asymmetric operations can only be applied in interconnected pairs for direct and inverse transformations. Symmetric operations of cryptographic encoding allow permutation of the operands' values in places, and can be used for both direct and inverse transformation of information [19]. Based on the above, it can be concluded that symmetric operations of cryptographic encoding will be more preferable for use in lightweight ciphers in comparison with asymmetric operations.

Despite the great practical possibilities for application in lightweight cryptography, symmetric two-operand operations of cryptographic encoding have almost not been studied. There are only partial studies' results of symmetric two-bit two-operand operations of cryptographic encoding.

In [19], a group of symmetric two-bit two-operand operations of cryptographic encoding, synthesized on the basis of bitwise modulo two addition is described and investigated. Operations of this group are represented in the Table 1.

Table 1 represents two-bit two-operand operations of cryptographic encoding of the first group, $O_1^1 - O_{24}^1$ and their interrelation with one-operand operations (F_1, F_2, F_3 – basic operations, F_3, F_4, F_5 – operations of permutations, $\begin{bmatrix}0\\0\end{bmatrix}$, $\begin{bmatrix}0\\1\end{bmatrix}$, $\begin{bmatrix}1\\0\end{bmatrix}$, $\begin{bmatrix}1\\1\end{bmatrix}$ – operations of inversion). In the mathematical models of these operations, it is designated $x_1 - x_2$ values of the first operand's first and second bits, $k_1 - k_2$ values of the second operand's first and second bits.

The synthesis of the represented operations is based on using a group of one-operand two-bit operations of cryptographic encoding represented in the Table 2.

The method for synthesizing the first group of two-bit two-operand operations based on the operation of bitwise modulo two addition for symmetric stream cipher is as follows [19]:

- the operation of bitwise modulo two addition is being splitted into two suboperations of processing the first and the second operands

$$O_{\oplus 2} = O_{\oplus 2}^{1*} \oplus O_{\oplus 2}^{2*} = \begin{bmatrix} x_1 \oplus k_1 \\ x_2 \oplus k_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \quad (1)$$

where $O_{\oplus 2}$ is the operation of bitwise modulo two addition; $O_{\oplus 2}^{1*}$ and $O_{\oplus 2}^{2*}$ are the suboperations of processing the first and the second operands, respectively;

Table 1

Two-bit two-operand operations of cryptographic encoding, synthesized on the basis of bitwise modulo two addition (first group of operations)

Operation classifier		Operations of inversion	
		$\begin{bmatrix}0\\0\end{bmatrix}$ / $\begin{bmatrix}1\\0\end{bmatrix}$	$\begin{bmatrix}0\\1\end{bmatrix}$ / $\begin{bmatrix}1\\1\end{bmatrix}$
Basic operations	$F_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$	$O_1^1 = \begin{bmatrix} x_1 \oplus k_1 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_7^1 = \begin{bmatrix} x_1 \oplus k_1 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{13}^1 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_{19}^1 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_2 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix}$	$O_2^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_8^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{14}^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_{20}^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_3 = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix}$	$O_3^1 = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_9^1 = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{15}^1 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{21}^1 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
Operations of permutations	$F_4 = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$	$O_4^1 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{10}^1 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$
		$O_{16}^1 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{22}^1 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$
	$F_5 = \begin{bmatrix} x_2 \\ x_1 \oplus x_2 \end{bmatrix}$	$O_5^1 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{11}^1 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{17}^1 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{23}^1 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_6 = \begin{bmatrix} x_1 \oplus x_2 \\ x_1 \end{bmatrix}$	$O_5^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{12}^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$
		$O_{18}^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{24}^1 = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$

- a basic group of two-operand operations is being synthesized based on the transforming the suboperations of processing the first and the second operands by basic one-operand operations $F_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $F_2 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix}$, and $F_3 = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix}$ [19]:

$$O_1^1 = F_1^1(O_{\oplus 2}^{1*}) \oplus F_1^2(O_{\oplus 2}^{2*}) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} x_1 \oplus k_1 \\ x_2 \oplus k_2 \end{bmatrix}$$

$$O_2^1 = F_2^1(O_{\oplus 2}^{1*}) \oplus F_2^2(O_{\oplus 2}^{2*}) = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \oplus k_2 \\ k_2 \end{bmatrix} = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_2 \oplus k_2 \end{bmatrix}$$

$$O_3^1 = F_3^1(O_{\oplus 2}^{1*}) \oplus F_3^2(O_{\oplus 2}^{2*}) = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \\ k_1 \oplus k_2 \end{bmatrix} = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$$

where F_i^1 and F_i^2 – *i*-th one-operand operations of transforming the first and the second suboperations, respectively;

- performing the permutation of elementary functions on the two-bit two-operand operations of the base group [19];
- performing operations of inversion on synthesized operations [19].

