УДК 681.04

В. А. КРАСНОБАЕВ, д-р техн. наук

МЕТОДЫ СРАВНЕНИЯ ОПЕРАНДОВ В СИСТЕМЕ ОСТАТОЧНЫХ КЛАССОВ

В настоящее время известно, что использование непозиционной системы счисления в остаточных классах (СОК) значительно повышает надежность и производительность ЭВМ [1, 2]. Одновременно 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 данной статье предлагаются методы сравнения чисел в СОК, основанные на представлении и обработке специального однорядового унитарного кода без непосредственного преобразования сравниваемых операндов из модулярного кода (кода СОК) в позиционный код и обратно.

Известные методы сравнения чисел в СОК предполагают определение номера $j + 1$ интервала попадания операндов [1].

Пусть задана СОК упорядоченными ($m_i < m_{i+1}$) взаимно попарно простыми натуральными числами (основаниями) $m_1, m_2, \ldots, m_n$ и пусть сравниваемые операнды представлены в виде

$$A = (a_1, a_2, \ldots, a_n), \qquad B = (\beta_1, \beta_2, \ldots, \beta_n).$$

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

Данный метод арифметического сравнения чисел в СОК предполагает преобразование чисел к виду $A^{(H)} = (0, 0, .., \gamma_n)$, что требует $n - 1$ тактов операции нулевизации [1]. Кроме этого необходимо произвести позиционное сравнение номеров $(j_1 + 1)$ и $(j_2 + 1)$ 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.

Метод параллельных вычитаний

Будем рассматривать сравниваемые операнды в произвольных интервалах

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

Исходные операнды A, B приводятся к числам, кратным m_i, путем модульного вычитания следующего вида

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

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

где:

$$a_i = \left(a_1', a_2', \ldots, a_i, a_n'\right); \quad \beta_i = \left(\beta_1', \beta_2', \ldots, \beta_i, \beta_n'\right).$$

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

$$K_n^{(n_A)} = \left\{z_N z_{N-1} \ldots z_2 z_1\right\},\ z_{n_A} = 0\ \left(z_l = 1; l = \overline{1,N},\ l \neq n_A\right),$$
$$K_N^{(n_B)} = \left\{z'_N z'_{N-1} \ldots z'_2 z'_1\right\},\ z'_{n_B} = 0\ \left(z'_l = 1, l = \overline{1,N},\ l \neq n_B\right).$$

Алгоритм построения однорядового кода в СОК можно представить следующим образом [3]:

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

$$\begin{cases} B_{m_i} - 0 = z'_1, \\ B_{m_i} - m_i = z'_2, \\ B_{m_i} - 2m_i = z'_3, \\ \ldots\ldots\ldots\ldots \\ B_{m_i} - \left(\prod\limits_{\substack{\rho=1\\\rho\neq i}}^{n-1} m_\rho - 1\right) m_i - z_N. \end{cases} \tag{2}$$

В этом случае имеем: $z_{n_A} = 0$ при $A_{m_i} - n_A \cdot m_i = 0$,

$z_{n_A} = 1$ при $A_{m_i} \neq n_A \cdot m_i$,

$z'_{n_B} = 0$ при $B_{m_i} - n_B \cdot m_i = 0$,

$z'_{n_B} = 1$ 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 смещают путем вычитания констант (1), (2) на левый край интервала их попадания. Это равнозначно приведению сравниваемых операндов к числам, кратным основанию m_i СОК. Исходя из этого, точность **W** сравнения операндов зависит от величины основания m_i, т.е. $W = W(m_i)$. Однако при максимальной точности сравнения $W_{max} = W(m_{min})$ (для упорядоченной СОК $W_{max} = W(m_1)$) резко возрастает количество оборудования для технических устройств, реализующих операцию сравнения в СОК. Действительно, количество оборудования N_0 устройства сравнения в СОК значительно зависит от количества сумматоров N_1, осуществляющих операцию параллельного вычитания (1), (2).

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

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

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

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

В этом случае правомерно сформулировать задачу в виде определения

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

Очевидно, что при максимальной точности, равной единице длины интервала, $W_{max} = W(m_i = 2)$. Однако в этом случае $N_1(W_{max}) = \max$ и этот результат не удовлетворяет поставленному условию (3). С другой стороны – $N_1(W_{min}) = \min$. Таким образом, необходимо разработать такой метод сравнения, который в максимальной степени удовлетворял бы условию (3).

Метод параллельных вычитаний со сравнением

Если ввести операцию сравнения и непосредственно остатков a_n, β_n по основанию m_n, то результат этого сравнения, одновременно с результатом сравнения однорядового кода $K_N^{(n_A)}$ и $K_N^{(n_B)}$, будет определяться с максимальной точностью W_{max} и при минимальном количестве оборудования N_{min}. Алгоритм определения результата операции арифметического сравнения может быть представлен следующим образом [4]:

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

Совокупность соотношений (4) представляет общий алгоритм реализации операции арифметического сравнения операндов в СОК. Целесообразно рассмотреть примеры конкретного выполнения операции арифметического сравнения чисел в СОК. Пусть СОК задана основаниями $m_1 = 2$, $m_2 = 3$ и $m_3 = 5$. Кодовые слова приведены в табл. 1. В табл. 2 заданы константы a_n (β_n), представленные в заданной СОК, а в табл. 3 по (n – 1)-му основаниям СОК $m_i\left(i = \overline{1, n-1}\right)$ заданы константы однорядового кода вида $\Delta \cdot m_n\left(\Delta = \overline{0, \prod_{i=1}^{n-1} m_i}\right)$.

