Н.И. Червяков, Лобес М.В.

Россия, Ставрополь, Ставропольский Государственный Университет

dechernov@yndex.ru

ЦЕЛОЧИСЛЕННОЕ ДЕЛЕНИЕ В СИСТЕМЕ ОСТАТОЧНЫХ КЛАССОВ

В работе рассмотрены алгоритмы деления в системе остаточных классов на основе методов Ферма и Ньютона. Определены диапазоны их эффективного применения.

In the work realization algorithms of division in Residue Number Systems based on methods Ferma and Newton. Their diapasons are showed of effective application.

Деление в СОК относится к немодульным операциям и является одной из важнейших, так как лежит в основе многих других операций и входит в состав различных вычислительных алгоритмов. Как и в позиционной системе счисления, операция деления в СОК представляет собой наиболее сложную и длинную операцию. Деление в СОК обычно рассматривается в трех вариантах: деление без остатка, масштабирование (деление на один или несколько модулей системы), деление произвольных чисел [1,2].

Наибольший интерес представляет последний случай деления, при котором на делимое и делитель не накладывается ни каких ограничений, кроме того, что они целые. Однако до недавнего времени этот случай деления в СОК являлся трудно реализуемым.

В работе [3] предложен алгоритм деления целых чисел в СОК на основе метода спуска Ферма.

Допустим, что необходимо разделить число a на число b (a и b целые положительные числа) и $p_1, p_2, ..., p_n$ - модули СОК.

Первым этапом этого алгоритма является выбор приблизительного делителя $\bar{b}$. Для этого делитель b представляется в обобщенной позиционной системе счисления (ОПСС) в порядке уменьшаемой значимости по основаниям $p_1, p_2, ..., p_n$, которые совпадают с модулями СОК, то есть в виде

$$b = b_n \cdot \prod_{i=1}^{n-1} p_i + b_{n-1} \cdot \prod_{i=1}^{n-2} p_i + ... + b_2 \cdot p_1 + b_1. \qquad (1)$$

Далее определяется приблизительный делитель $\bar{b}$ по формуле

$$\bar{b} = Q \cdot \prod_{i=1}^{k-1} p_i, \qquad (2)$$

где Q можно найти путем использования наиболее значимой ненулевой цифры в ОПСС представлении делителя b. Эту ненулевую цифру заменяют ближайшим модулем или произведением модулей СОК так чтобы выполнялось условие

$$b \le \bar{b} < 2b. \qquad (3)$$

Метод нахождения $\bar{b}$, удовлетворяющего условию (3) рассмотрен в [4].

Вторым этапом алгоритма является определение округленного частного $q = \lfloor a/b \rceil$. Для этого сначала определяется $q_1 = \lfloor a/\bar{b} \rceil$. Найденное таким образом значение q_1 далее используется в рекурсивных соотношениях

$$a_i = a_{t-1} - bq_i, a_0 = a \text{ и } q_i = \lfloor a_{i-1}/\bar{b} \rceil \qquad (4)$$

для получения q_2, q_3 и так далее. Эта повторяющаяся процедура продолжается до тех пор, пока $q_i = 0$, либо до $a_i = 0$. Если это возникает на r - ом повторении, то

$$q = \lfloor a/b \rceil = \sum q_i + q', \qquad (5)$$

где $q' = \begin{cases} q_r, \textit{если} \quad q_r \neq 0 \textit{ и } a_r = 0; \\ 1, \textit{если} \quad q_r = 0 \textit{ и } a_{r-1} \geq b \textit{ для} \quad \textit{любых } \bar{b} \neq 0; \\ 0, \textit{во} \quad \textit{всех} \quad \textit{остальных} \quad \textit{случаях}. \end{cases}$

Так как $\bar{b}$ является или модулем или произведением модулей СОК, то нахождение величины q_i представляет собой масштабирование, которое может быть выполнено эффективным методом, описанным в работе [5] на основе представления ортогональных базисов в ОПСС.

Рассмотрим алгоритм на примере.

Пример 1. Пусть $p_1 = 47, p_2 = 43, p_3 = 41, p_4 = 37, p_5 = 31, p_6 = 29, p_7 = 23, p_8 = 19, p_9 = 17, p_{10} = 13$ модули СОК и $P = 266186053068611$ - диапазон. Разделим $a = 93$ на $b = 8$ и найдем округленное частное $q = \lfloor a/b \rceil$.

Представим b в виде (1) в ОПСС в порядке уменьшаемой значимости:

$b = b_{10}(47 \cdot 43 \cdot 41 \cdot 37 \cdot 31 \cdot 29 \cdot 23 \cdot 19 \cdot 17) + b_9(47 \cdot 43 \cdot 41 \cdot 37 \cdot 31 \cdot 29 \cdot 23 \cdot 19) +$

$+ b_8(47 \cdot 43 \cdot 41 \cdot 37 \cdot 31 \cdot 29 \cdot 23) + b_7(47 \cdot 43 \cdot 41 \cdot 37 \cdot 31 \cdot 29) + b_6(47 \cdot 43 \cdot 41 \cdot 37 \cdot 31) +$

$b_5(47 \cdot 43 \cdot 41 \cdot 37) + b_4(47 \cdot 43 \cdot 41) + b_3(47 \cdot 43) + b_2 47 + b_1,$

где $b_i = 0, \quad i = \overline{n,2}$, $b_1 = 8$. Тогда приблизительный делитель с учетом (2) и (3) будет равен $\bar{b} = 13$.