Table 2

One-operand two-bit operations of cryptographic encoding

Operation classifier	Operations of inversion			
	$\begin{bmatrix}0\\0\end{bmatrix}$	$\begin{bmatrix}0\\1\end{bmatrix}$	$\begin{bmatrix}1\\0\end{bmatrix}$	$\begin{bmatrix}1\\1\end{bmatrix}$
Basic operations	$F_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$	$F_7 = \begin{bmatrix} x_1 \\ x_2 \oplus 1 \end{bmatrix}$	$F_{13} = \begin{bmatrix} x_1 \oplus 1 \\ x_2 \end{bmatrix}$	$F_{19} = \begin{bmatrix} x_1 \oplus 1 \\ x_2 \oplus 1 \end{bmatrix}$
	$F_2 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix}$	$F_8 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \oplus 1 \end{bmatrix}$	$F_{14} = \begin{bmatrix} x_1 \oplus x_2 \oplus 1 \\ x_2 \end{bmatrix}$	$F_{20} = \begin{bmatrix} x_1 \oplus x_2 \oplus 1 \\ x_2 \oplus 1 \end{bmatrix}$
	$F_3 = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix}$	$F_9 = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \oplus 1 \end{bmatrix}$	$F_{15} = \begin{bmatrix} x_1 \oplus 1 \\ x_1 \oplus x_2 \end{bmatrix}$	$F_{21} = \begin{bmatrix} x_1 \oplus 1 \\ x_1 \oplus x_2 \oplus 1 \end{bmatrix}$
Operations of permutations	$F_4 = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$	$F_{10} = \begin{bmatrix} x_2 \\ x_1 \oplus 1 \end{bmatrix}$	$F_{16} = \begin{bmatrix} x_2 \oplus 1 \\ x_1 \end{bmatrix}$	$F_{22} = \begin{bmatrix} x_2 \oplus 1 \\ x_1 \oplus 1 \end{bmatrix}$
	$F_5 = \begin{bmatrix} x_2 \\ x_1 \oplus x_2 \end{bmatrix}$	$F_{11} = \begin{bmatrix} x_2 \\ x_1 \oplus x_2 \oplus 1 \end{bmatrix}$	$F_{17} = \begin{bmatrix} x_2 \oplus 1 \\ x_1 \oplus x_2 \end{bmatrix}$	$F_{23} = \begin{bmatrix} x_2 \oplus 1 \\ x_1 \oplus x_2 \oplus 1 \end{bmatrix}$
	$F_6 = \begin{bmatrix} x_1 \oplus x_2 \\ x_1 \end{bmatrix}$	$F_{12} = \begin{bmatrix} x_1 \oplus x_2 \\ x_1 \oplus 1 \end{bmatrix}$	$F_{18} = \begin{bmatrix} x_1 \oplus x_2 \oplus 1 \\ x_1 \end{bmatrix}$	$F_{24} = \begin{bmatrix} x_1 \oplus x_2 \oplus 1 \\ x_1 \oplus 1 \end{bmatrix}$

Basing on the modulo four addition, the second group of symmetric two-bit two-operand operations of cryptographic encoding was synthesized [20]. This group of operations is represented in the Table 3.

Table 3

Two-bit two-operand operations of cryptographic encoding, synthesized on the basis of left-handed modulo four addition (second group of operations)

Operation classifier		Operations of inversion $\begin{bmatrix}0\\0\end{bmatrix}$ / $\begin{bmatrix}1\\0\end{bmatrix}$	Operations of inversion $\begin{bmatrix}0\\1\end{bmatrix}$ / $\begin{bmatrix}1\\1\end{bmatrix}$
Basic operations	$F_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$	$O_1^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_7^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{13}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_{19}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_2 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix}$	$O_2^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_8^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{14}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_2 \oplus k_2 \end{bmatrix}$	$O_{20}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_3 = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix}$	$O_3^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_9^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{15}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{21}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
Operations of permutations	$F_4 = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$	$O_4^2 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \end{bmatrix}$	$O_{10}^2 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \end{bmatrix}$
		$O_{16}^2 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \end{bmatrix}$	$O_{22}^2 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \end{bmatrix}$
	$F_5 = \begin{bmatrix} x_2 \\ x_1 \oplus x_2 \end{bmatrix}$	$O_5^2 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{11}^2 = \begin{bmatrix} x_2 \oplus k_2 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{17}^2 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{23}^2 = \begin{bmatrix} x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_6 = \begin{bmatrix} x_1 \oplus x_2 \\ x_1 \end{bmatrix}$	$O_6^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \end{bmatrix}$	$O_{12}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \end{bmatrix}$
		$O_{18}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \end{bmatrix}$	$O_{24}^2 = \begin{bmatrix} x_1 \oplus x_2 \cdot \bar{k}_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \oplus x_2 \cdot k_2 \oplus k_1 \oplus 1 \end{bmatrix}$

