УДК 681.04

В.А. КРАСНОБАЕВ, Я.В. ИЛЮШКО

Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Украина

МЕТОДЫ ОБРАБОТКИ ИНФОРМАЦИИ В СИСТЕМЕ ОСТАТОЧНЫХ КЛАССОВ

Предложены методы сравнения чисел в системе остаточных классов. На основе предложенных методов, основанных на принципе получения и сравнения унитарного однорядного кода, рассмотрены эффективные алгоритмы обработки информации при проведении арифметического и алгебраического сравнения чисел в системе остаточных классов.

система остаточных классов (СОК), модулярный код, позиционный код, однорядный унитарный код, модульные операции, операнды в СОК, сравнение чисел в СОК

1. Постановка задачи

Известно, что использование непозиционной системы счисления в остаточных классах (СОК) значительно повышает надежность и производительность ЭВМ [1–7]. Однако необходимость в определении позиционных характеристик чисел в СОК снижает общую эффективность применения модулярных кодов. Существующие методы определения позиционных характеристик, в частности методы сравнения чисел в СОК, обладают существенными недостатками, главным из которых является необходимость преобразования операндов из СОК в позиционную систему счисления и обратно, что снижает пользовательскую производительность и надежность ЭВМ.

В статье предлагаются методы обработки информации, которые используются при сравнении чисел в СОК и основаны на представлении и обработке специального однорядного унитарного кода без непосредственного преобразования сравниваемых операндов из модулярного кода (кода СОК) в позиционный код и обратно.

Пусть СОК задана упорядоченными, т.е.

$$\mathbf{m_i < m_{i+1}},$$

взаимно попарно простыми натуральными числами (основаниями)

$$\mathbf{m_1, m_2, \ldots, m_n},$$

а сравниваемые операнды представлены в виде

$$\mathbf{A = (a_1, a_2, \ldots, a_n)}, \quad \mathbf{B = (\beta_1, \beta_2, \ldots, \beta_n)}.$$

При этом предполагается, что исходные операнды лежат в соответствующих интервалах

$$\left[\frac{j_1 M}{m_n}, \frac{(j_1+1)M}{m_n}\right) \text{ и } \left[\frac{j_2 M}{m_n}, \frac{(j_2+1)M}{m_n}\right),$$

где $M = \prod_{i=1}^{n} m_i$, а номер $\mathbf{j_k + 1}$ интервала определяется известным выражением

$$j_k = \gamma_n \overline{m}_n (\operatorname{mod} m_n),$$

где величина $\mathbf{m_n}$ определяется из решения сравнения $\overline{m}_n M/m_n \equiv 1(\operatorname{mod} m_n)$. При $\mathbf{j_1 \neq j_2}$ операция арифметического сравнения может быть реализована сравнением номеров интервалов, а именно: если $\mathbf{j_1 > j_2}$, то $\mathbf{A > B}$; если $\mathbf{j_1 < j_2}$, то $\mathbf{A < B}$. При $\mathbf{j_1 = j_2}$ определяется номер $\mathbf{j_3 + 1}$ интервала $\left[\frac{j_3 M}{m_n}, \frac{(j_3+1)M}{m_n}\right)$, в котором расположено число $\mathbf{A - B}$. Если $\mathbf{0 \le j_3 < (m_n + 1)/2}$, то $\mathbf{A < B}$, а если $\frac{m_n+1}{2} \le j_3 < m_n$, то $\mathbf{A > B}$.

Метод арифметического сравнения чисел в СОК предполагает преобразование чисел к виду $\mathbf{A^{(H)} = (0, 0, .., \gamma_n)}$, что требует $\mathbf{n - 1}$ тактов операции нулевизации [1]. Кроме этого необходимо выполнить позиционное сравнение номеров $\mathbf{(j_1 + 1)}$ и $\mathbf{(j_2 + 1)}$ интервалов попадания исходных операндов **A** и **B**. Все это усложняет алгоритм сравнения и увеличивает время сравнения чисел, что приводит к необходимости разработки эффективных методов сравнения операндов в СОК, не требующих определения позиционных характеристик.

2. Метод арифметических параллельных вычитаний

Будем рассматривать сравниваемые операнды в следующих числовых интервалах:

$$[jm_i, (j+1)\ m_i),$$

где $\overline{j=1,\ N\text{-}1}$ $\left(N = \prod_{\substack{k=1 \\ k \neq i}}^{n} m_k\right)$.

