Scientific Issues
Jan Długosz University
in Częstochowa
Mathematics XIX (2014)
217–226

APPLICATION OF MODULAR COMPUTING TECHNIQUE FOR HIGH SPEED IMPLEMENTATION OF CYCLIC CONVOLUTION

MIKHAIL SELYANINOV

Abstract

This article is a continuation of research on the modular computing structures defined on the set of polynomials over finite rings of integers. Advantages of minimal redundant polynomial-scalar modular number system are demonstrated on the example of computing cyclic convolution of discrete signals. Methods of execution of ring arithmetical operations as well as coding and decoding operations are considered.

1. Introduction

At the present time, quality of the execution of information processing procedures in particular data domain is largely determined by the selected mathematical model of the organization of information processing and the information technology implemented on this basis. During the last years specialists in the field of the analysis and processing of digital information have an increased interest in parallel forms of information transform on the basis of modular computing structures (MCS) [1–3].

Usage of arithmetic of modular number systems (MNS) is especially effective, first of all, in such areas as digital signal processing (DSP), synthesis of fast algorithms for discrete orthogonal conversions, cryptography, error-correcting codes, etc. Possessing a maximum level of internal parallelism, the MCS represent a unique means of decomposition of computing processes into independent from each other elementary sub-processes defined on mathematical models with elements of small digit capacity.

The main advantages of the MCS are the high efficiency, which is achieved through parallelism of modular arithmetic (MA) algorithms and their pipelining using tabular method of calculation, the possibility of mathematical calculations with high accuracy, effectiveness of control of failure situations

• *Mikhail Selyaninov* — e-mail: m.selininov@ajd.czest.pl
Jan Długosz University in Częstochowa.

by means of correcting modular codes. Due to regularity and uniformity the MCS give maximum effect with the use of special VLSI.

2. Formulation of the problem

The most optimal range of applications of the MCS constitutes DSP systems, which have higher requirements for such features as performance, computational accuracy and fault tolerance. The spectrum of the modern applications of the MCS includes numerous high-speed implementations of the laborious computing procedures belonging to algorithmic kernels of DSP systems for one or another purpose.

In theory and applications of DSP the discrete linear system with constant parameters, whose mechanism of action is based on the calculation of discrete convolution, are of fundamental importance. Calculation of the convolution of two periodic sequences is a widely used task of DSP. For example, digital filtering is based on calculating a convolution of the input signal and the impulse response of the filter. Also, in a number of cryptography tasks there appears a necessity of multiplication of two numbers whose magnitudes exceed limits in which the hardware representation of operations on the basis of existing computing techniques is possible. In particular, the Schönhage-Strassen method [4] reduces multiplication of large integers to evaluation of convolution of the sequences associated with arrays of digits of their representation in a positional number system.

It should be noted that the calculation of convolution by the "direct" method requires the excessive computational cost. Existing techniques based on the discrete Fourier transform (DFT) allow us to reduce computational complexity for certain values of the length of the convolution due to the existence of fast algorithms for computing the DFT. The traditional calculation of the convolution by means of DFT for large lengths of convoluted sequences may lead to computational errors, sometimes significant. This is due to the fact that the values of basis functions of the DFT are irrational numbers and the calculations can be presented only with limited accuracy, in connection with finite digit length of a computer.

In recent years, experts in the field of DSP exhibit heightened interest in polynomial MA. It is caused by increment in this area of the methods which are based on polynomial transformations and, as a consequence, by sharp increase of an amount of operations over polynomials in synthesized computing procedures.

Application of polynomial MA allows us to carry out calculations of the large convolutions by replacing them by a sequence of short convolutions on the basis of special methods of multiplication of polynomials. This also

allows us to implement a parallelization of computations at a level of micro-operations. If two initial sequences of discrete samples are represented in the polynomial form, then the cyclic convolution procedure can be reduced to the multiplication of polynomials.

The advantages and application features for the implementation of MNS procedures of such a class defined on the set of polynomials are demonstrated below by the example of computation of cyclic convolution of sequences $a_0, a_1, \ldots, a_{n-1}$ and $b_0, b_1, \ldots, b_{n-1}$, where the elements of the output sequence are defined by the equation:

$$c_\nu = \sum_{u=0}^{n-1} a_{|\nu-u|_n} b_u \quad (\nu = 0, 1, \ldots, n-1) \tag{1}$$

3. Polynomial modular number systems

There are many scientific and applied researches demanding of processing the information presented in the form of polynomials. Operations over polynomials are very important in modern computer algebra, DSP, coding theory, cryptography, etc. At the same time, modular technology of parallel computing structures defined on polynomial ranges is of great interest.