Synthesizing the second group of two-bit two-operand operations based on the modulo four addition (left-handed modulo four addition) for symmetric stream cipher is implemented similarly to synthesizing the first group of operations with the following differences [20]:

- the operation of left-handed modulo four addition is split into two suboperations of processing the first and the second operands

$$O_{\oplus 4\rightarrow} = O^{1*}_{\oplus 4\rightarrow} \oplus O^{2*}_{\oplus 4\rightarrow} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_2 \cdot k_2 \\ x_2 \oplus k_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \oplus x_2 \cdot k_2 \\ k_2 \end{bmatrix} \quad (2)$$

where $O_{\oplus 4\rightarrow}$ is the operation of left-handed modulo four addition; $O^{1*}_{\oplus 4\rightarrow}$ and $O^{2*}_{\oplus 4\rightarrow}$ are the suboperations of processing the first and the second operands, respectively;

- the synthesis of the basic two-operand operation through transforming the suboperation of processing the first and the second operands is implemented as [20]:

$$O_1^2 = F_1^1(O^{1*}_{\oplus 4\rightarrow}) \oplus F_1^2(O^{2*}_{\oplus 4\rightarrow}) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} x_2 \cdot k_2 \oplus k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_2 \cdot k_2 \\ x_2 \oplus k_2 \end{bmatrix};$$

$$O_2^2 = F_2^1(O^{1*}_{\oplus 4\rightarrow}) \oplus F_2^2(O^{2*}_{\oplus 4\rightarrow}) = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \oplus x_2 \cdot k_2 \oplus k_2 \\ k_2 \end{bmatrix} = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus x_2 \cdot k_2 \\ x_2 \oplus k_2 \end{bmatrix};$$

$$O_3^2 = F_3^1(O^{1*}_{\oplus 4\rightarrow}) \oplus F_3^2(O^{2*}_{\oplus 4\rightarrow}) = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \oplus x_2 \cdot k_2 \\ k_1 \oplus x_2 \cdot k_2 \oplus k_2 \end{bmatrix} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_2 \cdot k_2 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus x_2 \cdot k_2 \end{bmatrix}.$$

The processes of further synthesizing the operations of the first and the second groups of operations coincide.

Synthesizing the third group of symmetric two-bit two-operand operations of cryptographic encoding based on right-handed modulo four addition is considered in [21]. This group of operations is represented in the Table 4.

Synthesizing the third group of symmetric two-bit two-operand operations based on the operation of right-handed modulo four addition is implemented similarly to synthesizing the first and the second groups of operations [21]. The operation of right-handed modulo four addition is split into two suboperations of processing the first and the second operands

$$O_{\oplus 4\leftarrow} = O^{1*}_{\oplus 4\leftarrow} \oplus O^{2*}_{\oplus 4\leftarrow} = \begin{bmatrix} x_1 \oplus k_1 \\ x_2 \oplus k_2 \oplus x_1 \cdot k_1 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \\ k_2 \oplus x_1 \cdot k_1 \end{bmatrix} \quad (3)$$

where $O_{\oplus 4\rightarrow}$ is the operation of right-handed modulo four addition.

Synthesizing the basic group of symmetric two-bit two-operand operations through the operation of right-handed modulo four addition is implemented as [21]:

$$O_1^2 = F_1^1(O^{1*}_{\oplus 4\leftarrow}) \oplus F_1^2(O^{2*}_{\oplus 4\leftarrow}) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \\ k_2 \oplus x_1 \cdot k_1 \end{bmatrix} = \begin{bmatrix} x_1 \oplus k_1 \\ x_2 \oplus k_2 \oplus x_1 \cdot k_1 \end{bmatrix};$$

$$O_2^2 = F_2^1(O^{1*}_{\oplus 4\leftarrow}) \oplus F_2^2(O^{2*}_{\oplus 4\leftarrow}) = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \oplus k_2 \oplus x_1 \cdot k_1 \\ k_2 \oplus x_1 \cdot k_1 \end{bmatrix} = \begin{bmatrix} x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus x_1 \cdot k_1 \\ x_2 \oplus k_2 \oplus x_1 \cdot k_1 \end{bmatrix};$$

$$O_3^2 = F_3^1(O^{1*}_{\oplus 4\leftarrow}) \oplus F_3^2(O^{2*}_{\oplus 4\leftarrow}) = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix} \oplus \begin{bmatrix} k_1 \\ k_1 \oplus k_2 \oplus x_1 \cdot k_1 \end{bmatrix} = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus x_1 \cdot k_1 \end{bmatrix}.$$

The processes of further synthesizing the operations of all three groups of operations coincide.