Вначале исходные операнды A, B приводятся к числам, кратным $\mathbf{m_i}$, путем модульного вычитания следующего вида:

$$A_{m_i} = A - a_i = \left(a_1^{(i)}, a_2^{(i)}, \dots, a_{i-1}^{(i)}, 0, a_{i+1}^{(i)}, \dots, a_n^{(i)}\right),$$

$$B_{m_i} = B - \beta_i = \left(\beta_1^{(i)}, \beta_2^{(i)}, \dots, \beta_{i-1}^{(i)}, 0, \beta_{i+1}^{(i)}, \dots, \beta_n^{(i)}\right),$$

где $a_i = (a_1', a_2', \dots, a_i, a_n')$, $\beta_i = (\beta_1', \beta_2', \dots, \beta_i, \beta_n')$.

Далее посредством набора констант **0**, $\mathbf{m_i}$, $\mathbf{2m_i}$, …, $\mathbf{(N-1)\ m_i}$, представленных по $\mathbf{(n-1)}$-м основаниям СОК $\mathbf{m_1}$, $\mathbf{m_2}$, …, $\mathbf{m_{i-1}}$, $\mathbf{m_{i+1}}$, …, $\mathbf{m_n}$, осуществляется построение однорядного кода соответственно в виде:

$$K_n^{(n_A)} = \{z_N z_{N-1} \dots z_2 z_1\},$$

$$z_{n_A} = 0\ \left(z_l = 1; l = \overline{1, N},\ 1 \neq n_A\right);$$

$$K_N^{(n_B)} = \{z_N' z_{N-1}' \dots z_2' z_1'\},$$

$$z_{n_B}' = 0\ \left(z_l' = 1, l = \overline{1, N},\ 1 \neq n_B\right).$$

Непосредственно алгоритм построения однорядного кода можно представить следующим образом [8]:

$$\begin{cases} A_{m_i} - 0 = z_1, \\ A_{m_i} - m_i = z_2, \\ A_{m_i} - 2m_i = z_3, \\ \dots\dots\dots\dots \\ A_{m_i} - \left(\prod_{\substack{\rho=1 \\ \rho \neq i}}^{n-1} m_\rho - 1\right) m_i = z_N. \end{cases} \qquad (1)$$

$$\begin{cases} B_{m_i} - 0 = z_1', \\ B_{m_i} - m_i = z_2', \\ B_{m_i} - 2m_i = z_3', \\ \dots\dots\dots\dots \\ B_{m_i} - \left(\prod_{\substack{\rho=1 \\ \rho \neq i}}^{n-1} m_\rho - 1\right) m_i - z_N. \end{cases} \qquad (2)$$

В этом случае имеем:

$$z_{n_A} = 0 \text{ при } A_{m_i} - n_A m_i = 0;$$

$$z_{n_A} = 1 \text{ при } A_{m_i} \neq n_A m_i;$$

$$z_{n_B}' = 0 \text{ при } B_{m_i} - n_B m_i = 0;$$

$$z_{n_B}' = 1 \text{ при } B_{m_i} \neq n_B m_i.$$

Геометрически данный метод сравнения можно пояснить следующим образом. Интервал $\left[0,\ \prod_{i=1}^{n} m_i\right)$ разбивается на $\prod_{\substack{\rho=1 \\ \rho \neq i}}^{n-1} m_\rho$ отрезков. Исходные операнды **A** и **B** путем вычитания констант (1), (2) смещают на левый край интервала их попадания. Это равнозначно приведению сравниваемых операндов к числам, кратным основанию $\mathbf{m_i}$ СОК. Исходя из этого, точность **W** сравнения операндов зависит от величины основания $\mathbf{m_i}$, т.е. $\mathbf{W = W(m_i)}$. Однако при максимальной точности сравнения $\mathbf{W_{max} = W(m_{min})}$ (для упорядоченной СОК $\mathbf{W_{max} = W(m_1)}$) резко возрастает количество оборудования для технических устройств, реализующих операцию сравнения в СОК. Действительно, количество оборудования $\mathbf{N_0}$ устройства сравнения в СОК значительно зависит от количества сумматоров $\mathbf{N_1}$, осуществляющих операцию параллельного вычитания (1), (2).

При $\mathbf{W_{max} = W(m_1)}$ имеем $N_1 = N_{max} = \prod_{i=2}^{n} m_i$,

а при $\mathbf{W_{min} = W(m_n)}$ – $N_1 = N_{min} = \prod_{i=1}^{n-1} m_i$.

