Same Bit-Size Moduli Formation of Residue Number System for Application in Asymmetric Cryptography

Mykhailo Kasianchuk[a], Ihor Yakymenko[a], Vasyl Yatskiv[a], Stepan Ivasiev[a] and Andriy Sverstiuk[b]

[a] *West Ukrainian National University, 11 Lvivska str., Ternopil, 46009, Ukraine*
[b] *I. Gorbachevsky Ternopil National Medical University, 1 Maidan Voli, Ternopil, 46001,Ukraine*

Abstract

This paper presents three methods (factorization, exhaustive search and replacement) of four equal bit size moduli sets formation in a residue number system modified perfect form. This allows using the bit grid registers more efficiently. Such problem is relevant for asymmetric cryptography and noise-protected coding algorithms. The theoretical bases of residue number system, its perfect and modified perfect forms are considered, their advantages and disadvantages are defined. It is shown that the most commonly used moduli in the form of power of two, Mersenne numbers and Fermat numbers require searching for the inverse element and multiplying by it, which makes it difficult to recover a decimal number from its residues using the Chinese remainder theorem. A modified perfect form of the residue number system simplifies this procedure. The graphical dependency of the fourth modulo on two prior ones with one known modulo is presented. Different bit sizes of moduli sets are considered. It is shown that in sets of four modulo with the same bit size in a modified perfect form of residue number system, the first and fourth moduli are negative, second and third are positive.

Keywords

Residue number system, modified perfect form, computing range, modulo system, speed, asymmetric cryptography.

1. Introduction

Nowadays the rapid progress in the information technology area, in particular in mission-critical applications, leads to the new demands for improved reliability, performance and productivity of various computing systems [1]. Modern requirements for reliability of information transmission using technical means lead to a decrease in productivity or increase in time and computational complexity and, accordingly, to increased energy consumption [2]. On the other hand, economic factors and development level of the technical means also impose appropriate restrictions. Therefore, it is advised to use special code systems that do not have such restrictions to solve these problems [3]. However, existing approaches and methods for data transmission and processing, that operate in positional numeral system (PNS), can not achieve increased requirements data processing reliability without reducing the performance of computing system with limited hardware and economic resources [4, 5]. It's caused by such disadvantages of PNS as high digit capacity, strictly consistent structure and the presence of inter-bit transfers, that complicate the architecture of computing systems and reduce their speed [6].

IntelITSIS’2021: 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security, March 24–26, 2021, Khmelnytskyi, Ukraine
EMAIL: kasyanchuk@ukr.net (M. Kasianchuk); iyakymenko@ukr.net (I. Yakymenko); jazkiv@ukr.net (V. Yatskiv); stepan.ivasiev@gmail.com (S. Ivasiev); sverstyuk@tdmu.edu.ua (A. Sverstiuk);
ORCID: 0000-0002-4469-8055 (M. Kasianchuk); 0000-0003-3446-1596 (I. Yakymenko); 0000-0001-9778-6625 (V. Yatskiv); 0000-0003-2243-5956 (S. Ivasiev); 0000-0001-8644-0776 (A. Sverstiuk)

CEUR Workshop Proceedings (CEUR-WS.org)

The most relevant task is the processing speed improvement for large amounts of numerical data in asymmetric cryptography [7] and noise-protected coding problems during data transmission [8-10]. One of the possible ways of solving it is to use non-positional number systems, in particular, the residue number system (RNS). Data processing and transmission in RNS has a number of advantages due to independence, lack of inter-bit transfers, low bit size and equality of residues, as well as possibility of parallel arithmetic operations execution. However, currently RNS is used only for solving some specialized problems [11-16], due to required conversion of binary code, which is used by universal computers and data processing devices, into RNS code, which allows to parallelize computational processes [17, 18]. In addition, RNS has a number of disadvantages that have slowed down its development, in particular, difficulties in performing division and numbers comparison operations [19]. But since the main operations in asymmetric cryptography are multiplication and exponentiation, the use of RNS becomes a very effective tool for processing multi-bit numbers [20] in asymmetric encryption of information flows. Furthermore there are effective correction codes developed for RNS, that are able to detect and correct error packages [21].