Table 4

Two-bit two-operand operations of cryptographic encoding, synthesized on the basis of right-handed modulo four addition (third group of operations)

Operation classifier		Operations of inversion $\begin{bmatrix}0\\0\end{bmatrix}\ \begin{bmatrix}1\\0\end{bmatrix}$	Operations of inversion $\begin{bmatrix}0\\1\end{bmatrix}\ \begin{bmatrix}1\\1\end{bmatrix}$
Basic operations	$F_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$	$O_1^3 = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \end{bmatrix}$	$O_7^3 = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{13}^3 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \end{bmatrix}$	$O_{19}^3 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_2 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \end{bmatrix}$	$O_2^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \end{bmatrix}$	$O_8^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{14}^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \end{bmatrix}$	$O_{20}^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_3 = \begin{bmatrix} x_1 \\ x_1 \oplus x_2 \end{bmatrix}$	$O_3^3 = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_9^3 = \begin{bmatrix} x_1 \oplus k_1 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{15}^3 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{21}^3 = \begin{bmatrix} x_1 \oplus k_1 \oplus 1 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
Operations of permutations	$F_4 = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$	$O_4^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{10}^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$
		$O_{16}^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{22}^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$
	$F_5 = \begin{bmatrix} x_2 \\ x_1 \oplus x_2 \end{bmatrix}$	$O_5^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{11}^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
		$O_{17}^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \end{bmatrix}$	$O_{23}^3 = \begin{bmatrix} x_1 \cdot k_1 \oplus x_2 \oplus k_2 \oplus 1 \\ x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \end{bmatrix}$
	$F_6 = \begin{bmatrix} x_1 \oplus x_2 \\ x_1 \end{bmatrix}$	$O_6^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{12}^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$
		$O_{18}^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \end{bmatrix}$	$O_{24}^3 = \begin{bmatrix} x_1 \cdot \bar{k}_1 \oplus x_2 \oplus k_1 \oplus k_2 \oplus 1 \\ x_1 \oplus k_1 \oplus 1 \end{bmatrix}$

In [19-21], particular methods of synthesizing some groups of symmetric two-bit two-operand operations of cryptographic encoding are considered. These methods were developed taking into account the simplicity of their practical implementation. Currently, there is no unified method for synthesizing groups of symmetric multi-bit two-operand operations of cryptographic encoding. Therefore, the purpose of this study is to develop a to increase the variability of lightweight low-resource cryptographic algorithms.

3. The Method for Synthesizing Symmetric Two-Operand Operations of Cryptographic Encoding

The results of a computational experiment [22] indicate that there are 4 groups of symmetric two-bit two-operand operations of cryptographic encoding by 24 operation. The conducted researches have shown that the unexplored group of operations is operations up to the permutation accuracy of the O_i^4 values of the truth table of the operation O_1^4, which are represented in the Table 5.

Table 5

The truth table of the operations O_1^4 and O_i^4

The operation	O_1^4				O_i^4			
The operands' values	00	01	10	11	00	01	10	11
00	00	01	10	11	a	b	c	d
01	01	11	00	10	b	d	a	c
10	10	00	11	01	c	a	d	b
11	11	10	01	00	d	c	b	a

$a \neq b \neq c \neq d \in \{00; 01; 10; 11\}$, $i \in \{1; 2; ...; 24\}$

The operations' O_1^4 truth table, can be represented as:

$$O_1^4 = \begin{cases} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \ if \ k_1 = 0; \ k_2 = 0 \\ \begin{bmatrix} x_2 \\ x_1 \oplus 1 \end{bmatrix}, \ if \ k_1 = 0; \ k_2 = 1 \\ \begin{bmatrix} x_2 \oplus 1 \\ x_1 \end{bmatrix}, if \ k_1 = 1; \ k_2 = 0 \\ \begin{bmatrix} x_1 \oplus 1 \\ x_2 \oplus 1 \end{bmatrix}, if \ k_1 = 1; \ k_2 = 1 \end{cases} = \begin{bmatrix} x_1 \oplus k_1 \oplus (x_1 \oplus x_2) \cdot (k_1 \oplus k_2) \\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2) \cdot (k_1 \oplus k_2) \end{bmatrix} \quad (4)$$

It turned out to be difficult to apply the considered synthesis methods [19 - 21] to synthesize a group of operations based on the operation O_1^4. This is due to the fact that synthesizing the considered groups of operations is constructed on extracting the operation of the suboperations' groups for processing the first and the second operands:

$$O = \begin{bmatrix} f_1(x_1, k_1) \\ f_2(x_2, k_2) \end{bmatrix} = \begin{bmatrix} f_1^*(x_1) \\ f_2^*(x_2) \end{bmatrix} \oplus \begin{bmatrix} f_1^*(k_1) \\ f_2^*(, k_2) \end{bmatrix} = O^{1*} \oplus O^{2*} \quad (5)$$

where f_1, f_2 are the elementary functions for obtaining the first and the second bits of the result, respectively; f_1^*, f_2^* elementary functions for processing the corresponding bit of the suboperand.