Таким образом, необходимость обеспечения высокой степени точности сравнения требует значительного количества оборудования, что снижает эффективность применения сравнивающих устройств в СОК. Данное обстоятельство обуславливает актуальность и важность поисков более эффективных методов сравнения чисел в СОК, обеспечивающих высокую точность W сравнения при приемлемом количестве $\mathbf{N_0}$ оборудования сравнивающих устройств.

В общем виде поставленную задачу сформулируем следующим образом. Необходимо найти $\mathbf{N_0 = min}$ при $\mathbf{W_{max}}$, т.е. $\mathbf{N_0(W_{max}) = min}$. Как было показано выше, $\mathbf{W_{max} = W(m_i)}$. При этом изменение величины основания $m_i\left(i = \overline{1,n}\right)$ влияет только на количество $\mathbf{N_1}$ оборудования группы сумматоров.

В этом случае правомерно сформулировать задачу в виде определения:

$$\mathbf{N_1(W_{max}) = min}. \qquad (3)$$

При максимальной точности сравнения операндов A и B (равной единице длины интервала попадания чисел) будем иметь $\mathbf{W_{max} = W(m_i = 2)}$. Однако в этом случае количество оборудования группы сумматоров равно $\mathbf{N_1(W_{max}) = max}$, что не удовлетворяет условию (3). С учетом того, что $\mathbf{N_1(W_{min}) = min}$, возникает задача по разработке нового метода сравнения операндов, который в максимальной степени удовлетворял бы условию (3).

3. Метод параллельных вычитаний со сравнением остатков по основанию $\mathbf{m_n}$ системы остаточных классов

Введем операцию сравнения непосредственно остатков $\mathbf{a_n}$, $\boldsymbol{\beta_n}$ по основанию $\mathbf{m_n}$. В данном случае результат этого сравнения одновременно с результатом сравнения однорядного кода $K_N^{(n_A)}$ и $K_N^{(n_B)}$ будет определяться с максимальной точностью $\mathbf{W_{max}}$ и при минимальном количестве оборудования $\mathbf{N_{min}}$. Алгоритм определения результата операции арифметического сравнения может быть представлен следующим образом [9 – 12]:

$$\begin{cases} \text{если } n_A > n_B, \text{ то } A > B; \\ \text{если } n_A < n_B, \text{ то } A < B; \\ \text{если } n_A = n_B \text{ и при этом} \\ \begin{cases} a_n = \beta_n, \text{ то } A = B, \\ a_n > \beta_n, \text{ то } A > B, \\ a_n < \beta_n, \text{ то } A < B. \end{cases} \end{cases} \qquad (4)$$

Совокупность соотношений (4) представляет собой общий алгоритм реализации операции арифметического сравнения операндов в СОК. Целесообразно рассмотреть примеры конкретного выполнения операции арифметического сравнения чисел в СОК. Пусть СОК задана основаниями $\mathbf{m_1 = 2}$, $\mathbf{m_2 = 3}$ и $\mathbf{m_3 = 5}$. Кодовые слова приведены в табл. 1. В табл. 2 заданы константы $\mathbf{a_n}$ ($\boldsymbol{\beta_n}$), представленные в заданной СОК, а в табл. 3 по $\mathbf{(n - 1)}$-м основаниям СОК $m_i\left(i = \overline{1, n-1}\right)$ заданы константы однорядного кода вида

$$\Delta \cdot m_n \left(\Delta = \overline{0, \prod_{i=1}^{n-1} m_i} \right).$$

Пусть сравниваемые операнды представлены в виде $\mathbf{A_{23} = (1,10,011)}$ и $\mathbf{B_{21} = (1,00,001)}$. В этом случае по значениям констант (табл. 2) определяем значения $A_{m_n} = A_{23} - a_n = (0,10,000)$,

Таблица 1

Таблица кодовых слов

A	A в СОК			A	A в СОК		
	m_1 = 2	**m_2 = 3**	**m_3 = 5**		**m_1 = 2**	**m_2 = 3**	**m_3 = 5**
0	0	00	000	15	1	00	000
1	1	01	001	16	0	01	001
2	0	10	010	17	1	10	010
3	1	00	011	18	0	00	011
4	0	01	100	19	1	01	100
5	1	10	001	20	0	10	000
6	0	00	001	21	1	00	001
7	1	01	010	22	0	01	010
8	0	10	011	23	1	10	011
9	1	00	100	24	0	00	100
10	0	01	000	25	1	01	000
11	1	10	001	26	0	10	001
12	0	00	010	27	1	00	010
13	1	01	011	28	0	01	011
14	0	10	100	29	1	10	100

