Scientific Issues
Jan Długosz University
in Częstochowa
Mathematics XXII (2017)
117–127
DOI http://dx.doi.org/10.16926/m.2017.22.09

THEORETICAL AND METHODOLOGICAL BASES OF MODULAR TECHNOLOGY OF PARALLEL TABULAR COMPUTATIONS USING UNIVERSAL PROCESSORS

MIKHAIL SELIANINAU

Abstract

In this paper, we deal with the methodology for application of the theory and methods of the tabular technology of modular information processing on the basis of modern computing machinery. The use of the minimal redundant modular number system and the interval-modular form of representation of an integer number determined by its modular code creates the computer-arithmetical basis of a methodology under consideration. The main advantage of the offered methodology consists inincrease of the computation speed and accuracy at the organization of high-precision arithmetic processing of multi-digit data by means of universal processors on the basis of minimal redundant modular encoding method.

1. Introduction

At present, the branch of high-performance computing is developing extremely rapidly. The development of computer science determines a rapid increase of computing scales that leads to qualitatively new requirements imposed on numerical methods and computing algorithms. The main requirement is the receiving in an acceptable time of the correct results of solving the task which are not distorted by rounding errors.

The need for science in high-speed and high-precision computing has always existed and now it is especially urgent. The most important tasks sensitive to the computing speed and accuracy arise in various areas: in physics and cosmology, in experimental mathematics and computational geometry, in the growing fields of high-tech industry and so on [1–5]. A characteristic feature of such problems is their high dimensionality. In this connection, the time to solve the task becomes a very critical parameter.

• *Mikhail Selianinau* — m.selianinov@ajd.czest.pl
Jan Długosz University in Częstochowa.

The main reason affecting the accuracy of calculations is the use of rounding in arithmetic operations that is caused by the fixed and relatively small length of operands in modern processors. At the same time, the choice of the optimal algorithm which is stable to rounding errors is often time-consuming. Therefore, the main way to get the correct results of numerical calculations is the use of the program processing of extremely large numbers for which the digit capacity exceeds the finite computer word length. However, the construction of large number arithmetic on the basis of positional number systems leads to a significant and often unacceptable reduction of the processing speed.

The top manufacturers of modern processor platforms first of all place special emphasis on parallelism. The technologies of vector and multi-core computing are actively developing. Now there are more perfect SIMD-commands that allow us to process simultaneously increasingly more pairs of operands, the compilers can perform automatic vectorization under the condition of the suitable data structure, the number of processor cores grows steadily and moreover there are specialized multicore coprocessors [6]. As a result, there is a task of developing a new approach to representation of multi-bit numbers which allows an easy parallelization of calculations at the level of elementary arithmetic operations.

In view of the sequential internal structure of classical positional arithmetic algorithms the traditional implementation of both high-speed and high-precision computations on their basis is most often inefficient and, in many cases, is just unacceptable. Therefore, research and development of new flexible techniques of reliable parallel data processing in the set of real numbers as well as on more complicated mathematical models are very relevant today.

The use of modular arithmetic (MA) which has a natural internal data parallelism is a promising scientific direction in the field of high-speed high-precision computation [7–10]. The lack of carries between the adjacent digits of modular number representation allows performing the main arithmetic operations easily and quickly, vectorizing them effectively and allocating on computation kernels.

The method of implementing of computer arithmetic on the base of the number representation in the modular computational basis is the most consistent among others with the concept of developing modern high-performance systems and technologies of parallel programming.