2. Theoretical basics of RNS and its usage in asymmetric cryptosystems

Let us consider positive pairwise coprime integers $p_1, p_2,..., p_z$, which are called bases or system modulo (z-moduli count). Lets define $P = \prod_{i=1}^{z} p_i$. This value represents the range of numbers in the selected moduli system. RNS is a non-positional number system in which the nonnegative integer N can be presented as a set of nonnegative residues from division of this number on the chosen bases of the system, such as

$$b_i = N \bmod p_i. \tag{1}$$

The RNS usage in computational algorithms is possible due to the presence of a certain isomorphism between mathematical operations on integers and the corresponding operations on the system of nonnegative integers over individual moduli. Moreover, the addition, multiplication and exponentiation of any nonnegative integers are completely identical to the corresponding operations performed on the residues system [18].

The reverse conversion into a positional number system is usually based on the Chinese remainder theorem [22]:

$$N = \left(\sum_{i=1}^{z} b_i M_i m_i \right) \bmod P\,, \tag{2}$$

where $P = \prod_{i=1}^{z} p_i$ (inequality should be satisfied $N<P$), $M_i = \frac{P}{p_i}$, to find m_i multiplicative inverse element should be calculated:

$$m_i = M_i^{-1} \bmod p_i\,. \tag{3}$$

Obviously, one of the ways to increase the speed of computers that use RNS is the choice of specialized moduli sets, on which significantly depends the execution time of both modular and non-modular operations. Therefore, for each specific application with the specified arithmetic operations, hardware components and constraints, it is necessary to select the appropriate set of moduli. For example, digital signal processing requires fewer modules than cryptography.

In the vast majority of works moduli, such as 2^k, $2^k \pm 1$, are considered, which allows using of bit grid registers effectively [23-25]. The worst modulo, which has the greatest execution complexity (it can be the largest one or the complex type modulo), defines the general parameter of the direct converter or the arithmetic channel.

Additionally, RNS offers many different moduli sets of different types and different quantities for certain applications, which significantly affect all parts of the hardware implementation, including direct converters, modular arithmetic channels, inverse converters. In particular, [26-27] presents safe and effective approaches for RNS usage on elliptic curves in cryptography. They are especially effective as a response to attacks on the side channel of the source and for protection when the malfunctions are injected in the computer system. In [28, 29], efficient algorithms for the RSA-cryptographic system implementation based on RNS were developed, and the experimental studies showed that they have greater speed and better resistance to brute force attacks compared to the classical ones. In paper [30] the methods for fast execution of arithmetic operations such as addition, multiplication and exponentiation in the modular number system with the implementation of cryptographic transformations were developed, which show a significant reduction in the execution time of the crypto algorithms basic operations.

But all the considered approaches require calculation of the inverse element (3) and multiplying it by (2), which significantly reduces the speed while recovering decimal number from the RNS [31-33]. This is, along with the difficulties in dividing and comparing numbers, the main drawback that has slowed down the development of RNS. One of the options to simplify this procedure is to use different forms of RNS. In particular, in [34] methods for perfect form (PF) of RNS formation, where modules system p_i is selected so that the values of all coefficients m_i=1, are developed and investigated. This approach eliminates the complicated procedure of finding a multiplicative inverse element by modulus and multiplying by it while using a Chinese remainder theorem. However, the PF of RNS has a limited application, because the bits of the corresponding moduli, and therefore the residues, significantly differ, which leads to irrational usage of the bit grid registers. In addition, the first moduli must be equal to 2 and 3, which limits its usage while building a modules system with the same bit size.

In [35] modified perfect form (MPF) of RNS was developed, which satisfies the condition:

$$M_i \bmod p_i = 1, p_i\text{-}1 \text{ or } M_i \bmod p_i = \pm 1. \tag{4}$$

In this case coefficients m_i=1 and accordingly the sum in (2) change the sign. In addition to eliminating mentioned drawback of the PF RNS, it significantly reduces the sum $\sum_{i=1}^{z} b_i M_i m_i$ in (2), which in many cases will not exceed the value of the calculation range. This will simplify the finding of residue by modulo P.