$B_{m_n} = B_{21} - \beta_n = (01{,}00{,}000)$, что соответствует сдвигу операндов **A** и **B** на левый край интервала [20, 25). Далее посредством констант однорядного кода (табл. 3) определяем однорядный код для входных операндов в виде

$$K_N^{(n_A)} = K_6^{(4)} = \{110111\};$$

$$K_N^{(n_B)} = K_6^{(n_B)} = K_6^{(4)} = \{110111\},$$

где $N_1 = \prod_{i=1}^{n-1} m_i = 6$.

Таблица 2

Таблица констант нулевизации

γ_3	Константы
000	(0, 00, 000)
001	(1, 01, 001)
010	(0, 10, 010)
011	(1, 00, 011)
100	(0, 01, 100)

Одновременно с этим посредством **($n = [\log_2(m_n - 1)] + 1$)**-разрядной схемы сравнения, параллельно во времени определяется результат сравнения: **$a_n = 011 > \beta_n = 001$**. Так как **$n_A = n_B = 4$**, то в соответствии с алгоритмом (4) определяем, что **$A_{23} > B_{21}$**.

Пусть сравниваемые операнды представлены в виде **$A_{23} = (1{,}10{,}011)$**, **$B_3 = (1{,}00{,}011)$**. В этом случае по значениям констант (табл. 2) определяем следующие разности:

$$A_{m_n} = A_{23} - a_n = (0{,}10{,}000)$$

и $$B_{m_n} = B_3 - b_n = (1{,}00{,}000),$$

что соответствует сдвигу операнда **A_{23}** на левый край интервала [20, 25), а операнда **B_3** – на левый край интервала [0, 5). Далее посредством констант однорядного кода (табл. 3) определяем однорядный код для рассматриваемых входных операндов A_{23}^4 и **B_3**: $K_N^{(n_A)} = K_6^{(5)} = \{101111\}$, $K_N^{(n_B)} = K_6^{(1)} = \{111110\}$.

Таблица 3

Таблица констант однорядного унитарного кода

(0 ÷N - 1) m_n **(Δ·5)**	СОК		Номер позиции нуля
	m_1 = 2	**m_2 = 3**	
0 **m_3 = 0**	0	00	1
1 **m_3 = 5**	1	10	2
2 **m_3 = 10**	0	01	3
3 **m_3 = 15**	1	00	4
4 **m_3 = 20**	0	10	5
5 **m_3 = 25**	1	01	6

Так как $\mathbf{n_A = 5 > n_B = 1}$, то в соответствии с алгоритмом (4) определяем, что $\mathbf{A_{23} > B_3}$ [13].

4. Метод и алгоритм сравнения с константой разности

Рассмотрим эффективный метод и алгоритм реализации операции арифметического сравнения операндов в СОК. Суть данного алгоритма состоит в том, что производится сравнение не непосредственно операндов **A** и **B**, а величин их разности $\boldsymbol{\chi = (A - B) \bmod M = (\gamma_1, \gamma_2, \ldots, \gamma_n)}$ и основания $\mathbf{m_1}$.

В этом случае определяется значение

$$\chi_{m_1} = \chi - \gamma_1 = (0, \gamma'_2, \gamma'_3, \ldots, \gamma'_n),$$

где константы $\gamma_1 = (\gamma_1, \gamma'_2, \gamma'_3, \ldots, \gamma'_n)$ и $\boldsymbol{\gamma_1 = (a_1 - \beta_1) \bmod m_1}$ представлены в заданной СОК.