Let us consider the set $\mathbf{Z}_m[x]$ of all polynomials of finite degree with coefficients from the ring $\mathbf{Z}_m = \{0, 1, \ldots, m-1\}$ of absolutely least residues modulo m, where m is a natural number, and the real argument x. In this case the technique of constructing a MNS [5] first of all requires the creation of the complete set of residues (CSR) with respect to selected pairwise relatively prime polynomial modules. The following theorem is true.

Theorem 1. *In the set* $\mathbf{Z}_m[x]$ *for any polynomial* $f(x)$ *and arbitrary polynomial modules* $p(x)$ *with the degree* $\deg p(x) \geq 1$ *there are unique elements* $q(x)$ *and* $r(x)$ *such that*

$$f(x) = q(x)\, p(x) + r(x) \quad (\deg r(x) < \deg p(x)). \tag{2}$$

Let $p(x)$ be any element of s-th degree from $\mathbf{Z}_m[x]$. Then, according to Theorem 1, the set of all residual $r(x)$ of division of $f(x)$ by $p(x)$ (see (2)), where $f(x)$ represents every element from the set $\mathbf{Z}_m[x]$, coincides with the set

$$\mathbf{Z}_m^s[x] = \left\{ A(x) = \sum_{j=0}^{s-1} a_j x^j \;\mid\; (a_0, a_1, \ldots, a_{s-1}) \in (\mathbf{Z}_m \times \ldots \times \mathbf{Z}_m) \right\}, \tag{3}$$

where m and s are the fixed positive integers; $m \geq 2$. The cardinality of the set (3) is equal to $N = |\mathbf{Z}_m^s[x]| = m^s$. Thus, the set $\mathbf{Z}_m^s[x]$ is a CSR

modulo $p(x)$. For the CSR of this type a special notation $\langle\cdot\rangle_{p(x)}$ is used, while the residue $r(x)$ modulo $p(x)$ over the polynomial $f(x)$ is designated as $\langle f(x)\rangle_{p(x)}$.

In the general case, the polynomial modular number system (PMNS) with pairwise relatively prime polynomial modules $p_1(x), p_2(x), \ldots, p_n(x)$ is induced by the isomorphic mapping

$$\phi:\ \langle\cdot\rangle_{P(x)} \to \langle\cdot\rangle_{p_1(x)} \times \langle\cdot\rangle_{p_2(x)} \times \ldots \times \langle\cdot\rangle_{p_n(x)},$$

where $P(x) = \prod\limits_{l=1}^{n} p_l(x)$. The isomorphism ϕ associates each polynomial $A(x) \in P(x)$ with the polynomial modular code (MC)

$$(a_1(x); a_2(x); \ldots; a_n(x)),$$

whose components are the residues $a_l(x) = \langle A(x)\rangle_{p_l(x)}$ $(l = 1, 2, \ldots, n)$ [6]. The set $\langle\cdot\rangle_{P(x)}$ is called the range of the PMNS.

The ring operations on polynomial modules $p_1(x), p_2(x), \ldots, p_n(x)$ over any two polynomials $A(x) = (a_1(x);\ a_2(x);\ \ldots;\ a_n(x))$ and $B(x) = (b_1(x);\ b_2(x);\ \ldots;\ b_n(x))$ $(a_l(x) = \langle A(x)\rangle_{p_l(x)}, b_l(x) = \langle B(x)\rangle_{p_l(x)}, l = 1, 2, \ldots, n)$ are executed independently for each module, i.e. according to the rule

$$\langle A(x) \circ B(x)\rangle =$$

$$\Big(\langle a_1(x) \circ b_1(x)\rangle_{p_1(x)};\ \langle a_2(x) \circ b_2(x)\rangle_{p_2(x)};\ \ldots;\ \langle a_n(x) \circ b_n(x)\rangle_{p_n(x)}\Big), \quad (4)$$