Пусть сравниваемые операнды представлены в виде $A_{23} = (1, 10, 011)$ и $B_{21} = (1, 00, 001)$. В этом случае по значениям констант (табл. 2) определяем значения $A_{m_n} = A_{23} - a_n = (0, 10, 000)$, $B_{m_n} = B_{21} - \beta_n = (01,00,000)$, что соответствует сдвигу операндов A и B на левый край интервала [20, 25). Далее посредством констант однорядового кода (табл. 3) определяем однорядовый код для входных операндов в виде

$$K_N^{(n_A)} = K_6^{(4)} = \{110111\}, \quad K_N^{(n_B)} = K_6^{(n_B)} = K_6^{(4)} = \{110111\}, \text{ где } N_i = \prod_{i=1}^{n-1} m_i = 6.$$

Таблица 1

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

Таблица 2

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

Таблица 3

$(0 \div N-1)\ m_n$ $(\Delta \cdot 5)$	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

Одновременно с этим посредством $(n = [\log_2(m_n - 1)] + 1)$-й 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:

$$A_{m_n} = A_{23} - a_n = (0, 10, 000) \quad \text{и} \quad B_{m_n} = B_3 - \beta_n = (1,00,000),$$

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 однорядового кода (табл. 3) определяем однорядовый код для рассматриваемых входных операндов A_{23} и B_3:

$$K_N^{(n_A)} = K_6^{(5)} = \{101111\}, \qquad K_N^{(n_B)} = K_6^{(1)} = \{111110\}.$$

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

Проведем сравнительный анализ времени реализации операции сравнения двух операндов $A_{СОК}$ и $B_{СОК}$ для предложенного метода и наиболее известного [1]. Сущность известного метода состоит в переводе чисел $A_{СОК}$, $B_{СОК}$ из системы остаточных классов в позиционную систему счисления $A_{ПСС}$, $B_{ПСС}$ и дальнейшего сравнения операндов $A_{ПСС}$ и $B_{ПСС}$. Причем перевод чисел из СОК в ПСС производится в соответствии с выражением $A_{ПСС} = \left| \sum_{i}^{n} a_i B_i \right|_M$, где $a_i = [A_{ПСС}/m_i]m_i$; B_i – ортогональный базис по m_i основанию СОК. Для известного и предлагаемого методов сравнения чисел время реализации определится соответствующими математическими соотношениями

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

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

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

$t^{(12)}$ – время (n – 1)-го сложения типа $a_iB_i + a_{i+1}B_{i+1}\left(i = \overline{1, n-1}\right)$;
$t^{(13)}$ – время определения вычета числа $A_{ПСС}$ ($B_{ПСС}$);
$t^{(14)}$ – время сравнения позиционных операндов $A_{ПСС}$ и $B_{ПСС}$;
$t^{(21)}$ – время реализации операции вычитания в СОК $A_{СОК} - a_i$;
$t^{(22)}$ – время реализации операции вычитания в СОК $A_{m_i} - c \cdot m_i$;
$t^{(23)}$ – время сравнения позиционного N-разрядного однорядового унитарного кода двух операндов.

Известно, что время сложения t_c и t_y умножения двух операндов в ПСС определяется следующими соотношениями: $t_c = \tau(2\rho - 1)$ и $t_y = 2\tau\rho^2$, 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

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

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

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

Таблица 4

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

Из табл. 4 видно, что с увеличением длины машинного слова ЭВМ, что характерно для современных систем и средств переработки цифровой информации, эффективность применения принципа однорядового кодирования, по сравнению с практически существующими, возрастает.

Таким образом, рассмотрены эффективные методы сравнения операндов в СОК. Данные методы основаны на получении и сравнении унитарного однорядового кода. На основе предложенных методов описаны алгоритмы для их реализации, в соответствии с которыми разработан класс патентноспособных устройств для выполнения операции сравнения чисел в СОК. Результаты проведенных исследований могут быть использованы в системах и устройствах обработки цифровой информации при решении задач цифровой фильтрации, криптографических преобразованиях в полях Галуа, при разработке криптографических систем на основе использования преобразований Хартли, а также при необходимости реализации модульных преобразований.

Список литературы: 1. *Акушский И.Я., Юдицкий Д.И.* Машинная арифметика в остаточных классах. М.: Сов. радио, 1968. 440 с. 2. *Краснобаев В.А.* Основы создания вычислителей на основе остаточных классов // Системи обробки інформації / НАНУ; ПАНМ; ХВУ. Харків, 2001. Вип. 1(11). С. 3 – 7. 3. *Краснобаев В.А.* Методы арифметического сравнения чисел, представленных кодом системы остаточных классов // АСУ и приборы автоматики . 1987. Вып. 84. С. 74 – 76. 4. *Краснобаев В.А.* Методы сравнения чисел, представленных кодом системы остаточных классов // Электрон.моделирование. 1988. № 2. С. 84 – 87.

Харьковский национальный университет радиоэлектроники

Поступила в редколлегию 15.04.2003