However, currently there are no methods of constructing moduli systems that meet the conditions of MPF RNS and have the same bit size, which will allow using the bit grid registers for efficient asymmetric cryptography problems solving problems and noise-tolerant coding. This work is devoted to the elimination of this shortcoming.

3. Methods for construction MPF RNS moduli system of the same bit size

3.1. Factorization method

To construct a system of moduli with the same bit size for MPF RNS using the factorization method, we write expression (4) as a system:

$$\begin{cases} M_1 \bmod p_1 = \pm 1 \\ \ldots \\ M_z \bmod p_z = \pm 1. \end{cases} \tag{5}$$

Mathematical calculations similar to the ones performed in [33] lead to the following expression:

$$\sum_{i=1}^{z} \frac{1}{p_i} = \gamma \pm \frac{1}{\prod_{i=1}^{z} p_i}, \tag{6}$$

where γ =0, 1, 2, 3, ...

To simplify the calculations, the selected coefficient unlike the PF RNS, can be equal to 0, which for a given number of moduli corresponds to the largest value P. So the last equality can be described as:

$$\frac{1}{p_1} + \frac{1}{p_2} + \frac{1}{p_3} + \ldots + \frac{1}{p_{z-1}} + \frac{1}{p_z} = \pm \frac{1}{p_1 p_2 p_3 \cdots p_{z-1} p_z}. \tag{7}$$

Since the use of rational fractions necessarily presents rounding of the results with a certain accuracy, for the convenience of software implementation of the developed method, expression (7) should be reduced to a common denominator:

$$\sum_{i=1}^{z} M_i = \pm 1. \tag{8}$$

Suppose that last two moduli p_{z-1} and p_z are unknown. Therefore (8) can be defined as:

$$p_{z-1}p_z\left(p_2p_3 \cdots p_{z-2} + p_1p_3 \cdots p_{z-2} + \ldots + p_1p_2 \cdots p_{z-3}\right) + p_1p_2 \cdots p_{z-2}\left(p_{z-1} + p_z\right) = \pm 1. \tag{9}$$

Let's consider:

$$p_{z-1,z} = \frac{a, b - p_1p_2 \cdots p_{z-2}}{p_2p_3 \cdots p_{z-2} + p_1p_3 \cdots p_{z-2} + \ldots + p_1p_2 \cdots p_{z-3}}. \tag{10}$$

After substituting (10) into (9) and performing some mathematical transformations we get a condition that must be satisfied to determine a MPF RNS moduli set of any capacity with two unknown moduli:

$$\pm\left(p_2p_3 \cdots p_{z-2} + p_1p_3 \cdots p_{z-2} + \ldots + p_1p_2 \cdots p_{z-3}\right) + \left(p_1p_2p_{z-2}\right)^2 = ab. \tag{11}$$

Therefore, the left part (10) should be factorized, and the parameters a and b are determined based on it. Additionally, as a result of (10), the moduli p_z and p_{z-1} should be integers, so

$$\left(a, b - p_1p_2 \cdots p_{z-2}\right) \bmod \left(\left|p_2p_3 \cdots p_{z-2} + p_1p_3 \cdots p_{z-2} + \ldots + p_1p_2 \cdots p_{z-3}\right|\right) = 0. \tag{12}$$

Expressions (11) and (12) define the conditions to find any number of MPF RNS moduli, two of which are unknown.

3.2. The method of exhaustive search

For exhaustive search application it is advisable to choose formula (8). For four moduli we will have following expression:

$$p_1p_2p_3 + p_1p_2p_4 + p_1p_3p_4 + p_2p_3p_4 = \pm 1. \tag{13}$$

Figure 1 shows the graphical dependence of the modulo p_4 on the values of the moduli p_2 and p_3. As we can see the absolute value p_4 increases with increasing of p_2 and p_3 at p_1=-131. Integer values of parameters p_2, p_3 and p_4 create a moduli system that satisfies the conditions of MPF RNS.