2. The basic notation and terminology

Let us introduce the following notation: $\mathbf{Z}$ is the set of all integers;

$\lfloor x \rfloor$ denotes the floor of x, i.e. the greatest integer less than or equal to x, $\lfloor x \rfloor = \max\{\, y \in \mathbf{Z} \mid y \le x \,\}$;

$\lceil x \rceil$ denotes the ceiling of x, i.e. the smallest integer greater than or equal to x, $\lceil x \rceil = \min\{\, y \in \mathbf{Z} \mid y \ge x \,\}$;

$\mathbf{X} \times \mathbf{Y} = \{\, \forall(x, y) \mid x \in \mathbf{X},\ y \in \mathbf{Y} \,\}$ is the Cartesian product of two sets $\mathbf{X}$ and $\mathbf{Y}$;

$\gcd(A, B)$ stands for the greatest common divisor of two integers A and B;

$\mathbf{Z}_m = \{0, 1, \ldots, m-1\}$ and $\mathbf{Z}_m^- = \left\{-\left\lfloor \frac{m}{2} \right\rfloor, -\left\lfloor \frac{m}{2} \right\rfloor + 1, \ldots, -\left\lceil \frac{m}{2} \right\rceil - 1\right\}$ are the sets (rings) of least nonnegative residues and absolutely least residues modulo $m > 1$, respectively;

$|x|_m$ denotes the element of $\mathbf{Z}_m$ congruent to x modulo m;

$\left|X^{-1}\right|_m$ designates the multiplicative inversion of an integer X modulo m ($\gcd(X, m) = 1$);

$m_1, m_2 \ldots, m_k$ are the natural modules ($k \ge 2$);

$M_l = \prod_{i=1}^{l} m_i$, $M_{i,l} = M_l / m_i$ $\ (l = k-1, k;\ i = 1, 2, \ldots, k)$

$\left(|X|_{m_1}, |X|_{m_2}, \ldots, |X|_{m_k}\right)$ represents a modular code (MC) of the integer $X \in \mathbf{Z}$ with respect to the basis $\{m_1, m_2, \ldots, m_k\}$.

3. The basic principles of modular number processing

As is known, the parallel computing structures play a fundamental role in the modern computer science and its numerous applications. The analysis of the modern directions of development of parallel computing structures shows that in recent years special attention has been paid to the so-called parallel ring structures based on modular number systems (MNS). The theoretical basis of MA is created by abstract algebra and number theory [7, 11–13].

A classical non-redundant MNS on the set $\mathbf{Z}$ is determined by pairwise relatively prime modules $m_1, m_2 \ldots, m_k$ $(\gcd(m_i, m_j) = 1;\ i, j = 1, 2, \ldots, k;\ i \ne j;\ k > 1)$ using the definition of a homomorphic mapping $\mathbf{\Phi} : \mathbf{Z} \to \mathbf{Z}_{m_1} \times \mathbf{Z}_{m_2} \times \ldots \times \mathbf{Z}_{m_k}$ which assigns the set $(\chi_1, \chi_2, \ldots, \chi_k)$ of residues $\chi_i = |X|_{m_i}$ of the division of an integer X by a modulo m_i $(i = 1, 2, \ldots, k)$ to each $X \in \mathbf{Z}$. In this case the standard notation $X = (\chi_1,\ \chi_2, \ldots, \chi_k)$ is used.

It is known that the set of integers satisfying to the system of simultaneous linear congruences

$$\begin{cases} X \equiv \chi_1 \pmod{m_1} \\ X \equiv \chi_2 \pmod{m_2} \\ \quad \dots \\ X \equiv \chi_k \pmod{m_k} \end{cases} \tag{1}$$

corresponds to an MC of integer X. If $m_1, m_2, \dots, m_k$ are pairwise relatively prime modules, then the simultaneous congruences (1) have a unique solution being the residue class modulo M_k determined by the congruence

$$X \equiv \sum_{i=1}^{k} M_{i,k}\mu_{i,k}\chi_i \pmod{M_k}, \tag{2}$$

where $\mu_{i,k} = \left|M_{i,k-1}^{-1}\right|_{m_i}$. In essence, the formula (2) represents the so-called Chinese remainder theorem (CRT) [7, 11].