The operation O_1^1 (1) is correctly decomposed into suboperands in accordance with (5). Operations O_1^2 (2) and O_1^3 (3) are conditionally decomposed into suboperands, since a complete decomposition in accordance with (5) has not been obtained. Decomposition of the operation O_1^4 (5) into suboperands, causes even more difficulties. At the same time, it should be taken into account that each method is based on its decomposition into suboperands. To address the noted shortcomings, it is necessary to change the synthesis concept of the operations' groups:

$$\text{If } O = \begin{bmatrix} f_1(x, k) \\ f_2(x, k) \end{bmatrix} \text{ then } O_i = F_i(O) = F_i\begin{pmatrix} f_1(x, k) \\ f_2(x, k) \end{pmatrix}. \quad (6)$$

To reduce the synthesis complexity, it is advisable to synthesize only the operations of the base group through this concept, and the rest to obtain by permutations and inversions, which is implemented in the prototype methods.

By applying the new concept, a group of operations up to the permutation accuracy is being synthesized, on the basis of the operation O_1^4. The synthesis results are represented in the Table 6.

Table 6

Two-bit two-operand operations of cryptographic encoding, synthesized basing on the operation O_1^4 (fourth group of operations)

Operation classifier		Operations of inversion $\begin{bmatrix}0\\0\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix}$	Operations of inversion $\begin{bmatrix}0\\1\end{bmatrix}\begin{bmatrix}1\\1\end{bmatrix}$
Basic operations	$F_1 = \begin{bmatrix}x_1\\x_2\end{bmatrix}$	$O_1^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_7^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
		$O_{13}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_{19}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
	$F_2 = \begin{bmatrix}x_1 \oplus x_2\\x_2\end{bmatrix}$	$O_2^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_8^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
		$O_{14}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_{20}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
	$F_3 = \begin{bmatrix}x_1\\x_1 \oplus x_2\end{bmatrix}$	$O_3^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2\end{bmatrix}$	$O_9^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\end{bmatrix}$
		$O_{15}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2\end{bmatrix}$	$O_{21}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\end{bmatrix}$
Operations of permutations	$F_4 = \begin{bmatrix}x_2\\x_1\end{bmatrix}$	$O_4^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_{10}^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
		$O_{16}^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_{22}^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
	$F_5 = \begin{bmatrix}x_2\\x_1 \oplus x_2\end{bmatrix}$	$O_5^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2\end{bmatrix}$	$O_{11}^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\end{bmatrix}$
		$O_{17}^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2\end{bmatrix}$	$O_{23}^4 = \begin{bmatrix}x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\end{bmatrix}$
	$F_6 = \begin{bmatrix}x_1 \oplus x_2\\x_1\end{bmatrix}$	$O_6^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_{12}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$
		$O_{18}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\end{bmatrix}$	$O_{24}^4 = \begin{bmatrix}x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus 1\\ x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2)\oplus 1\end{bmatrix}$

The obtained synthesis results coincided with the results of the computational experiment, in which the truth tables of symmetric two-bit two-operand operations of cryptographic encoding were modeled and sorted [22].

The proposed concept makes it possible synthesizing symmetric two-operand operations of arbitrary bitness, besides the symmetric two-bit two-operand operations of encoding. To do this, in concept (6), it is necessary to expand the number of bits:

$$\text{If } O = \begin{bmatrix} f_1(x_1, x_2, .., x_n, k_1, k_2, .., k_n) \\ f_2(x_1, x_2, .., x_n, k_1, k_2, .., k_n) \\ . \ . \ . \ . \\ f_n(x_1, x_2, .., x_n, k_1, k_2, .., k_n) \end{bmatrix} \text{ then } O_i = F_i(O) = \begin{pmatrix} f_1(x_1, x_2, .., x_n, k_1, k_2, .., k_n) \\ f_2(x_1, x_2, .., x_n, k_1, k_2, .., k_n) \\ . \ . \ . \ . \\ f_n(x_1, x_2, .., x_n, k_1, k_2, .., k_n) \end{pmatrix}. \quad (7)$$

Based on this suggestion (7), the algorithm of the method for synthesizing symmetric two-operand operations of cryptographic encoding can be represented as follows:

1. Based on a complete iterating over n-bit one-operand operations of the base group, applying the concept (7), it is synthesized symmetric two-operand operations of cryptographic encoding of the base group;
2. Having performed the operations of elementary functions permutation over the two-operand operations of the base group, it will be obtained the extended group of symmetric operations;
3. Having performed the operations of inverting the preliminary transformation results over the two-operand operations of the extended group, it will be obtained the complete group of symmetric two-operand operations of a given bitness.

On the example of synthesizing symmetric two-bit two-operand operations, the proposed method provides the synthesis of a complete group of operations (24 operations) based on an arbitrary operation from this group.