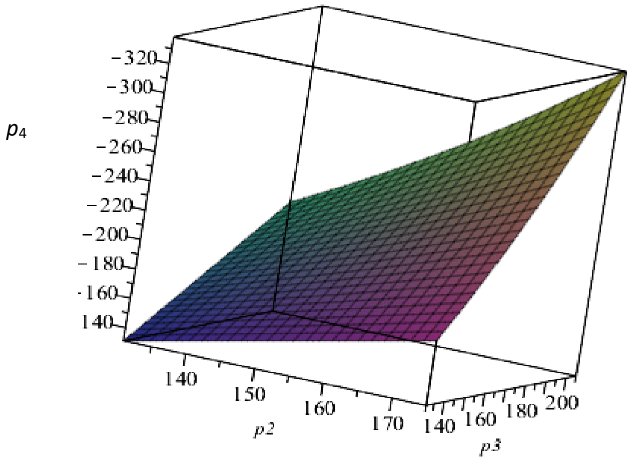

Figure 1: Graphical dependence of the modulo p_4 on values of the moduli p_2 and p_3

3.3 Replacement method

For four-moduli MPF RNS after replacement of p_2, p_3, p_4=a, b, c-p_1, and corresponding mathematical transformations, formula (13) is converted to the following expression:

$$abc - p_1^2\left(a+b+c\right) = \pm 1 - 2p_1^2 . \tag{14}$$

Reverse replacement allows returning to the original parameters:

$$\left(p_2+p_1\right)\left(p_3+p_1\right)\left(p_4+p_1\right) - p_1^2\left(p_2+p_3+p_4+p_1\right) = \pm 1 . \tag{15}$$

Dividing the left and right parts by p_1^2, we get the conditions that must be satisfied to create MPF RNS:

$$\left(p_2+p_1\right)\left(p_3+p_1\right)\left(p_4+p_1\right) \mathrm{mod}\, p_1^2 = \pm 1, \quad \frac{\left(p_2+p_1\right)\left(p_3+p_1\right)\left(p_4+p_1\right)\pm 1}{p_1^2} = p_2+p_3+p_4+p_1 . \tag{16}$$

The first expression (16) indicates that $\left(p_2+p_1\right)\left(p_3+p_1\right)\left(p_4+p_1\right) = k\cdot p_1^2 \pm 1$, where k=1, 2, … .

It means that $k\cdot p_1^2 \pm 1$ must be decomposed into at least two factors (then the third will be 1). The calculations show that in the vast majority of cases, following equality must be used to satisfy conditions (16):

$$\left(p_2+p_1\right)\left(p_3+p_1\right)\left(p_4+p_1\right) = -p_1^2 + 1 . \tag{17}$$

4. Results and discussion

4.1. Factorization method

For example, let`s consider MPF RNS, which consists of four moduli. In this case conditions (10) - (12) will be:

$$p_{3,4} = \frac{a,b - p_1 p_2}{p_1+p_2};\ \pm\left(p_1+p_2\right)+\left(p_1 p_2\right)^2 = ab;\ \left(a,b-p_{1,2}\right)\mathrm{mod}\left(p_1+p_2\right) = 0 \tag{18}$$

Suppose that the moduli bit size is 4 bits and p_1=8, p_2=-9. Then from (18) we get:

$$p_{3,4} = -\left(a,b+72\right) \text{ and } ab = \pm 1 + 5184 = \begin{cases} 5183 = 71\cdot 73 \\ 5185 = 5\cdot 17\cdot 61 \end{cases}.$$

All possible options for systems with four moduli for MPF RNS with p_1=8, p_2=-9 and calculation range are presented in table 1 (ab=5185) and table 2 (ab=5183).