The practical application of the MNS assumes that instead of a residue class there is only a single integer corresponding to the MC $(\chi_1,\ \chi_2, \dots, \chi_k)$. Therefore, in the MNS with the basis $\{m_1, m_2, \dots, m_k\}$ one or the other set of representatives of the residue classes is used as the numerical range $\mathbf{D}$ in order to ensure the required one-to-one mapping $X \to (\chi_1,\ \chi_2, \dots, \chi_k)$. In this case, the maximum cardinality of a set of integers $\mathbf{D}$ is equal to M_k.

In computer applications the rings $\mathbf{Z}_{M_k}$ and $\mathbf{Z}^{-}_{M_k}$ are usually used as a numerical range of the MNS. Taking the above into account, in the first case the modular coding is defined as a mapping $\mathbf{\Phi}_{MNS} : \mathbf{Z}_{M_k} \to \mathbf{Z}_{m_1} \times \mathbf{Z}_{m_2} \times \dots \times \mathbf{Z}_{m_k}$ assigns the MC $(\chi_1, \chi_2, \dots, \chi_k)$ to each $X \in \mathbf{Z}_{M_k}$. A decoding mapping $\mathbf{\Phi}^{-1}_{MNS} : \mathbf{Z}_{m_1} \times \mathbf{Z}_{m_2} \times \dots \times \mathbf{Z}_{m_k} \to \mathbf{Z}_{M_k}$ based on the relationship (2) operates according to the following rule

$$X = \left|\sum_{i=1}^{k} M_{i,k}\chi_{i,k}\right|_{M_k} \quad (\chi_{i,k} = |\mu_{i,k}\chi_i|_{m_i}). \tag{3}$$

Let two arbitrary integers A and B be represented in the MNS with the base numbers $m_1, m_2 \dots, m_k$, i.e. $A = (\alpha_1, \alpha_2, \dots, \alpha_k)$, $B = (\beta_1, \beta_2, \dots, \beta_k)$ $\big(\alpha_i = |A|_{m_i},\ \beta_i = |B|_{m_i},\ i = 1, 2, \dots, k\big)$. Then each ring (modular) operation $\circ \in \{+, -, \times\}$ on them has the following general form:

$$A \circ B = \Big(|\alpha_1 \circ \beta_1|_{m_1}, |\alpha_2 \circ \beta_2|_{m_2}, \dots, |\alpha_k \circ \beta_k|_{m_k}\Big). \tag{4}$$

In other words, the result of the operation " $\circ$ " on two integers A and B in the modular representation is formed by an independent application of " $\circ$ " to all pairs of their residues $(\alpha_i,\ \beta_i)$ with respect to the corresponding modules m_i : $|\alpha_i \circ \beta_i|_{m_i}$, $i = 1, 2, \dots, k$. The main fundamental advantage of the MA over a position number systems (PNS) arithmetic consists in parallel digit-wise realization of ring operations (see (4)).

In an MNS, the MC does not contain explicit information about the value of the element of the numerical range that corresponds to it. In contrast to the modular operations (4), for the realization in MA of the so-called non-modular operations it is not enough only to have the separate residue values but the evaluation of positional values of the whole number is also required. Therefore, for the performing non-modular operations in MNS it is necessary to use the such forms for number representation (via the residues of MC) which allow us to obtain some characteristics for determining the position of a given number in a numerical range. Such operations include scaling, division (in the general case), multiplication of fractions, sign detection, overflow control, direct and inverse conversions of the positional and modular codes, errors detection and correction.

It is obvious that in the final analysis the complexity of computing of the so-called integral characteristics of MC determines the effectiveness of a MA created on their basis. The optimization of integral characteristic base of a MNS (first of all, MNS with integer ranges in the first place) plays a key role in the development of efficient MA [7, 8, 14-16].

4. Minimal redundant modular coding

Within the framework of development of the theory and applications of modular computing structures (MCS) a special place is devoted to optimization of methods and algorithms of performance of nonmodular operations in MSS with respect to the input redundancy of coding of the elements of the numerical ranges, the implementation time of these operations, the throughput of modular processors, etc.

