UDC 621.391

V.V. Vasuta, V.M. Kurchanov, I.O. Chernitska

Poltava national technical Yuri Kondratyuk university, Poltava

ANALYSIS OF METHOD OF INCREASING PERFORMANCE PRODUCTIVITY OF MODULAR OPERATIONS BASED ON NON-POSITION ENCODING OF NUMBERS IN RESIDUE CLASS

In the article, four principles of arithmetical operations implementation in the modular arithmetic are considered. Analysis of method of increasing performance productivity of arithmetic operations in residue class based on circular shift register is considered. The analysis shows the high efficiency of the residue class implementation of arithmetic operations, and can significantly improve parameters of computers in comparison with machines that are built on the same physical and technological base, but in positional numeral system.

Keywords: *residue class system, positional numeral system, the circular shift register.*

Introduction

Currently, scientific and technical computing tasks require significant amounts of calculations carried out in real time operation of the data processing system.

Positional numeral system (PNS) used in modern computing systems, in which information is submitted and processed, has significant drawbacks. Arithmetic operations are very time-consuming and operating devices are unreliable. This is due to "strong" number position bonds. In this case, partial improvement of workflow is possible due to improving computer hardware.

The main problem of building a computing system is to ensure its reliable, fault-tolerant and long-term operation. This problem is especially important in areas where the error or failure of the system can lead to terrible and even fatal consequences. Focusing on addressing the performance and increase of the reliability of the source data, researchers concluded that under the use of PNS, acceleration of operations and improvement of their reliability is almost impossible.

The results showed that one of the most promising and effective ways of increasing productivity, reliability and accuracy of computational tools is the development and implementation of computer arithmetic using theoretical concepts of some sections of number theory. This is a so-called modular system of calculation, particularly non-positional numeral system in residue class.

The idea that goes back to Euler's classic works and Gauss's theory of divisibility and residues was relevant in 1954 in the USSR while forming non-positional numeral system. We know that assumptions of this system are based on the theory of congruencies that underlies the theory of numbers. The first works on the theory of congruencies date back to the heyday of Chinese culture. Chinese theorem on the solution of linear congruencies is widespread in the classical theory of numbers and is crucial for building non-positional representations of numbers.

Its main properties are the following.

1. Independence of residue numbers presented in residue class (RC). This makes it possible to build data-processing system as a set of independent computing channels.

2. Equality of residue numbers. This structure promotes the synthesis of ultra-reliable data-processing system.

3. Low number position of residues in RC. This can significantly improve the performance of arithmetic operations by using table arithmetic.

There are four principles of arithmetic operations in modular arithmetic:

- adder principle (based on low number position binary adders);
- table principle (based on permanent storage devices ROM);
- direct logical principle of arithmetic operations based on the description of modular operations at the level of function switches with the help of which binary bit value of resulting residues is formed;
- circular shift principle, based on the use of circular shift registers.

Low number position of residues a_i enables us to implement NRS arithmetic operations, either based on low number position binary adders or in table form. While using the first method of implementing arithmetic operations the same disadvantage as in PNS appears (though to a much lesser extent), it is availability of between number position links within a given bases m_i NRS. In table form of arithmetic operations there are no number position links between processing operands in general, however, for a sufficiently large word length computer (for large modules of NRS) the amount of equipment operating devices dramatically increases. It is important and urgent to consider implementing interim version of NRS arithmetic operations based on application of the principle of the circular shift using circular shifting registers (CSR). In the current imple-

mentation of the principle of arithmetic operations, circular shift principle (CSP) was defined, its distinctive feature is that the result of an arithmetic operation $(a_i \pm \beta_i) \bmod m_i$ on an arbitrary module of RCS, given by a set $\{m_j\}$, $j = \overline{1,n}$ of bases is determined only by consistent circular shift of a given digital structure.