Table 1

Possible options for systems with four moduli for MPF RNS with p_1=8, p_2=-9, ab=5185 and calculation range

№	*a*	*b*	p_3	p_4	*P*
1	1	5185	-73	-5257	27630792
2	-1	5185	-71	5113	26137656
3	5	1037	-77	-1109	6148296
4	-5	-1037	-67	965	4655160
5	17	305	-89	-377	2415816
6	-17	-305	-55	233	922680
7	61	85	-133	-157	1503432
8	-61	-85	-11	13	10296

Table 2

Possible options for systems with four moduli for MPF RNS with p_1=8, p_2=-9, ab=5183 and calculation range

№	*a*	*b*	p_3	p_4	*P*
1	1	5183	-73	-5255	27620280
2	-1	-5183	-71	5111	26127432
3	71	73	-143	-145	1492920
4	-71	-73	-1	1	72

It is important to note that this table requires clarification of the moduli signs according to expression (4). Therefore, a system of four 4-bit moduli with the same bit size, which satisfies the MPF RNS conditions, will have following values: -8, 9, 11, -13.

4.2. The method of exhaustive search

Table 3 presents possible options for sets of four 8-bit moduli for building MPF RNS.

Table 3
Possible options for sets of four 8-bit moduli for building MPF RNS

№	p_1	p_2	p_3	p_4
1	-131	134	151	-155
2	-134	141	143	-151
3	-151	178	203	-255
4	-169	177	179	-188
5	-197	199	241	-244
6	-208	217	219	-229

Obviously, this method is not suitable for finding a large number of high-bit size moduli, because it is time consuming.

4.3 Replacement method

Table 4 presents the procedure for finding moduli with the same bit size (5-7 bits) to form the MPF RNS.

Table 4
Procedure for finding moduli with the same bit size (5-7 bits) to form the MPF RNS.

n, bit	p_1	$-p_1^2+1$	Factorization	p_2+p_1	p_3+p_1	p_4+p_1	p_2	p_3	p_4
5	-19	-360	$-2^3 \cdot 3^2 \cdot 5$	2	4	-45	21	23	-26
6	-34	-1155	$-3 \cdot 5 \cdot 7 \cdot 11$	3	5	-77	37	39	-43
7	-76	-5775	$-3 \cdot 5^2 \cdot 7 \cdot 11$	5	7	-165	81	83	-89
7	-103	-10608	$-2^4 \cdot 3 \cdot 13 \cdot 17$	6	8	-221	109	111	-118

Based on results in tables 1-3 while creating the four-moduli MPF RNS, which are convenient for application in asymmetric cryptosystems and noise-tolerant coding problems, for all moduli of the same bit size the following relations are fulfilled: p_1, p_4 <0, p_2, p_3 >0.

5. Conclusions

In this research the methods for the four-moduli sets of the same bit size creation in a modified perfect form of residue number system are developed. This allows using bit grid registers more efficiently. This problem is relevant for usage in asymmetric cryptography and noise-protected coding algorithms. The theoretical bases of residue number system, its perfect and modified perfect forms are considered, their advantages and disadvantages are defined. It is shown that the most commonly used moduli are the power of two, Mersenne numbers and Fermat numbers require the finding the inverse element and multiplying by it, which makes it difficult to recover a decimal number from its residuals using the Chinese remainder theorem. A modified perfect form of residue number system simplifies this procedure. Three methods of moduli system formation have been developed: factorization, exhaustive search and replacement. The graphical dependence of the fourth modulo on two previous ones with one known modulo is presented. Different bit sizes of moduli sets are considered. It is shown that in four-moduli sets of the same bit size in a modified perfect form of residue number system, the first and fourth moduli are negative, the second and third - positive.

6. References

[1] N. Vozna, Ya. Nykolaichuk, O. Zastavnyy, V. Pikh, System complexity criteria and synthesis of high-performance multifunctional parallel ADC in Rademacher's and Haar-Krestenson's theoretical and numerical bases. Experience of Designing and Application of CAD Systems in Microelectronics (CADSM-2017): Proceedings of the 14th International Conference, 2017, pp. 218-221.

[2] N. Alrajeh et al. Error Correcting Codes in Wireless Sensor Networks: An Energy Perspective. Applied Mathematics & Information Sciences, 2015, pp.809-818.

[3] R. Bassoli, H. Marques, J. Rodriguez, K. Shum and R. Tafazolli, Network coding theory: A survey. Communications Surveys & Tutorials, IEEE, 2013, 15(4), pp.1950-1978.