It is known that the use of code redundancy often allows us to improve significantly the arithmetic or other properties of the number system, including a MNS. The immediate application of expression (3) as the basic form of integer numbers representation for the synthesis of non-modular procedures is practically unacceptable due to the complexity of direct computer implementation, especially in the case of large values of M_k. This is caused by the fact that the CRT demands a large modulo operations which are very labor-consuming, especially when using a wide ranges. In addition, the CRT is not well adapted for performing operations on the elements from the symmetric ranges and is not suitable for design of scaling procedures.

At the same time, the parallel representation forms of the integer numbers on the basis of CRT (3), which provide a significant simplification of non-modular operations and have much better realization properties in comparison with non-redundant analogs, can be obtained by introducing some code redundancy.

The priority positions in this context belong to a minimal redundant modular coding, an interval index characteristic and an interval modular form (IMF) of integer numbers associated with it [7, 8, 15, 16]. A minimal redundant modular coding $\mathbf{\Phi}_{MRMNS} : \mathbf{D} \to \mathbf{Z}_{m_1} \times \mathbf{Z}_{m_2} \times \ldots \times \mathbf{Z}_{m_k}$ provides the use of a working numerical range $\mathbf{D}$ whose cardinality is less than the cardinality of the range of classical (non-redundant) MNS with modules $m_1, m_2, \ldots, m_k$, i.e. $|\mathbf{D}| < M_k$.

A redundant MNS on the set of integers $\mathbf{Z}$ is determined by means of $k > 1$ pairwise relatively prime modules $m_1, m_2, \ldots, m_k$ and an additional module m_0 satisfying the conditions $\gcd(m_i, m_0) = 1$ $(i = 1, 2, \ldots, k)$. In this case, a set $\mathbf{Z}_{2M}^{-} = \{ -M, -M+1, \ldots, M-1 \}$, where $M = m_0 M_{k-1}$, is usually used as a numerical range $\mathbf{D}$. In view of the condition $2M < M_k$, the additional module m_0 should be chosen on the basis of the following estimation $2m_0 < m_k$. The resulting redundant MNS is naturally a narrowing of the initial non-redundant MNS and possesses all its advantages.

The decoding mapping $\mathbf{\Phi}_{MRMNS}^{-1} : \mathbf{Z}_{m_1} \times \mathbf{Z}_{m_2} \times \ldots \times \mathbf{Z}_{m_k} \to \mathbf{D}$, which is a restoration of a number $X \in \mathbf{D}$ by residues of its MC $(\chi_1, \chi_2, \ldots, \chi_k)$, is carried out using the so-called interval index (II) $I(X)$ and a corresponding IMF of a number X which is represented as

$$X = \sum_{i=1}^{k-1} M_{i,k-1} \left| M_{i,k-1}^{-1} \chi_i \right|_{m_i} + I(X) M_{k-1}. \tag{5}$$

According to a CRT, the interval-index characteristic $I(X)$ of the MC is uniquely determined by the relation (5) [7, 8]. At the same time, an IMF (5) is free from the above-mentioned disadvantages of using a CRT (see (2) and (3)).

In order to achieve the required level of code redundancy of a MNS with the range $\mathbf{D} = \mathbf{Z}_{2M}^{-}$ the conditions under which the remainder $\hat{I}_k(X) = |I(X)|_{m_k}$ uniquely determines the II $I(X)$ of each number $X \in \mathbf{D}$ were obtained in [8, 15]. For this purpose it is necessary and sufficient that the kth module m_k of the MNS satisfies the condition $m_k \geq 2m_0 + \rho$, where $\rho = \max\{\rho_{k-1}(X)\}$ is the maximum value of the rank characteristic of the $(k-1)$th order [7, 8]. As this takes place, the minimum redundancy is achieved in the case when the equality $m_k - 2m_0 - \rho = |m_k - \rho|_2$ is satisfied.