4. Implementation of the Method for Synthesizing Symmetric Two-Operand Operations of Cryptographic Information Encoding

Consider the synthesis of symmetric three-bit two-operand operations of cryptographic encoding.

The number of one-operand operations of cryptographic encoding is being determined [23].

$$K_o^1(n) = 2^n! \quad (7)$$

$$K_o^1(n) = K_{об}(n) \cdot K_{оп}(n) \cdot K_{ou}(n) = K_{об}(n) \cdot n! \cdot 2^n \quad (8)$$

where n is the operation's bitness, $K_{об}(n)$, $K_{оп}(n) = n!$, $K_{ou}(n) = 2^n$ are the number of basic operations, operations of permutations and operations of inversion, respectively.

Based on the expressions (7) and (8), the number of two-bit one-operand operations of cryptographic encoding is determined as:

$K_o^1(2) = 4! = 24$; $K_o^1(2) = K_{об}(2) \cdot 2! \cdot 2^2 = 3 \cdot 6 \cdot 4 = 24$.

Since, according to the results of the experiment, there are 96 symmetric two-bit two-operand operations, and they make up 4 groups of 24 operations, it can be assumed that $K_o^2(2) = 96 = 4 \cdot 2^2!$. Consequently:

$$K_o^2(n) = k \cdot 2^n! \quad (9)$$

where k is the groups' number of symmetric n-bit two-operand operations of cryptographic encoding.

The number of operations in each group of symmetric three-bit two-operand operations in accordance with (9) is determined: $K_o^2(3) = k \cdot 2^3! = k \cdot 8! = k \cdot 40320$ and is 40320 operations.

In practice, at the time, it is not possible to synthesize a group from such a number of operations. This is due to the lack of the unified mathematical apparatus that makes it possible to simulate the entire set of three-bit one-operand operations [23]. Therefore, in the process of synthesizing symmetric three-bit two-operand operations (Table 7), there will be a limitation only to synthesizing basic two-operand operations on the basis of the matrix single-operand operations.

In accordance with [23], the number of basic three-bit one-operand matrix operations is 28.

It is synthesized a basic group of symmetric three-bit two-operand matrix operations of cryptographic encoding based on the operation $O_1 = \begin{bmatrix} x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_3 \oplus k_3 \end{bmatrix}$, applying the proposed concept (7). The synthesis results are represented in the table 5.

Application of the proposed method for synthesizing symmetric two-operand operations of cryptographic information encoding will provide, in accordance with (8), constructing groups of 1344 matrix operations ($K_o^{2^*}(n^*) = 28 \cdot 3! \cdot 2^3 = 28 \cdot 6 \cdot 8 = 1344$).

To check the correctness of the synthesis results of the obtained symmetric matrix operations, it was applied the requirements for the symmetry of the operations given by the truth tables [22]. Additionally, each operation was checked on the basis of a complete iteration of all input data.

Table 7

The basic group of symmetric three-bit two-operand matrix operations of cryptographic encoding (one of the cases)

Basic operations	
$F_1 = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$	$O_1 = \begin{bmatrix} x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_3 \oplus k_3 \end{bmatrix}$
$F_2 = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \\ x_3 \end{bmatrix}$	$O_2 = \begin{bmatrix} x_1 \oplus k_1 \oplus x_2 \oplus k_2 \\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_3 \oplus k_3 \end{bmatrix}$
$F_3 = \begin{pmatrix} x_1 \\ x_1 \oplus x_2 \\ x_3 \end{pmatrix}$	$O_3 = \begin{bmatrix} x_1 \oplus k_1 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \\ x_3 \oplus k_3 \end{bmatrix}$
$F_4 = \begin{bmatrix} x_1 \oplus x_3 \\ x_2 \\ x_3 \end{bmatrix}$	$O_4 = \begin{bmatrix} x_1 \oplus k_1 \oplus x_3 \oplus k_3 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_2 \oplus k_2 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_3 \oplus k_3 \end{bmatrix}$
. . . . .	
$F_{25} = \begin{bmatrix} x_1 \oplus x_3 \\ x_1 \oplus x_2 \\ x_3 \end{bmatrix}$	$O_{25} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_3 \oplus k_3 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \\ x_3 \oplus k_3 \end{bmatrix}$
$F_{26} = \begin{bmatrix} x_1 \oplus x_3 \\ x_1 \oplus x_2 \\ x_1 \oplus x_2 \oplus x_3 \end{bmatrix}$	$O_{26} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_3 \oplus k_3 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus x_3 \oplus k_3 \end{bmatrix}$
$F_{27} = \begin{bmatrix} x_1 \oplus x_3 \\ x_2 \oplus x_3 \\ x_1 \oplus x_2 \oplus x_3 \end{bmatrix}$	$O_{27} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_3 \oplus k_3 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_2 \oplus k_2 \oplus x_3 \oplus k_3 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus x_3 \oplus k_3 \end{bmatrix}$
$F_{28} = \begin{bmatrix} x_1 \oplus x_2 \\ x_2 \oplus x_3 \\ x_1 \oplus x_2 \oplus x_3 \end{bmatrix}$	$O_{28} = \begin{bmatrix} x_1 \oplus k_1 \oplus x_2 \oplus k_2 \\ x_2 \oplus k_2 \oplus x_3 \oplus k_3 \oplus (x_1 \oplus x_2)\cdot(k_1 \oplus k_2) \\ x_1 \oplus k_1 \oplus x_2 \oplus k_2 \oplus x_3 \oplus k_3 \end{bmatrix}$