where $\circ \in \{+,\ -,\ \times\}$

Thus, both addition and multiplication of any two polynomials modulo $P(x)$ for their realizations require, respectively, n real additions and multiplications which can be executed in parallel in one modular clock tick. In the PMNS, all the operations (both modular (4) and non-modular) are performed in the ring $\mathbf{Z}_m$. This ring is called the scalar range or the numeric range of the PMNS. Decoding mapping assigning a polynomial $A(x)$ from the range $\langle\cdot\rangle_{P(x)}$ to a polynomial MC $(a_1(x),\ a_2(x),\ \ldots,\ a_n(x))$ is implemented by means of the Chinese reminder theorem [1, 7] which for the PMNS with modules $p_l(x)$ $(l = 1, 2, \ldots, n)$ gives

$$A(x) = \left\langle \sum_{l=1}^{n} P_l(x)\langle P_l(x)^{-1} A(x)\rangle_{p_l(x)} \right\rangle_{P(x)} =$$

$$= \sum_{l=1}^{n} P_l(x)\big\langle P_l(x)^{-1} A_l\big\rangle_{p_l(x)}, \quad (5)$$

where $P_l(x) = P(x)/p_l(x)$, $\left\langle P_l(x)^{-1}\right\rangle_{p_l(x)}$ is the residue which satisfies the equality $\left\langle P_l(x)\left\langle P_l(x)^{-1}\right\rangle_{p_l(x)}\right\rangle_{p_l(x)} = 1$.

The PMNS being the most appropriate for practical applications in the field of DSP have the polynomial modules $p_1(x), p_2(x), \ldots, p_n(x)$ which are the normalized polynomials of the first degree and the polynomial

$$P(x) = x^n - 1$$

has a factorization of the form

$$P(x) = \prod_{l=1}^{n}(x - r_l) \quad (r_l \in \mathbf{Z}_m; l = 1, 2, \ldots, n).$$

4. Implementation of discrete cyclic convolution in the PMNS

Let us consider the calculation of cyclic convolution of the sequences a_0, a_1, …, a_{n-1} and b_0, b_1, …, b_{n-1} according to formula (1). All the elements a_i and b_i $(i = 0, 1, \ldots, n-1)$ of the input sequences without loss of generality can be treated as integer variables taking values from the $\mathbf{Z}_m$. If the considered initial discrete sequences are presented in the polynomial form:

$$A(x) = a_{\nu-1}x^{\nu-1} + a_{\nu-2}x^{\nu-2} + \ldots + a_2x^2 + a_1x^1 + a_0$$

and

$$B(x) = b_{\nu-1}x^{\nu-1} + b_{\nu-2}x^{\nu-2} + \ldots + b_2x^2 + b_1x^1 + b_0,$$

then cyclic convolution operation can be reduced to the procedure of polynomials multiplication [8]. Thus, the realization of (1) is equivalent to the computation of the coefficients of the polynomial

$$C(x) = \sum_{\nu=0}^{n-1} c_\nu x^\nu = \left\langle \sum_{\nu=0}^{n-1} a_\nu x^\nu \sum_{\nu=0}^{n-1} b_\nu x^\nu \right\rangle_{x^n-1} = \langle A(x)\, B(x)\rangle_{x^n-1}\,.$$

The polynomials $A(x)$ and $B(x)$ are uniquely encoded in the PMNS, respectively, by sets of residues

$$A(x) = (a_1(x); a_2(x); \ldots; a_n(x)) \tag{6}$$

and

$$B(x) = (b_1(x); b_2(x); \ldots; b_n(x)) \tag{7}$$

corresponding to the selected polynomial modules $p_1(x)$, $p_2(x)$, …, $p_n(x)$ ($p_l(x) = x - r_l$ is the normalized polynomial of the first degree,

$$a_l(x) = \langle A(x)\rangle_{p_l(x)}\,;$$

$$b_l(x) = \langle B(x)\rangle_{p_l(x)}\,; \quad (l = 1, 2, \ldots, n).$$

The resultant polynomial

$$C(x) = (c_1(x); c_2(x); \ldots; c_n(x)) \tag{8}$$

is obtained by multiplying of polynomials (6) and (7) in the PMNS

$$C(x) = \langle A(x)\, B(x) \rangle_{P(x)}$$

where $P(x) = \prod_{l=1}^{n} p_l(x) = x^n - 1$.