The main advantage of MRMNS over non-redundant analogs consists in a significant simplification of the calculation of the interval index characteristic $I(X)$. In spite of the fact that the introduced redundancy is very small,just owing to it, the calculation of the II $I(X)$ is actually reduced to summation of k residues modulo m_k, i.e. it is a trivial operation. In contrast, in order to calculate the traditionally used integral characteristics of MC (the mixed-radix digits, the rank characteristics and some others) it is necessary to perform the calculation of the sums of k sets of residues with respect to modules of MNS.

The use of a MRMNS instead of a MNS reduces the complexity of algorithm for computation of a base interval-index characteristic expressed by the number of the necessary of tables and the number of modulo additions from $O(k^2)$ to $O(k)$. In the final analysis, in particular this provides a simplicity of non-modular procedures synthesized on the basis of IMF (5), first of all, of the procedures for conversion of a MRMC to a position code and scaling which play an extremely important role in the technology of modular information processing (TMIP). The realization of the advantages of MRMNS over non-redundant MNS with respect to minimizing computational costs of non-modular operations opens up wide possibilities for creation of systems that are characterized by minimal computational burden in a class of equivalent modular analogs in terms of productivity, calculation accuracy and complexity of the basic reconfiguration mechanism [7, 8].

5. The fundamental advantages of a MCS over the position computing structures

An MNS has the maximum level of internal parallelism among a set of all number systems. Due to this property, the MA allows us:

- to expand the set of used modules without complicating the algorithms of ring operations, i.e. to increase the digit capacity of the elements of the numeric ranges;
- to change the number of the bases of MNS, i.e. the accuracy of data processing;
- to perform efficiently the tabular calculations both at the hardware and software level;
- to apply the principle of formal information processing at the modular segments of computational procedures; in the framework of this principle the cardinality of the numerical range is focused only on the final results of the calculations;
- to organize extremely simply the multiprocessor mode of calculations;
- to rebuild flexibly a system configuration changing the sets of modules and operating tables.

The features listed above as well as some other unique properties of MA open exclusively wide opportunities for execution of parallel high-precision calculations at a qualitatively new level (with respect to performance, simplicity of accuracy change and flexibility of operating modes).

The MCS constitute a unique structural method for the decomposition of computation procedures into a set of subprocesses independent from each other. The natural code parallelism along with the properties listed above provides a number of fundamental advantages of an MNS over a PNS, in particular:

- a complete independence of the time period of modular (ring) operations from the number of bases and hence from the code length of MNS;
- the high computational speed for the ranges of large numbers;
- simplicity of computation pipelining at the level of low-bit operations;
- an exclusively high performance of data processing for mathematical models more complicated than the real ranges, for example, the sets of complex numbers, quaternions, polynomials, etc.;
- efficiency of modular code constructions for error control and correction.

Because of the noted advantages an MCS ideally correlates with the concepts of advanced computer technology including the underlying ideologies of supercomputer, multicore processor, neural network and some similar systems of parallel information processing [7-10].

Taking into account the reasons mentioned above, during the past 10-15 years the modular direction goes through a stage of rapid development in computer science, digital signal processing and other fields of science and technology.

6. The methodology for constructing the multiprocessor information processing in MRMNS

The information processing technology is evolving rapidly and steady expands its applications and scope. Along with a hardware approach for creating high-performance parallel information processing systems (PIPS) based on MNS, the alternative multiprocessor methodology which allows us to realize the advantages of MCS at the software level, can also be applied.

If some computation process is modular in nature, i.e. is composed of modular operations (see (4)), then it can be parallelized by executing subprocesses associated with the different modules of MNS on separate universal processors (cores) of modern computers. At the same time, one of the most important (from a practical point of view) properties of MCS consisting in their unique reconfiguration capability can be used to a great extent.

The ability of MCS to change flexibly its own configuration is conditioned by the table nature of MA and in practice is achieved by trivial changing of sets (arrays) of useful constants, including the bases of MNS and corresponding operation tables.