The developed method for synthesizing symmetric two-operand operations of cryptographic information encoding provides an opportunity of increasing the variability of lightweight cryptographic algorithms. In addition, synthesizing symmetric operations of cryptographic encoding which belong to different mathematical groups increases the cryptographic strength of the algorithm. The application of two-operand operations of cryptographic encoding, to which the synthesized

operations are related, leads to a slight increase in the complexity associated with the implementation of the operations' synthesis both at the hardware and software levels [24–27].

5. Conclusions

1. For improving the quality of low-resource cryptographic systems, it was proposed to apply groups of symmetric operations of cryptographic information encoding.
2. For combination of the existing and new results of studying groups of symmetric operations of cryptographic encoding, a new concept for synthesizing operations was proposed.
3. A new method for synthesizing groups of symmetric multi-bit two-operand operations of cryptographic encoding has been developed to increase the variability of lightweight low-resource cryptographic algorithms.
4. Having been applied this method, a new, previously unknown group of symmetric two-bit two-operand operations of cryptographic encoding has been synthesized. For the first time, a symmetric group of matrix three-bit two-operand operations of cryptographic encoding has been synthesized.
5. The obtained results of operations' synthesis coincided with the results of a computational experiment by simulation of the obtained operations.
6. The application of two-operand operations of cryptographic encoding, synthesized on the basis of this method, leads to a slight increase in the implementation complexity of the cryptographic algorithm both at the hardware and software levels.
7. The proposed concept and the developed method for synthesizing symmetric multi-bit two-operand operations of cryptographic encoding provide the construction of operations belonging to different mathematical groups. Applying operations from different mathematical groups provides an increase in both the variability and the encryption strength.

6. References

[1] Khaled Salah Mohamed. New Frontiers in Cryptography. Quantum, Blockchain, Lightweight, Chaotic and DNA. Springer International Publishing. Springer, Cham. P. 104, (2020). DOI: https://doi.org/10.1007/978-3-030-58996-7

[2] Máté Horváth, Levente Buttyán. Cryptographic Obfuscation. A Survey. SpringerBriefs in Computer Science. Springer International Publishing. Springer, Cham. P. 107, (2020). DOI: https://doi.org/10.1007/978-3-319-98041-6

[3] Dunkelman, Orr, Jacobson, Jr., Michael J., O'Flynn, Colin (Eds.). Selected Areas in Cryptography. 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, Revised Selected Papers, Springer International Publishing. Springer, Cham. P.722, (2020). DOI: https://doi.org/10.1007/978-3-030-81652-0

[4] Bernstein, Daniel J., Buchmann, Johannes, Dahmen, Erik (Eds.) Post-Quantum Cryptography. Springer-Verlag Berlin Heidelberg. Springer, Berlin, Heidelberg. P. 246, (2009). DOI: https://doi.org/10.1007/978-3-540-88702-7

[5] Shang, Tao, Liu, Jianwei. Secure Quantum Network Coding Theory. Springer, Singapore. P. 283, (2020). DOI: https://doi.org/10.1007/978-981-15-3386-0

[6] Alex Biryukov and Leo Perrin. State of the art in lightweight symmetric cryptography. Cryptology ePrint Archive, Report 2017/511, (2017). http://eprint.iacr.org/2017/511.

[7] Engels, D., Fan, X., Gong, G., Hu, H., Smith, E.M.: Hummingbird: ultra-lightweight cryptography for resource-constrained devices. In: Financial Cryptography and Data Security—FC 2010, Springer, LNCS, 6054, pp. 3–18 (2010). DOI: https://doi.org/10.1007/978-3-642-14992-4_2

[8] George Hatzivasilis, Konstantinos Fysarakis, Ioannis. Papaefstathiou, and Charalampos Manifavas. A review of lightweight block ciphers. J. Cryptographic Engineering, 8(2):141–184,(2018). DOI: https://doi.org/10.1007/s13389-017-0160-y