Main part

Indeed, the well-known theorem of Cayley determines isomorphism between the elements of Abelian group and the elements of permutation. In this case, matrix assembly for an arbitrary module m_i of RCS will be given in Table 1 (for $m_i = 5$ – Table 2)

Table 1

Cayley table for arbitrary notion of m_i

β_i	a_i				
	0	1	2	3	4
0	0	1	2	…	m_i-1
1	1	2	3	…	0
2	2	3	4	…	1
…	…	…	…	…	…
m_i-1	m_i-1	0	1	…	m_i-2

Table 2

Cayley table for arbitrary notion of $m_i = 5$

β_i	a_i				
	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

One consequence of Cayley's theorem is the conclusion that the display of elements of an Abelian group on groups of all integers is homomorphic. This allows us to determine the outcome of arithmetic operations in RCS using CSP. As operand A in RCS is represented by a set of residues from dividing it by a set n of prime (generally prime in pairs) numbers $\{m_j\}$, $j = \overline{1,n}$, this set of residues can be identified directly with the amount of n Galois fields $\sum_{i=1}^{n} GF(m_i)$. To study the method of implementation of arithmetic operations in RCS it is enough to consider the option for an arbitrary Galois finite field $GF(m_i)$ at $i = \text{const}$ i.e. for a concrete reduced residue system on module m_i.

Suppose for a given transaction of module addition $(a_i \pm \beta_i) \bmod m_i$, a Cayley table is created in the field $GF(n_i)$ (Table 1). Due to the existence of a neutral element in the field $GF(m_i)$, a conclusion can be made that in the Table 3.1 there is a line in which elements of the field are in ascending order and due to the fact that in the field of residues $GF(m_i)$ these elements are different (order of group is equal m_i), it is considered that each row (column) of the Table 3.1 contains all the elements of the field once. Using these properties of a Cayley table allows you to implement transactions of module addition and subtraction of RCS by application of CSP with the help of n circular $M = ([\log(m_i - 1)] + 1)$ - CSR bit.

Let an arbitrary algebra system be represented as $S = \langle G, * \rangle$ where G is a non-empty set;

$*$ is a type of operation specified for any two elements. Operation • + of addition of a set of residue classes R, generated by the ideal J, creates a new circle called the circle of residue classes of R/J. It can be represented as $z/(m_i)$ where z is a set of integers $0 \pm 1 \ldots 2$; m_i is the basis of RCS.(If the base of RCS m_i is a prime number, then $z/(m_i)$ is a field). This fact, as already noted, makes possible the realization of arithmetic operation of addition of RCS without interbit transfers by circular shift using CSR.

Based on the principle formulated in CSP, let us consider the method of realization of arithmetic operations in residue class. The method is that digital output structure for each of the modules (bases) of RCS is in the form of content of the first row (column) of the table of modular addition (subtraction) $(a_i \pm \beta_i) \bmod m_i$ represented by

$$P_{вих}^{(m_i)} = \left[P_0(\alpha_0) \| P_1(\alpha_1) \| \ldots \| P_{m_i-1}(\alpha_{m_i-1}) \right], \quad (1)$$

where $\|$ is operation of concatenation;

$P_v(\alpha_v) - k$ is bit binary code corresponding to the value of α_v residue of $\alpha_v = \overline{0, m_i - 1}$ number by module.

For a given module $m_i = 5$ digital structure will be as follows

$$P_{вих}^{(5)} = \left[000 \| 001 \| 010 \| 011 \| 100 \right].$$

Thus, using circular shift registers that are widely applied in the PNS, it is easy to implement arithmetic operations of RCS, and based on the formula (1), degrees of cyclical permutations are defined by the following formula:

$$\left[P_0(\alpha_0) \| P_1(\alpha_1) \| \ldots \| P_{m_i-1}(\alpha_{m_i-1}) \right]^{+Z} = \left[\begin{matrix} P_z(\alpha_z) \| P_{z+1}(\alpha_{z+1}) \| P_{m_i-1}(\alpha_{m_i-1}) \| \\ P_0(\alpha_0) \| \ldots \| P_{z-1}(\alpha_{z-1}) \end{matrix} \right], \quad (2)$$

$$\left[P_0(\beta_0) \| P_1(\beta_1) \| \ldots \| P_{m_i-1}(\beta_{m_i-1}) \right]^{-Z} = \left[\begin{matrix} P_{m_i-1-z}(\alpha_{m_i-1-z}) \| P_{m_i-z}(\alpha_{m_i-z}) \| \ldots \| P_0(б_0) \| \\ P_1(б_1) \| P_{m_i-z-2}(\alpha_{m_i-z-2}) \end{matrix} \right]. \quad (3)$$