A multiprocessor PIPS functioning in MRMNS with pairwise relatively prime modules $m_1, m_2, \ldots, m_k$ and the operating range $\mathbf{D} = \mathbf{Z}_{2M}^{-}$ is realized by using the k universal processors (cores) $PC_1, PC_2, \ldots, PC_k$, the ith of which performs the role of the modular datapath with respect to the base $m_i\,(i = 1, 2, \ldots, k)$. Thus, when executing any modular operation of the type (4) on integers A and B represented by their MRMC, the processor PC_i evaluates the ith digit $\gamma_i = |\alpha_i \circ \beta_i|_{m_i}$ of the result $C = A \circ B$ of the operation $\circ \in \{+, -, \times\}$. Therefore, in a multiprocessor PIPS the modular computing processes are reduced to subprocesses executed independently of each other on different processors (cores) according to the corresponding modules of MRMNS.

It is easy to see that an arbitrary computational process that does not contain logical operations in principle can always be transformed into a purely single modular computational process which in the most general case is not a one-step process but has a recursive organization and represents a set of typical (standard) computational procedures (segments). It is necessary to bear in mind that the application of the modular mode (also called the formal calculation mode (FCM)) is justified only when the calculation results for the modular segment of the implementable procedure do not exceed the bounds of the standard operating range of the MRMNS. But at the same time, the results of intermediate calculations may not meet this requirement. The noted circumstance exerts a decisive influence on the choice not only of the range $\mathbf{D}$, but also of the range of input data $\hat{\mathbf{D}} = \mathbf{Z}_{2P} = \{-P, -P+1, \ldots, P\} \subset \mathbf{D}$ (P is a some natural number). In this connection, the problem of choosing the cardinalities $2P$ and $2M$ of the ranges $\hat{\mathbf{D}}$ and $\mathbf{D}$, respectively, is governed by the strategy ensuring the conditions that allow us to realize all the most important advantages of MRMA to the maximum extent [7, 8]. This goal can be achieved if in the FCM the final results for any modular segment of the computational procedures are the elements of the range $\mathbf{D}$ for all permissible values of the input variables.

The multiprocessor PIPS on the basis of the MRMA allow us to use systems of modules $m_1, m_2, \ldots, m_k$ which can theoretically take on any integer values within the range of modern computers including the biggest ones, for example $2^{16} \pm p, 2^{32} \pm p, 2^{64} - p$ $(p = 0, 1, 2, \ldots)$. Thus, the essentially new possibilities (in comparison with traditional approaches) are opened up for extending the used number range in order to ensure the necessary conditions for the wider and more effective application of the formal calculation mode for the implementation of both modular and quasi-modular processes.

Another distinctive and very important feature of the modular multiprocessor PIPS is the extreme simplicity of modification of the used set of modules $m_1, m_2, \ldots, m_k$ and the corresponding package of operating tables by the purely programmable way, i.e. the simplicity of system reconfiguration. This reconfiguration operation can also be performed taking into account the possible change (increase or decrease) of the number of processors (cores).

7. Conclusions

In conditions of the rapid development and wide distribution of modern computer technology the multiprocessor systems of modular information processing represent a very promising and competitive alternative towards the systems which designed within the framework of the traditional methods that focus mainly on hardware-intensive approach.

The originality of the research and development of the new multiprocessing PIPS is determined by using the MRMNS as a computer-arithmetic basis. The MRMNS allow us to develop a much simpler and more effective arithmetic both on the real ranges and as well as on more complicated mathematical models (the sets of complex numbers, quaternions, polynomials, etc.) in comparison with the classical MNS [7, 8, 15-19].

The implementation of the formulated approach allows us:

- to extend or reduce the internal composition of the current set of modules without changing the execution procedure for the modular segments of computation processes, i.e. to regulate the calculation accuracy;

- to use super-large modules (in the range of values from 2^{16} to 2^{64}) and therefore the MNS operating ranges of an extra-high cardinality;

- to perform efficiently a lookup table data processing by means of arrays having a volume up to 2^{32} or more elements;

- to expand considerably the action limits of the FCM for computer models of computational procedures;

- to reconstruct flexibly the configuration of the modular information processing system by the programmable change of the sets of basic modules and operating tables.