In this case, the components of the set of residues $(c_1(x), c_2(x), \ldots, c_n(x))$ are the least residues of the division of products $a_l(x)\, b_l(x)$ by the corresponding polynomial modules

$$c_l(x) = \langle a_l(x)\, b_l(x) \rangle_{p_l(x)} \quad (l = 1, 2, \ldots, n)$$

Reconstruction of the positional code of the polynomial $C(x)$ by its polynomial MC (8) is carried out in accordance with formula (5).

5. Processing of polynomial residues in minimal redundant MC

It follows from formula (4) that the efficiency level of PMNS arithmetic depends not only on analytical form of the modules $p_l(x)$ $(l = 1, 2, \ldots, n)$ but also on the number system used for implementation of the computation over polynomial residues in the ring $\mathbf{Z}_m$. Since these calculations have a modular structure, then for encoding and processing of elements from the scalar range $\mathbf{Z}_m$ it is quite natural to use the real MNS with the modules m_1,m_2,...,m_k and the range $M_k = \prod_{i=1}^{k} m_i$ for number representation [1, 7]. In this approach, the parameter m is equal to M_k, i.e. the ring $\mathbf{Z}_{M_k} = \{0, 1, \ldots, M_k - 1\}$ is used as a numerical range of the PMNS. Such a PMNS with modular coding of elements of scalar range is called the polynomial-scalar MNS (PSMNS) [6].

Efficiency of computer arithmetic of PSMSS increases significantly when the minimal redundant modular coding of scalar elements is used. Minimal redundant encoding at the lower level allows us to optimize the execution of non-modular procedures [1, 7]. Such a PSMNS is called minimal redundant PSMNS. It is known that the principle of minimal redundant modular coding assumes that the set $\mathbf{Z}_{2M}^{-} = \{-M, -M+1, \ldots, M-1\}$ (where $M = \prod_{i=0}^{k-1} m_i$, $m_k \geq m_0 + k - 2, m_0 \geq k - 2$, m_0 is additional natural module) is used as a scalar range of the PSMNS instead of the range $\mathbf{Z}_{M_k}$ [1], [6] and [7].

Thus, in this case the minimal redundant PSMNS is defined by the set of pairwise relatively prime normalized polynomials of the first degree

$$p_l(x) = x - r_l,\ (r_l \in \mathbf{Z}^-_{2M}, l = 1, 2, \ldots, n)$$

and the set of pairwise relatively prime natural modules $m_1,\ m_2,\ \ldots,\ m_k$.

In accordance with the above, an arbitrary polynomial $A(x) \in \langle\cdot\rangle_{P(x)}$ in minimal redundant PSMNS is encoded by a set of residues

$$(\alpha_{1,1},\ \alpha_{1,2}, \ldots,\ \alpha_{1,k};\ \alpha_{2,1},\ \alpha_{2,2}, \ldots,\ \alpha_{2,k};\ \ldots;\ \alpha_{n,1},\ \alpha_{n,2}, \ldots,\ \alpha_{n,k}), \quad (9)$$

where $\alpha_{l,i} = |A_l|_{m_i}$; $A_l = \langle A(x)\rangle_{p_l(x)} = |A(r_l)|_{M_k}$; the value $|X|_m$ denotes the least non-negative residue of dividing X by natural modulo m,

$$l = 1, 2, \ldots, n; \quad i = 1, 2, \ldots, k.$$

Minimal redundant PSMNS are characterized by parallel structure both on the lower and upper levels of modular operations. In accordance with (4), the operations over any two polynomials $A(x)$ and $B(x)$ from the range $\langle\cdot\rangle_{P(x)}$ are executed by the rule

$$\begin{aligned}
&(\alpha_{1,1},\ \alpha_{1,2}, \ldots,\ \alpha_{1,k};\ \alpha_{2,1},\ \alpha_{2,2}, \ldots,\ \alpha_{2,k};\ \ldots;\ \alpha_{n,1},\ \alpha_{n,2}, \ldots, \alpha_{n,k})\circ\\
&\circ(\beta_{1,1},\ \beta_{1,2}, \ldots, \beta_{1,k};\ \beta_{2,1},\ \beta_{2,2}, \ldots, \beta_{2,k};\ \ldots;\ \beta_{n,1},\ \beta_{n,2}, \ldots, \beta_{n,k}) =\\
&\qquad = (\ |\alpha_{1,1} \circ \beta_{1,1}|_{m_1},\ |\alpha_{1,2} \circ \beta_{1,2}|_{m_2}, \ldots,\ |\alpha_{1,k} \circ \beta_{1,k}|_{m_k};\\
&\qquad |\alpha_{2,1} \circ \beta_{2,1}|_{m_1},\ |\alpha_{2,2} \circ \beta_{2,2}|_{m_2}, \ldots,\ |\alpha_{2,k} \circ \beta_{2,k}|_{m_k};\ \ldots\\
&\qquad |\alpha_{n,1} \circ \beta_{n,1}|_{m_1},\ |\alpha_{n,2} \circ \beta_{n,2}|_{m_2}, \ldots,\ |\alpha_{n,k} \circ \beta_{n,k}|_{m_k}\ ),
\end{aligned} \quad (10)$$