Let us indicate that

$$\left[P_0(\alpha_0)\|P(\alpha_1)\|...\|P_{m_i-1}(\alpha_{m_i-1})\right]^{m_i}=\varepsilon$$

i.e. in $z = m_i$ all elements of the ordered set $\{P_j(\alpha_j)\}(j=\overline{0,m_i-1})$ remain in the original position. During the technical implementation of this method first operand a_i indicates the number α_{α_i} category $P_{a_i}(\alpha_{a_i})$ of CRS determining the result of the modular operation by module m_i and the second operand β_i of CRS determines the number of bits ($\beta_i \cdot k$ is bit) that are necessary to offset the original content of CRS according to the algorithm (2) (3). Suppose $m_{i=5}$ $(S=\langle\{0,1,2,4\},\oplus\rangle)$. Then the table of values of module amounts $(a_i \oplus \beta_i) \bmod m_i$ for a circle of residue class of z/(5) will be presented in a matrix (Table 2). Content categories of CSR will be presented in the form of numerical data, for example, the first line (column) of Table 2 (Figure 1 a). On the figure, the sign $\oplus$ marks positive (clockwise) direction shift of content categories of CSR. The first operand a_i indicates the number of the category, the content of which determines the outcome of this operation; the second operand β_i indicates the number of shifts of content categories of CSR (Fig. 1, b, c).

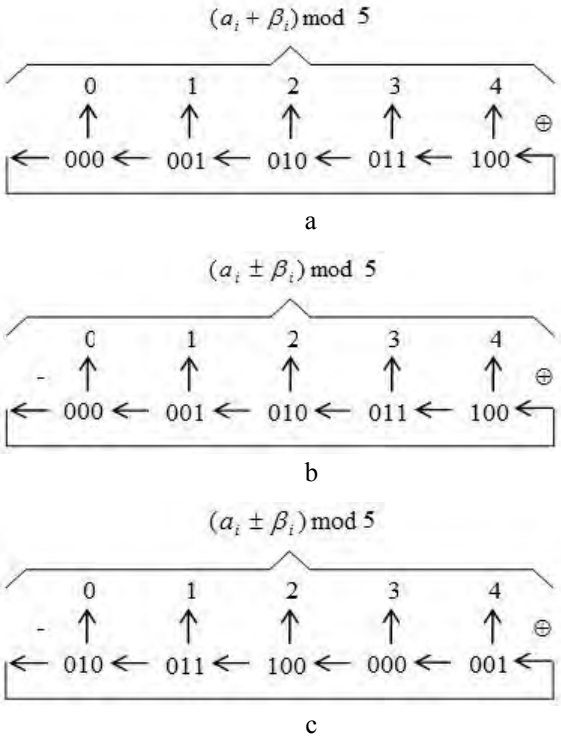

Fig. 1. Options of the circular shift register
a – 1st option; b – 2nd option; c – 3rd option

Let us introduce the concept of an operator of circular shift. OCS is a statement that defines the size (expressed as a number of CSR bits shifted) and direction of CSR bit shift and is represented as $k^{(z)}$.

Thus, for modular addition operation OCS will be presented in the form of $k^{(+\beta_i)}$ and the time shift t_c (basically the time of t operation) of content CSR bits is presented by the formula

$$t_c = k \cdot \tau \cdot z \qquad (4)$$

(further let us assume that $t \approx t_c$), where $k=\left[\log_2(m_n-1)\right]+1$ (m_n is module for running circuit modular addition); $\tau -$ is the time of one binary bit shift (one trigger operation time).