It should be emphasized that with the evolution of multiprocessor computers the realization of the advantages of MNS at the program level is extremely simplified. At the same time, the attainable practical effects increase significantly, this primary concerns the best performance and accuracy.

References

[1] U. Kulisch, *Computer Arithmetic and Validity: Theory, Implementation, and Applications*. De Gruyter Studies in Mathematics 33, Berlin, 2013.

[2] G.B. Arfken, H.J. Weber, F.E. Harris, *Mathematical Methods for Physicists. A Comprehensive Guide. 7th edition*. Academic Press, Boston, 2012.

[3] C.W. Ueberhuber, *Numerical Computation 1: Methods, Software, and Analysis*. Springer-Verlag, Berlin, 2013, DOI 10.1007/978-3-642-59118-1.

[4] C.W. Ueberhuber, *Numerical Computation 2: Methods, Software, and Analysis*. Springer-Verlag, Berlin, 2013.

[5] A.R. Krommer, C.W. Ueberhuber, *Numerical Integration: on Advanced Computer Systems (Lecture Notes in Computer Science)*. Springer-Verlag, Berlin, 2008.

[6] S. Mittal, *A Survey of Techniques for Architecting and Managing Asymmetric Multicore Processors*. ACM Computing Surveys, **48:3** (2016), 45:1–45:38, DOI 10.1145/2856125.

[7] A.A. Kolyada, I.T. Pak, *Modular Structures of Pipeline Digital Information Processing*. University Press, Minsk, 1992 (in Russian).

[8] A.F. Chernyavsky, V.V. Danilevich, A.A. Kolyada, M.Y. Selyaninov, *High-speed Methods and Systems of Digital Information Processing*. Belarusian State University Press, Minsk, 1996 (in Russian).

[9] P.V. Ananda Mohan, *Residue Number Systems: Algorithms and Architectures*. Kluwer Academic Publishers, 2002.

[10] A. Omondi, B. Premkumar, *Residue Number Systems: Theory and Implementation*. Imperial College Press, London, 2007.

[11] I.M. Vinogradov, *Fundamentals of Number Theory*. Nauka, Moscow, 1981 (in Russian).

[12] M.F. Atiyah, I.G. Macdonald, *Introduction to Commutative Algebra*. Addison-Wesley Publishing Co., Reading, Massachusetts, 1969.

[13] D.S. Dummit, R.M. Foote, *Abstract algebra. 3rd edition*. John Wiley & Sons, Inc., Hoboken, NJ, 2004.

[14] A.A. Kolyada, M.Y. Selyaninov, *On the formation of the integral characteristics of the codes of residue number systems with the symmetrical range*. Cybernetics **4** (1986), 20–24 (in Russian).

[15] M. Selyaninov, *Modular technique of parallel information processing*. Scientific Issues of Jan Długosz University of Czestochowa, Mathematics **XIII** (2008), 43–52.

[16] M. Selyaninov, *Construction of modular number system with arbitrary finite ranges*. Scientific Issues of Jan Długosz University of Czestochowa, Mathematics **XIV** (2009), 105–115.

[17] M. Selianinau, *High- speed modular structures for parallel computing in the space of orthogonal projections*. Scientific Issues, Jan Długosz University of Czestochowa, Ser. Technical and IT Education, **V** (2010), 87–96.

[18] M. Selyaninov, *Modular number systems in a complex plane*. Scientific Issues of Jan Długosz University of Czestochowa, Mathematics **XV** (2010), 131–138.

[19] M. Selyaninov, *Arithmetic of quadratic minimal redundant modular number systems*. Scientific Issues of Jan Długosz University of Czestochowa, Mathematics **XVI** (2011), 129–134.

Received: May 2017

Mikhail Selianinau
Jan Długosz University,
Institute of Technics and Safety Systems,
42-200 Częstochowa, Al. Armii Krajowej 13/15, Poland
E-mail address: m.selianinov@ajd.czest.pl