Тогда общий алгоритм сравнения операндов представим так:

$$\begin{cases} A > B \ \text{при} \ \chi_{m_1} \le m_1; \\ A < B \ \text{при} \ \chi_{m_1} > m_1; \\ A = B \ \text{при} \ \chi_{m_1} = 0. \end{cases} \tag{5}$$

Посредством набора констант **0**, $\mathbf{m_1}$, $\mathbf{2m_1}\ldots, (N-1)\cdot m_1 \ \left(N = \prod_{i=1}^{n} m_i\right)$, представленных в СОК с основаниями $\mathbf{m_2}$, $\mathbf{m_3}$, …, $\mathbf{m_n}$, осуществляется построение однорядного кода в виде

$$K_n^{(n_\chi)} = \{z_N, z_{N-1}, \ldots, z_2, z_1\},$$

где

$$\begin{cases} \chi_{m_1} - 0 = z_1, \\ \chi_{m_1} - m_1 = z_2, \\ \chi_{m_1} - 2m_1 = z_3, \\ \ldots\ldots\ldots \\ \chi_{m_1} - (N-1)m_1 = z_n. \end{cases} \tag{6}$$

При этом

$$z_{n_\chi} = 0 \quad \text{для} \quad \chi_{m_1} - n_\chi m_1 = 0,$$

$$z_{n_\chi} = 1 \quad \text{для} \quad \chi_{m_1} - n_\chi m_1 \ne 0.$$

Первый модуль $\mathbf{m_1}$ СОК также представляется однорядным кодом длиной **N** двоичных разрядов, в котором на втором месте справа будет ноль $\mathbf{(m_1{-}1, m_1{=}0)}$, а на остальных – единицы, т.е. однорядный унитарный код будет представлен как

$$K_N^{(2)} = \{11\ldots101\}.$$

Далее известными методами в соответствии с алгоритмом (5) сравниваются операнды **A и B**, представленные однорядным кодом.

Для вышеприведенной СОК рассмотрим пример реализации данного метода (табл. 1). Пусть **A = (0,01,010)** и **B = (1,10,011)**. Определим значение

$$\chi = (A - B)\mathrm{mod}M = [(0-1)\mathrm{mod}2,\ (01-10)\mathrm{mod}3,\ (010-011)\mathrm{mod}5] = (1,10,100).$$

По значению γ_1 **= 1** определим константу в виде γ_1 **= (1,01,001)** (табл. 4).

Таблица 4

Константы нулевизации

γ_1	Константы
00	(0,00,000)
01	(1,01,001)

Далее проводим операцию:

$\chi_{m_1} = \chi - \gamma = (00,01,011)$. Операнд χ_{m_i}, кратный значению модуля $m_1 = 2$, поступает на первые входы соответствующих сумматоров, на вторые входы которых поступают соответствующие константы (табл. 5).

Так как $\chi_{m_i} - 14_{m_1} = 0$, то однорядный код примет вид

$$K_N^{(n_\chi)} = K_{15}^{(14)} = \{011111111111111\}.$$

В соответствии с алгоритмом (5) определяем, что **A < B**.

Достоинство рассмотренного алгоритма заключается в обеспечении максимальной точности сравнения при приемлемом количестве оборудования устройства для его реализации.

5. Метод и общий алгоритм алгебраического сравнения чисел

Перейдем от реализации операции арифметического сравнения к алгебраическому.

В этом случае сравниваемые операнды **A** и **B** имеют по одному дополнительному знаковому разряду, т.е. число сопровождается признаком

$\Omega_A(\Omega_B)$ знака **sign A (sign B)**,

где

$$\Omega_A(\Omega_B) = \begin{cases} 0, \text{если } A(B) \ge 0, \\ 1, \text{если } A(B) < 0. \end{cases}$$

Таким образом, сравниваемые операнды представляются в виде

Таблица 5

Схема построения однорядного кода

(0 ÷N - 1) m_1 (Δ·2)	СОК		Номер позиции нуля
	$m_2 = 3$	$m_3 = 5$	
0 $m_1 = 0$	00	000	1
1 $m_1 = 2$	10	010	2
2 $m_1 = 4$	01	100	3
3 $m_1 = 6$	00	001	4
4 $m_1 = 8$	10	011	5
5 $m_1 = 10$	01	000	6
6 $m_1 = 12$	00	010	7
7 $m_1 = 14$	10	100	8
8 $m_1 = 16$	01	001	9
9 $m_1 = 18$	00	011	10
10 $m_1 = 20$	10	000	11
11 $m_1 = 22$	01	010	12
12 $m_1 = 24$	00	100	13
13 $m_1 = 26$	10	001	14
14 $m_1 = 28$	01	011	15

$$A' = (\Omega_A; A) = \{\Omega_A; (a_1, a_2, \ldots, a_n)\},$$

$$B' = (\Omega_B; B) = \{\Omega_B; (\beta_1, \beta_2, \ldots, \beta_n)\},$$

а алгоритм сравнения операндов A' и B' представим в таком виде:

$$\begin{cases}
\text{если } n_A > n_B \begin{cases} \Omega_A = 0, \Omega_B = 0, \text{ то } A' > B', \\ \Omega_A = 0, \Omega_B = 1, \text{ то } A' > B', \\ \Omega_A = 1, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 1, \Omega_B = 1, \text{ то } A' > B'; \end{cases} & (7) \\
\text{если } n_A < n_B \begin{cases} \Omega_A = 0, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 0, \Omega_B = 1, \text{ то } A' > B', \\ \Omega_A = 1, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 1, \Omega_B = 1, \text{ то } A' > B'; \end{cases} & (8) \\
\text{если } n_A = n_B \begin{cases} a_n = \beta_n \begin{cases} \Omega_A = 0, \Omega_B = 0, \text{ то } A' = B', \\ \Omega_A = 0, \Omega_B = 1, \text{ то } A' > B', \\ \Omega_A = 1, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 1, \Omega_B = 1, \text{ то } A' = B'; \end{cases} \\ a_n > \beta_n \begin{cases} \Omega_A = 0, \Omega_B = 0, \text{ то } A' > B', \\ \Omega_A = 0, \Omega_B = 1, \text{ то } A' > B', \\ \Omega_A = 1, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 1, \Omega_B = 1, \text{ то } A' < B'; \end{cases} \\ a_n < \beta_n \begin{cases} \Omega_A = 0, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 0, \Omega_B = 1, \text{ то } A' > B', \\ \Omega_A = 1, \Omega_B = 0, \text{ то } A' < B', \\ \Omega_A = 1, \Omega_B = 1, \text{ то } A' > B'. \end{cases} \end{cases} & (9)
\end{cases}$$

Совокупность выражений (7) – (9) определяет общий алгоритм алгебраического сравнения чисел в СОК (рис. 1), где $P_{г_A}, P_{г_B}$ – входные регистры; ПОК$_1$, ПОК$_2$ – схемы определения показателя унитарного однорядного кода соответственно чисел А и В; СС$_1$ – схема сравнения значений n_A и n_B; СС$_2$ – схема сравнения величин остатков a_n и $в_n$; ОЗ – схема сравнения знаковых признаков Ω_A и Ω_B; Р$_г$Б и Р$_г$Н – регистры соответственно наибольшего и наименьшего чисел.

6. Сравнительный анализ эффективности

Проведем сравнительный анализ времени реализации операции сравнения двух чисел $\mathbf{A_{СОК}}$ и $\mathbf{B_{СОК}}$ для предложенного метода и наиболее известного [1]. Сущность известного метода состоит в переводе чисел $\mathbf{A_{СОК}}$, $\mathbf{B_{СОК}}$ из системы остаточных классов в позиционную систему счисления $\mathbf{A_{ПСС}}$, $\mathbf{B_{ПСС}}$ и дальнейшем сравнении операндов $\mathbf{A_{ПСС}}$ и $\mathbf{B_{ПСС}}$. Причем перевод чисел из СОК в ПСС осуществляется в соответствии с выражением

$$A_{ПСС} = \left| \sum^{n} a_i B_i \right|_M,$$

где $a_i = [A_{ПСС}/m_i] m_i$; $\mathbf{B_i}$ – ортогональный базис по m_i-му основанию СОК. Для известного и предлагаемого методов сравнения чисел время реализации определится соответствующими математическими соотношениями:

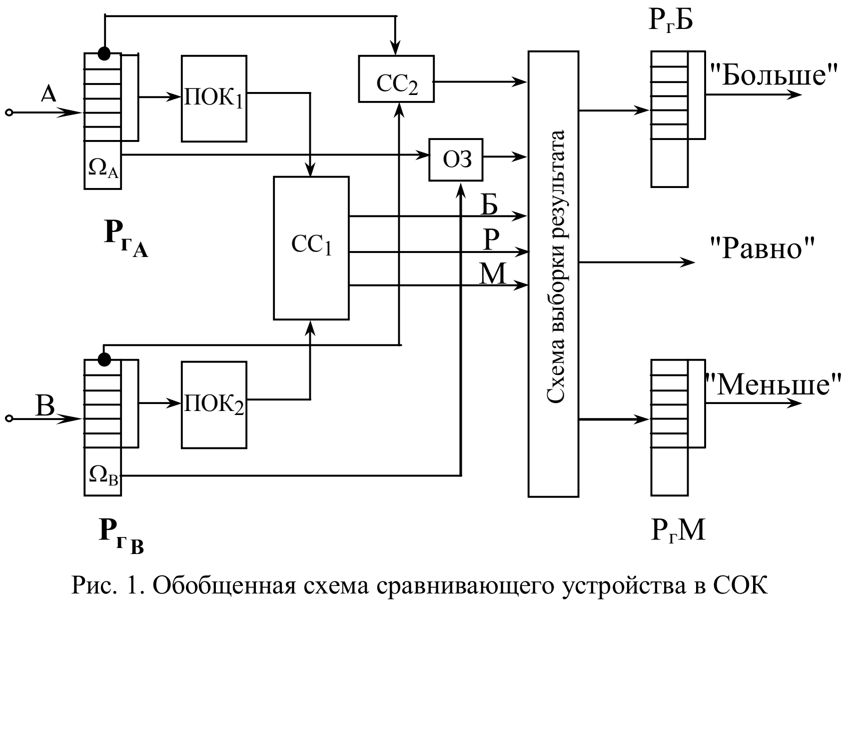

Рис. 1. Обобщенная схема сравнивающего устройства в СОК

$$T_{СОК}^{(1)} = t^{(11)} + t^{(12)} + t^{(13)} + t^{(14)}, \qquad (10)$$

$$T_{СОК}^{(2)} = t^{(21)} + t^{(22)} + t^{(23)}, \qquad (11)$$

где $\mathbf{t^{(11)}}$ – время реализации операции умножения для максимального $\mathbf{m_n}$ по величине модуля СОК (время реализации операции умножения двух ($\mathbf{K = [log_2(m_n–1)]+1}$)-разрядных двоичных операндов);

$\mathbf{t^{(12)}}$ – время $\mathbf{(n–1)}$-го сложения типа $a_iB_i + a_{i+1}B_{i+1}\left(i=\overline{1,n-1}\right)$;

$\mathbf{t^{(13)}}$ – время определения вычета числа $\mathbf{A_{ПСС}}$ ($\mathbf{B_{ПСС}}$);

$\mathbf{t^{(14)}}$ – время сравнения позиционных операндов $\mathbf{A_{ПСС}}$ и $\mathbf{B_{ПСС}}$;

$\mathbf{t^{(21)}}$ – время реализации операции вычитания в СОК $\mathbf{A_{СОК}} – a_i$;

$\mathbf{t^{(22)}}$ – время реализации операции вычитания в СОК $A_{m_i} - c \cdot m_i$;

$\mathbf{t^{(23)}}$ – время сравнения позиционного **N**-разрядного однорядного унитарного кода двух операндов.

Известно, что время сложения t_c и t_y умножения двух операндов в ПСС определяется следующими соотношениями: $\mathbf{t_c = \tau(2\rho - 1)}$ и $\mathbf{t_y = 2\tau\rho^2}$, где ρ – разрядность обрабатываемых операндов; τ – время «сдвига» одного двоичного разряда. При этом время «срабатывания» логического элемента И (ИЛИ) определяется как

$$t_И \approx t_{ИЛИ} \approx \tau/2, \text{ а } t^{(14)} \approx t^{(23)} \approx 6t_n \text{ и}$$

$$t^{(13)} \approx 2\tau\left\{\left[\log_2\left(\prod_{i=1}^{n} m_i - 1\right)\right]+1\right\}^2.$$

С учетом вышеизложенного соотношения (10) и (11), соответственно, представляются в виде

$$T_{СОК}^{(1)} = 2\tau\kappa^2 + (n-1)\,\tau\,(2k-1) + \\ +2\tau\left\{\left[\log_2\left(\prod_{i=1}^{n} m_i - 1\right)\right]+1\right\}^2 + 3\tau; \qquad (12)$$

$$T_{СОК}^{(2)} = 2\tau + 2\tau + 3\tau = 7\tau. \qquad (13)$$

В соответствии с выражениями (12) и (13) рассчитаны значения $T_{СОК}^{(1)}$, $T_{СОК}^{(2)}$ (табл. 6) для различных l-байтовых разрядных сеток ЭВМ $(l=\overline{1,4})$.