where $A_{l,i} = |A(r_l)|_{m_i}$ and $B_{l,i} = |B(r_l)|_{m_i}$ are the digits of polynomial-scalar modular codes of the operands $A(x)$ and $B(x)$, respectively (see (9)), $\circ \in \{+,\ -,\ \times\}$.

One of the main advantages of the PSMNS is the unique possibility to calculate the sum, difference and especially the product of two polynomials in accordance with (10) in one modular clock tick. Thus, in this system the multiplication of any two polynomials modulo $P(x) = x^n - 1$ for its implementation requires only n multiplications executed in parallel. In contrast, in the case of traditional arithmetic in positional number system the polynomials multiplication in the ring $\langle\cdot\rangle_{P(x)}$ requires the executions of $n(n-1)$ real additions and n^2 real multiplications.

It is quite clear that the efficiency of applied methods for conversion of polynomials from positional number system to PSMNS and vice versa, as well as execution of other non-modular operations, can have significant influence on real effect of introducing the polynomial MA in practice. This problem is successfully solved using a minimal redundant modular coding of scalars from the range $\mathbf{Z}^-_{2M}$ [1, 7].

At first, let us consider forming of digits of polynomial-scalar MC. In particular, for calculating the digits $\alpha_{l,i}$ (see (9)) of arbitrary polynomial $A(x) = \sum\limits_{\nu=0}^{n-1} a_\nu x^\nu$ from the range $\langle\cdot\rangle_{P(x)}$ ($a_\nu \in \mathbf{Z}_{2M}^-$) the following formula is used

$$\alpha_{l,i} = \left| \sum_{\nu=0}^{n-1} R_{\nu,\,l,\,i} \sum_{s=0}^{\mu-1} \left| F_s(a_\nu{}^{(s)}) \right|_{m_i} \right|_{m_i},$$

where $R_{\nu,\,l,\,i} = |r_l^\nu|_{m_i}$; $F_s(a_\nu{}^{(s)})$ are the additive components of λ-bit positional forms of coefficients a_ν:

$$a_\nu = \sum_{t=0}^{\lambda-1} a_{\nu,t} 2^t - a_{\nu,\,\lambda-1} 2^\lambda = \sum_{t=0}^{\lambda-2} a_{\nu,t} 2^t - a_{\nu,\,\lambda-1} 2^{\lambda-1} = \sum_{s=0}^{\mu-1} F_s(a_\nu^{(s)})$$

defined according to formulas

$$a_\nu^{(s)} = \sum_{t=0}^{\lambda_s-1} a_{\nu,\,q_s+t}\, 2^t \quad (s = 0, 1, \cdots, \mu-1)$$

$$F_s(a_\nu^{(s)}) = \begin{cases} a_{\nu,s}\, 2^{\,q_s} & \text{if } \ s = 0, 1, \cdots, \mu-2, \\ a_{\nu,\,\mu-1}\, 2^{\,q_{\mu-1}} - \lfloor a_{\nu,\,\mu-1} \,/\, 2^{\,q_{\mu-1}-1} \rfloor\, 2^\lambda & \text{if } \ s = \mu-1; \end{cases}$$

$q_0 = 0, q_1, \cdots, q_{\mu-1}$ is the increasing sequence of integer values that specifies the partition of the binary additional code $(a_{\nu,\,\lambda-1}, a_{\nu,\,\lambda-2}, \ \cdots, a_{\nu,0})_2$ on $\mu \geq 1$ groups, sth of which contains $\lambda_s = q_{s+1} - q_s$ bits, $q_{\mu-1} \leq \lambda - 1$, $q_\mu = \lambda$ [1, 9]. Here $\lfloor y \rfloor$ denotes the integer part of a real number y.