[4] V. Bavya, Uthira Devi R., Optimizing the Precision of Digital Signal Processors Using Residue Number System. Imperial Journal of Interdisciplinary Research, 2016, vol.2 (4), pp. 1113-1122.

[5] Hu Zhengbing, V. Yatskiv, A. Sachenko, Increasing the Data Transmission Robustness in WSN Using the Modified Error Correction Codes on Residue Number System. Elektronika ir Elektrotechnika, 2015,Vol 21. № 1, pp. 76-81.

[6] R. Dridi, H. Alghassi, Prime factorization using quantum annealing and computational algebraic geometry. Scientific Reports, 2017, vol.7, pp. 158-167.

[7] V. Adki, S. Hatkar, A Survey on Cryptography Techniques. International Journal of Advanced Research in Computer Science and Software Engineering, 2016, vol. 6 (6), pp. 469-475.

[8] H. Xiao, H. Garg, J. Hu, G. Xiao, New Error Control Algorithms for Residue Number System Codes. Electronics and Telecommunications Research Institute, 2016, vol. 38 (2), pp.326-336.

[9] Tay Thian Fatt, Chang ChipHong, A new algorithm for single residue digit error correction in Redundant Residue Number System, Circuits and Systems (ISCAS), IEEE International Symposium IEEE, 2014, pp. 1748-1751.

[10] H. Lo, T. Lin, Parallel Algorithms for Residue Scaling and Error Correction in Residue Arithmetic, Wireless Eng. Technol, 2013, vol. 4 (4), pp. 198–213.

[11] S. Lysenko, K. Bobrovnikova, S. Matiukh, I. Hurman & O. Savenko, Detection of the botnets' low-rate DDoS attacks based on self-similarity. International Journal of Electrical & Computer Engineering, 2020, 10, 2088-8708. DOI: http://doi.org/10.11591/ijece.v10i4.pp3651-3659.

[12] Savenko O., S. Lysenko, A. Nicheporuk, B. Savenko. Metamorphic Viruses' Detection Technique Based on the Equivalent Functional Block Search. CEUR-WS, ISSN: 1613–0073. 2017. Vol. 1844. Pp. 555–569.

[13] Savenko O., S. Lysenko, A. Kryshchuk, Y. Klots. Botnet detection technique for corporate area network. Proceedings of the 7-th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Berlin (Germany), September 12–14, 2013. Berlin, 2013. Pp. 363–368. ISBN 978-1-4799-1426-5.

[14] Savenko O., Lysenko S., Kryschuk A. Multi-agent based approach of botnet detection in computer systems. Communications in Computer and Information Science. 2012. Vol. 291. PP.171-180, ISSN: 1865-0929.

[15] S. Lysenko, K. Bobrovnikova, O. Savenko and A. Kryshchuk, BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience Against the Botnets' Cyberattacks, Communications in Computer and Information Science, 1039 (2019) 127-143. doi: 10.1007/978-3-030-21952-9_10.

[16] S. Lysenko, K. Bobrovnikova & O. Savenko, A botnet detection approach based on the clonal selection algorithm. In 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT). IEEE (2018) 424-428.

[17] M. Deryabin, N. Chervyakov, A. Tchernykh, M. Babenko, M. Shabalina, High Performance Parallel Computing in Residue Number System. International Journal of Combinatorial Optimization Problems and Informatics, 2018, vol. 9 (1), pp. 62-67.

[18] P.V. Ananda Mohan, Residue Number Systems: Theory and Applications, Birkhäuser, 2016, 351 p.

[19] V.A. Krasnobayev, A.S. Yanko, S.A. Koshman, Method for arithmetic comparison of data represented in a residue number system. Cybernetics and Systems Analysis, 2016, vol. 52 (1), pp. 145–150.

[20] P.V. Ananda Mohan, RNS to binary conversion using diagonal function and Pirlo and Impedovo monotonic function, Circuits Syst. Signal Process, 2016, vol. 35, pp. 1063-1076.

[21] V.A. Krasnobayev, A.S. Yanko, S.A. Koshman, The method of error correction in the system of residual classes. Nauka i studia. Przemysl (Poland), 2015, №5 (136), pp. 51-62.