Таблица 6

Данные времени реализации операций сравнения

T	*l*			
	1	2	3	4
$T_{СОК}^{1}$	324	870	1916	3334
$T_{СОК}^{2}$	7	7	7	7

Очевидно (табл. 6) следующее: с увеличением длины машинного слова ЭВМ, что характерно для современной тенденции развития систем и средств переработки цифровой информации, эффективность применения принципа однорядного кодирования, по сравнению с практически существующими, возрастает.

Заключение

В статье рассмотрены эффективные методы и алгоритмы обработки информации при проведении операций арифметического и алгебраического сравнения чисел в СОК. Представленные методы основаны на принципе получения и сравнения унитарного однорядного кода. На основе предложенных методов описаны алгоритмы для их реализации, в соответствии с которыми в дальнейшем был разработан класс патентоспособных устройств для выполнения операций арифметического и алгебраического сравнения чисел в СОК.

Результаты, полученные в данной статье, целесообразно использовать, в первую очередь, в системах и устройствах обработки цифровой информации при решении задач цифровой фильтрации, при выполнении криптографических преобразований в полях Галуа, при разработке криптографических систем на основе использования преобразований Хартли, для построения матричных и векторных процессоров, а также при необходимости реализации модульных операций в системах цифровой обработки информации различного назначения.

Литература

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

2. Торгашев В.А. Система остаточных классов и надежность. ЦВМ. – М.: Сов. радио, 1973. – 118 с.

3. Жихарев В.Я., Юнес Эль Хандасси, Краснобаев В.А. Пути повышения производительности и отказоустойчивости ЭВМ // Открытые информационные и компьютерные интегрированные технологии. – Х.: НАКУ «ХАИ». – 2003. – Вып. 19. – С. 269–282.

4. Жихарев В.Я., Юнес Эль Хандасси, Краснобаев В.А. Методы и алгоритмы реализации арифметических операций в классе вычетов // Открытые информационные и компьютерные интегрированные технологии. – Х.: НАКУ «ХАИ». – 2003. – Вып. 20. – С. 84–101.

5. Краснобаев В.А. Основы создания вычислителей на основе остаточных классов // Системи обробки інформації. – Х.: НАНУ, ПАНМ, ХВУ. – 2001. – Вип. 1(11). – С. 3–7.

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

7. Краснобаев В.А. Метод и алгоритмы коррекции ошибок в системах цифровой обработки информации // Радиотехника. – 2002. – Вып. 126. – С. 231 – 237.

8. Краснобаев В.А. Методы арифметического сравнения чисел, представленных кодом системы остаточных классов // АСУ и приборы автоматики. – 1987. – Вып. 84. – С. 74–76.

9. Устройство для сравнения чисел в системе остаточных классов: А.с. № 1037244 СССР / В.А. Краснобаев, Е.И. Бороденко, А.И. Бецков. Опубл. 1983, Бюл. № 31.

10. Устройство для сравнения чисел в системе остаточных классов: А.с. № 1224803 СССР / В.И. Долгов, В.А. Краснобаев, А.В. Брезгунов. Опубл. 1986, Бюл. № 18.

11. Устройство для сравнения чисел в системе остаточных классов: А.с. № 1427358 СССР / В.А. Краснобаев, И.Д. Горбенко, М.А. Гальцев. Опубл. 1988, Бюл. № 36.

12. Устройство для сравнения в системе остаточных классов: А.с. № 1552171 СССР / В.А. Краснобаев, О.Н. Фоменко, В.П. Ирхин. Опубл. 1990, Бюл. № 11.

13. Устройство для сравнения чисел в системе остаточных классов: А.с. № 1145338 СССР / В.А. Краснобаев. Опубл. 1985, Бюл. № 10.

14. Краснобаев В.А. Искусственный интеллект и система счисления в остаточных классах // Пробл. бионики. – 1987. – Вып. 39. – С. 53 – 58.

15. Краснобаев В.А., Удников А.Н. Выбор системы счисления при проектировании отказоустойчивых ЭВМ // ИУСЖТ. – 2002. – №1. – С.27 – 29.

16. Применение методов искусственного интеллекта в управлении проектами / Н.М. Бабынин, В.Я. Жихарев, В.М. Илюшко; Под ред. д-ра техн. наук А.Ю. Соколова. – Х.: НАКУ «ХАИ». – 2002. – 474 с.

Поступила в редакцию 11.05.04

Рецензент: д-р техн. наук, проф., акад. Академии высшей школы Украины И.А. Фурман, Харьковский государственный технический университет сельского хозяйства, г. Харьков