[9] Charalampos Manifavas, George Hatzivasilis, Konstantinos Fysarakis, and Yannis Papaefstathiou. A survey of lightweight stream ciphers for embedded systems. Security and Communication Networks, 9(10):1226–1246, (2016). DOI: https://doi.org/10.1002/sec.1399
[10] Liu, Leibo, Wang, Bo, Wei, Shaojun. Reconfigurable Cryptographic Processor. Springer Singapore. P. 386, (2018). DOI: https://doi.org/10.1007/978-981-10-8899-5
[11] Manifavas, C., Hatzivasilis, G., Fysarakis, K., Rantos, K.: Lightweight cryptography for embedded systems a comparative analysis. In: 6th International Workshop on Autonomous and Spontaneous Security SETOP 2012, Springer, LNCS, 8247, pp. 333–349, (2012). DOI: https://doi.org/10.1007/978-3-642-54568-9_21
[12] Thomas Xuan Meng, W. Buchanan. Lightweight Cryptographic Algorithms on Resource-Constrained Devices. Preprints. (2020), DOI: https://doi.org/10.20944/ PREPRINTS202009.0302.V1
[13] Gildas Avoine, Julio Hernandez-Castro. Security of Ubiquitous Computing Systems. Selected Topics. Springer. 265 p. (2021). DOI: https://doi.org/10.1007/978-3-030-10591-4
[14] Aumasson, J.-P., Henzen, L., Meier, W., and Naya-Plasencia, M., Quark: A Lightweight Hash, Journal of Cryptology, Vol. 26, (2), pp. 313-339, (2013), DOI: https://doi.org/10.1007/s00145-012-9125-6
[15] ISO, ISO/IEC 29192-1:2012, Information Technology - Security Techniques - Lightweight Cryptography - Part 1: General, (2012), http://www.iso.org/iso/home/store/catalogue_tc/ catalogue_detail.htm?csnumber=56425
[16] ISO, ISO/IEC 29192-2:2012, Information Technology - Security Techniques - Lightweight Cryptography - Part 2: Block Ciphers, (2012), https://doi.org/10.6028/NIST.IR.8114
[17] ISO, ISO/IEC 29192-3:2012, Information Technology - Security Techniques - Lightweight Cryptography - Part 3: Stream Ciphers, (2012) http://www.iso.org/iso/home/store/catalogue_tc/ catalogue_detail.htm?csnumber=56426
[18] Anthony Journault, François-Xavier Standaert, and Kerem Varici. Improving the security and efficiency of block ciphers based on ls-designs. Des. Codes Cryptography, 82(1–2):495–509. (2017). https://link.springer.com/article/10.1007/s10623-016-0193-8
[19] N.V. Lada, S.G. Kozlovskaya, S.V. Rudnytsky, Construction of a mathematical group of symmetric operations based on addition modulo two. Modern Special Equipment Journal, 4(59) (2019) 33-41. http://suchasnaspetstehnika.com/journal/ukr/2019_4/6.pdf
[20] N.V. Lada, S.G. Kozlovskaya, Y.V. Rudnytska, Investigation and synthesis of a group of symmetric modified addition operations modulo four. Central Ukrainian Scientific Bulletin, 2(33) (2019) 181–189. DOI: https://doi.org/10.32515/2664-262X.2019.2(33).181-189
[21] N.V. Lada, S.V. Rudnytsky, V.M. Zaghoma, Y.V. Rudnytska, Investigation and synthesis of a group of symmetric modified operations of right-hand addition modulo four. Control, Navigation and Communication Systems Journal, 1(59) (2020) 93-96. DOI: https://doi.org/10.26906/SUNZ.2020.1.093
[22] V. N. Rudnitsky, V. Ya. Milchevich (Eds), Cryptographic coding, Kharkiv, 2014.
[23] V.N. Rudnitsky, V.Ya. Milchevich, V.G. Babenko, R.P. Melnik, S.V. Rudnitsky, O.G. Melnik, Cryptographic coding: methods and means of implementation, part 2, Kharkiv, 2014.
[24] V. Rudnytskyi, I. Opirskyy, O. Melnyk, M. Pustovit The implementation of strict stable cryptographic coding operations Modern Information Systems Journal, T.3, Vol.4 (2019), pp. 109-114. DOI: https://doi.org/10.20998/2522-9052.2019.3.15.
[25] A. Bessalov, et al., Analysis of 2-isogeny properties of generalized form Edwards curves, in: Proceedings of the Workshop on Cybersecurity Providing in Information and Telecommunication Systems, July 7, 2020, vol. 2746, pp. 1–13.
[26] Bessalov, A., Sokolov, V., Skladannyi, P., Zhyltsov, O. Computing of odd degree isogenies on supersingular twisted edwards curves. in: Proceedings of the Workshop on Cybersecurity Providing in Information and Telecommunication Systems, 2021, vol. 2923, pp. 1–11.
[27] A. Bessalov, V. Sokolov, P. Skladannyi, Modeling of 3- and 5-isogenies of supersingular Edwards curves, in: Proceedings of the 2nd International Workshop on Modern Machine Learning Technologies and Data Science, June 2–3, 2020, no. I, vol. 2631, pp. 30–39.