[22] V. Hema, M. Ganaga Durga, Data Integrity Checking Based On Residue Number System and Chinese Remainder Theorem In Cloud. International Journal of Innovative Research in Science, Engineering and Technology, 2014, vol.3 (3), pp. 2584-2588.

[23] M. Labafniya, M. Eshghi, RNS division algorithm for 2n-1 and 2n dividers. 22nd Iranian Conference on Electrical Engineering (ICEE): Proceedings, 2014, pp. 111-114.

[24] I.A. Aremu, K.A. Gbolagade, Information encoding and decoding using Residue Number System for {22n-1, 22n, 22n+1} moduli sets. International Journal of Advanced Research in Computer Engineering & Technology, 2017, vol. 6 (8), pp. 1260-1267.

[25] P. Patronik, S.J. Piestrak, Design of Reverse Converters for General RNS Moduli Sets {2k, 2n-1, 2n+1, 2n-1-1}. IEEE Transactions on Circuits and Systems, 2014, vol.10 (1), pp. 143-148.

[26] A.P. Fournaris, L. Papachristodoulou, L. Batina, N. Sklavos, Secure and Efficient RNS Approach for Elliptic Curve Cryptography. Trustworthy Manufacturing and Utilization of Secure Devices (TRUDEVICE 2016): Proceedings of the 6th Conference, Barcelona, 2016, pp. 121-126.

[27] A.P. Fournaris, L. Papachristodoulou, L. Batina, N. Sklavos, Residue number system as a side channel and fault injection attack coun- termeasure in elliptic curve cryptography. Design and Technology of Integrated Systems in Nanoscale Era (DTIS): Proceedings of the 2016 International Conference, 2016, pp. 1–4.

[28] I.R. Fadulilahi, E.K. Bankas, J.B.A.K. Ansuura, Efficient Algorithm for RNS Implementation of RSA. International Journal of Computer Applications, 2015, vol. 127 (5), pp. 14-19.

[29] I.Z. Yakymenko, M.M. Kasianchuk, S.V. Ivasiev, A.M. Melnyk, Ya.M. Nykolaichuk, Realization of RSA cryptographic algorithm based on vector-module method of modular exponention. Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET–2018): Proceedings of the XIV–th International Conference, L'viv–Slavske, 2018, pp.550-554.

[30] N. Vivek, K. Anusudha, Design of RNS Based Addition Subtraction and Multiplication Units. International Journal of Engineering Trends and Technology, 2014, vol. 10 (12), pp. 593-596.

[31] T. Rajba, A. Klos-Witkowska, S. Ivasiev, I. Yakymenko, M. Kasianchuk, Research of Time Characteristics of Search Methods of Inverse Element by the Module. Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS–2017): Proceedings of the 2017 IEEE 9th International Conference, Bucharest, Romania, vol.1, 2017, pp.82–85.

[32] Hu. Zhengbing, I. Dychka, M. Onai, A. Bartkoviak, The Analysis and Investigation of Multiplicative Inverse Searching Methods in the Ring of Integers Modulo M. International Journal of Intelligent Systems and Applications (IJISA), 2016, vol. 8, №11, pp. 9-18.

[33] M. Karpinski, S. Rajba, S. Zawislak, K. Warwas, M.Kasianchuk, S. Ivasiev, I. Yakymenko. A Method for Decimal Number Recovery from its Residues Based on the Addition of the Product Modules Proceedings of the 2019 IEEE 9th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS–2019) – Metz, France, vol.1, 2019, pp. 13–17.

[34] M. Kasianchuk, I. Yakymenko, I. Pazdriy, O. Zastavnyy, Algorithms of findings of perfect shape modules of remaining classes system. The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM-2015): Proceedings of the XIII International Conference. Polyana-Svalyava, 2015, pp.168-171.

[35] M.N. Kasianchuk, Ya.N. Nykolaychuk, I.Z. Yakymenko, Theory and Methods of Constructing of Modules System of the Perfect Modified Form of the System of Residual Classes. Journal of Automation and Information Sciences, 2016, vol.48, №8, pp.56-63.