Based on CSP using the identical equation

$$(\alpha_i-\beta_i)\equiv\left[a_i+(m_n-\beta_i)\right]\bmod m_n \qquad (5)$$

we can implement modular subtraction operation $(a_i-\beta_i)\bmod m_n$.

In this case, OCS looks like $k^{(+(m_n-\beta_i))}$.

Apparently, the advantage of CSP compared to methods based on the use of binary adders, is the lack of interbit transfers, which significantly increases the authenticity of modular operations. However, it takes a long time to implement modular operations (see formula 4), which reduces overall efficiency of the use of computers in residue class. This circumstance necessitates the development of algorithms to improve performance productivity of these operations in the computer.

Algorithm of improving performance productivity of modular addition (subtraction) operation is to use identical equation (2) and the following formula:

$$a_i+\beta_i=\beta_i+a_i,$$

$$a_i+(m_n-\beta_i)=(m_n-\beta_i)+a_i.$$

In this case, for the operation of modular addition $(a_i+\beta_i)\bmod m_n$ OCS is represented as

$$z=\begin{cases}+a_i, \text{if } a_i \le \beta_i,\\ +\beta_i, \text{if } a_i \rangle \beta_i,\end{cases}$$

i.e. in $a_i \le \beta_i$ operand a_i determines the number of z shifts of CDR content and operand β_i is CDR bit number that determines the result of operations; in $a_i \rangle \beta_i$ operand β_i determines the number of z shifts of CDR content and operand a_i is CDR bit number that defines the result of the operation.

The operation of module subtraction $(a_i-\beta_i)\bmod m_n$ OCS is represented as

$$z=\begin{cases}+a_i, \text{if } a_i \le (m_n-\beta_i),\\ +\beta_i, \text{if } a_i \rangle (m_n-\beta_i),\end{cases}$$

i.e. in $a_i \le (m_n-\beta_i)$ operand a_i determines the number of z shifts of CDR content category and operand $(m_n-\beta_i)$ is CDR bit number that determines the outcome of an operation; in $a_i \rangle (m_n-\beta_i)$ operand $(m_n-\beta_i)$ determines the number of shifts of CDR content category and operand a_i is CDR bit number that determines the outcome of an operation.

This algorithm of modular operation implementation allows to decrease the time of module addition and subtraction operation.

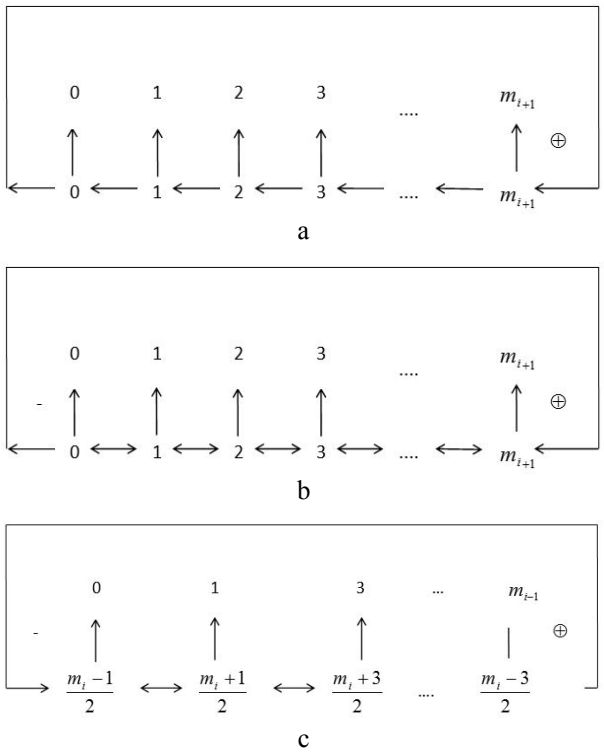

Fig. 2. Options of the circular shift register for module m_1 of RCS: a – 1st option; b – 2nd option; c – 3rd option