In order to restore the positional representation of the polynomial $A(x)$ by its minimal redundant polynomial-scalar MC (9) at first it is necessary to compute the minimal redundant MC $(\alpha_1^{(\nu)},\ \alpha_2^{(\nu)}, \cdots\ \alpha_k^{(\nu)})$ of the coefficient a_ν for every $\nu = 0, 1, \cdots, n-1$:

$$\alpha_i^{(\nu)} = |a_\nu|_{m_i} = \left| \sum_{l=1}^{n} R_{l,\,i}^{(\nu)}\, \alpha_{l,i} \right|_{m_i} \qquad (i = 1, 2, \cdots, k),$$

where $R_{l,\,i}^{(\nu)} = \left| n^{-1}\, r_l^{-\nu} \right|_{m_i}$ [6]. After that, the positional code of the coefficient a_ν can be formed by its MC in accordance with the formula

$$a_\nu = \sum_{i=1}^{k-1} M_{i,k-1} \left| M_{i,k-1}^{-1} \alpha_i^{(\nu)} \right|_{m_i} + I(a_\nu) M_{k-1},$$

where $M_{i,\,k-1} = M_{k-1}/m_i$, $M_{k-1} = \prod_{j=1}^{k-1} m_j$, $I(a_\nu)$ is the interval index of integer a_ν defined by the following calculating expressions [1, 7]

$$I(a_\nu) = \begin{cases} \hat{I}_k(a_\nu) \ \text{ if } \ \hat{I}_k(a_\nu) < m_0, \\ \hat{I}_k(a_\nu) - m_k \ \text{ if } \ \hat{I}_k(a_\nu) > m_k - m_0 - k + 2; \end{cases}$$

$$\hat{I}_k(a_\nu) = \left| \sum_{i=1}^{k} R_{i,k}(\alpha_i^{(\nu)}) \right|_{m_k}$$

$$R_{i,k}(\alpha_i^{(\nu)}) = \left| \frac{\left| M_{i,k-1}^{-1}\, \alpha_i^{(\nu)} \right|_{m_i}}{M_{k-1}} \right|_{m_k} \quad (i \neq k); \quad R_{k,k}(\alpha_k^{(\nu)}) = \left| \frac{\alpha_k^{(\nu)}}{M_{k-1}} \right|_{m_k}.$$

Thus, the efficiency of the PSMNS computer arithmetic is significantly increased due to the optimization of the non-modular procedures when using the minimal redundant coding at the lower level [1, 7]. Therefore, the minimal redundant PSMNS potentially takes the priority position in the field of computer applications.

The proposed developments allow us to create effective DSP systems using the minimally redundant PSMNS with sufficiently simple implementation. In these systems at the upper level the normalized polynomials of the first degree is used as a bases, whereas at the lower level the elements of scalar range is represented in minimal redundant MC.

References

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

[2] P. Kornerup, D. W. Matula, *Finite Precision Number Systems and Arithmetic,* Cambridge University Press, Cambridge, 2010.

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

[4] A. Schönhage, V. Strassen, *Schnelle Multiplikation Großer Zahlen*, Computing, **7**, (1971), 281–292.

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

[6] M. Selyaninov, *Modular technique of high-speed parallel computing on the sets of polynomials*, Scientific Issues, Jan Długosz University of Częstochowa, Ser. Mathematics, **XVII**, (2012), 69–76.

[7] M. Selyaninov, *Modular technique of parallel information processing*, Sci. Iss., Jan Długosz University of Częstochowa, Mathematics, **XIII**, (2008), 43–52.

[8] A. H. Diaz-Perez, R. Domingo, *Cyclic Convolution Algorithm Formulations Using Polynomial Transform Theory*, Journal of Computers, **2**, No. 7, (2007), 40–48.

[9] M. Selianinau, *Modular principles of high-speed adaptive filtration of discrete signals*, Scientific Issues, Jan Długosz University of Częstochowa, Ser. Technical and IT Education, **VI**, (2011), 75–84.
[10] M. Selyaninov, *Modular number systems in the complex plane*, Scientific Issues, Jan Długosz University of Częstochowa, Ser. Mathematics, **XV**, (2010), 131–138.
[11] M. Selyaninov, *Arithmetic of quadratic minimal redundant modular number systems*, Scientific Issues, Jan Długosz University of Częstochowa, Ser. Mathematics, **XVI**, (2011), 129–134.

Received: April 2014

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