One of the algorithms to improve performance productivity of modular addition (subtraction) operation is an algorithm based on the properties of the following identical equation:

$$(a_i + \beta_i) = [a_i - (m_i - b_i)] \bmod m_i . \qquad (6)$$

CDR shift can be carried out both in positive and in negative directions (for $m_i = 5$, Figure 1, b), where for the operation of module addition OCS is represented as

$$z = \begin{cases} +\beta_i, \text{if } 0 \le \beta_i \le (m_n - 1)/2, \\ -(m_n - \beta_n), \text{if } (m_n + 1)/2 \le \beta_i \le m_n - 1. \end{cases}$$

The use of this algorithm allows 90% reduction in the value z, which reduces the time t of modular operations performance (Fig. 2, b).

Conclusions

Thus, the system of residue classes can significantly improve parameters of computers in comparison with machines that are built on the same physical and technological base, but in positional numeral system, and get new, up-to-date design and structural solutions.

Literature

1. Акушский И.Я. Машинная арифметика в остаточных классах / И.Я. Акушский, Д.И. Юдицкий. – М.: Сов. радио, 1968. – 440 с.

2. Краснобаев В.А. Методы повышения надежности специализированных ЭВМ систем и средств связи / В.А. Краснобаев. – Х.: МО СССР, 1990. – 172 с.

3. Виноградов И.М. Основы теории чисел / И.М. Виноградов. – М.: Наука, 1981. – 175 с.

4. Краснобаев В.А. Принцип реализации арифметических операций в системе остаточных классов / В.А. Краснобаев // АСУ и приборы автоматики. –1988. – С. 36-38.

5. Краснобаев В.А. Методы реализации модульных операций в системах цифровой обработки информации / В.А. Краснобаев // Радиотехника. – 2001. – Вып. 119. – С. 130-134.

Надійшла до редколегії 29.04.2016

Рецензент: д-р техн. наук, проф. В.А. Краснобаєв, Харківський національний університет ім. В.Н. Каразіна, Харків.

АНАЛІЗ МЕТОДУ ПІДВИЩЕННЯ ПРОДУКТИВНОСТІ ВИКОНАННЯ МОДУЛЬНИХ ОПЕРАЦІЙ НА ОСНОВІ НЕПОЗИЦІЙНОГО КОДУВАННЯ ЧИСЕЛ У КЛАСІ ЗАЛИШКІВ

В.В. Васюта, В.М. Курчанов, І.О. Черницька

У статті розглядається чотири принципи реалізації арифметичних операцій в модулярній системі числення. Розглядається аналіз методу підвищення продуктивності реалізації арифметичний операцій в класі залишків на основі кільцевих регістрів зсуву. Проведений аналіз свідчить про високу ефективність використання класу залишків при реалізації модульних операцій і дозволяє істотно поліпшити параметри обчислювальних машин в порівнянні з машинами, які побудовані на тій же фізико-технологічній базі, але в позиційній системі числення.

Ключові слова: *система залишкових класів, позиційна система числення, регістри кільцевого зсуву.*

АНАЛИЗ МЕТОДА ПОВІШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ ВЫПОЛНЕНИЯ МОДУЛЬНЫХ ОПЕРАЦИЙ НА ОСНОВЕ НЕПОЗИЦИОННОО КОДИРОВАНИЕ ЧИСЕЛ В КЛАССЕ ОСТАТКОВ

В.В. Васюта, В.Н. Курчанов, І.А. Черницкая

В статье рассматривается четыре принципа реализации арифметических операций в модулярной системе счисления. Рассматривается анализ метода повышения производительности реализации арифметический операций в классе остатков на основе кольцевых регистров сдвига. Проведенный анализ свидетельствует о высокой эффективности использования класса вычетов при реализации модульных операций и позволяет существенно улучшить параметры вычислительных машин по сравнению с машинами, которые построены на той же физико-технологической базе, но в позиционной системе счисления.

Ключевые слова: *система остаточных классов, позиционная система исчисления, регистры кольцевого сдвига.*