Найдем q_1 и выполним действия с учетом рекурсивных соотношений (4)

$$\begin{array}{ll} q_1 = \lfloor 93/13 \rceil = 7 & \\ a_1 = 93 - 8 \cdot 7 = 37 & q_2 = \lfloor 37/13 \rceil = 2 \\ a_2 = 37 - 8 \cdot 2 = 21 & q_3 = \lfloor 21/13 \rceil = 1 \\ a_3 = 21 - 8 \cdot 1 = 13 & q_4 = \lfloor 13/13 \rceil = 1 \\ a_4 = 13 - 8 \cdot 1 = 5 & q_5 = \lfloor 5/13 \rceil = 0 \end{array}$$

Из (5) так как $q_5 = 0$, $a_4 = 5 < b = 8$, то $q' = 0$. Тогда округленное частное будет $q = \sum_{i=1}^{4} q_i + q' = 7 + 2 + 1 + 1 + 0 = 11$. Действительно $\lfloor a/b \rceil = \lfloor 93/8 \rceil = 11$.

Рассмотренный алгоритм является эффективным, простым в реализации и легко может

быть модифицирован на язык кольцевых операций СОК.

Недостатком является то, что в некоторых случаях для вычисления округленного частного может потребоваться много итераций. Это происходит в тех случаях, когда делимое a - большое число, делитель b - относительно малое. На рисунке 1 показан график зависимости количества итераций от размера делимого a, если в качестве делителя выбрано $b = 8$, а в качестве делимого значения $a_1 = 25, a_2 = 93, a_3 = 345, a_4 = 843, a_5 = 1303, a_6 = 1541, a_7 = 12326,$ $a_8 = 213743, a_9 = 3543261, a_{10} = 7641234,$ $a_{11} = 10304312$ (из-за большого разброса значений a_i, $i = \overline{1,11}$ по оси OX отложены значения $\ln a$).

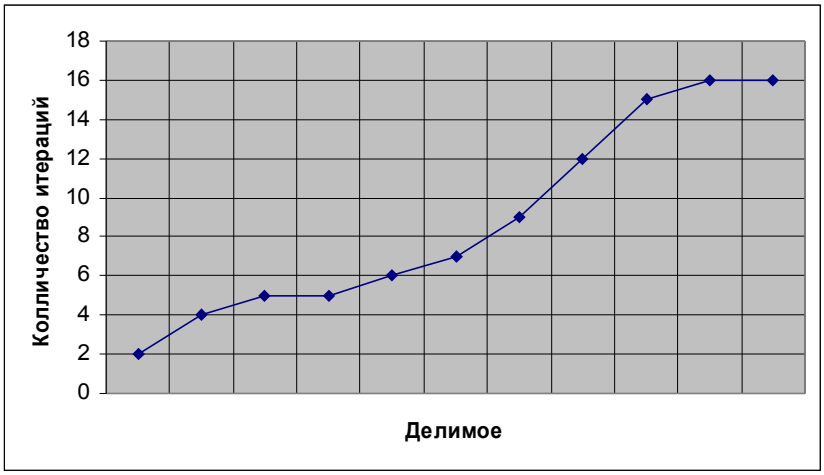

Рисунок 1. Зависимость количества итераций от величины делимого a при $b = 8$.

Из графиков видно, что при увеличении разрыва между делимым и делителем количество итераций будет бесконечно возрастать. Эта проблема может быть решена применением другого метода деления, основанного на итерациях Ньютона [6].

Для реализации этого алгоритма система оснований выбирается специальным образом. Пусть $q_1, q_2, ..., q_{2n}$ - взаимно простые числа, для которых

$$1 < q_1 < q_2 < ... < q_{2n-1} < q_{2n}, \qquad (6)$$

где $n \in Z$, $n \geq 1$. Разобьем эти числа на две группы следующим образом

$$p_1 = q_1, \quad p_2 = q_3, \quad p_3 = q_5, ..., p_n = q_{2n-1} \qquad (7)$$

и

$$p_{n+1} = q_2, \quad p_{n+2} = q_4, \quad p_{n+3} = q_6, ..., p_{2n} = q_{2n}. \qquad (8)$$

Тогда: $p_1, p_2, ..., p_{2n}$ - основания расширенной СОК; $p_1, p_2, ..., p_n$ - основания базовой СОК; $p_{n+1}, p_{n+2}, ..., p_{2n}$ - основания расширения базовой СОК до расширенной; $P = \prod_{i=1}^{2n} p_i$, $k = \prod_{i=1}^{n} p_i$, $\overline{k} = \prod_{i=n+1}^{2n} p_i$ -соответствующие диапазоны. Расширенная СОК определяет вычислительный диапазон для промежуточных значений, равный приблизительно квадрату от

соответствующего нормального диапазона.

Пусть необходимо разделить целое число a на целое число b.

Алгоритм деления, основанный на итерациях Ньютона, состоит из двух этапов: вычисление целочисленной обратной величины для делителя по отношению к нормальному диапазону СОК; нахождение частного и остатка.

В работе [6] показано, что обратная величина $\lfloor k/b \rfloor$ может быть вычислена применением формулы

$$z_{i+1} = \lfloor z_i(2k - bz_i)/k \rfloor . \quad (9)$$

В общем случае в качестве начального приближения можно выбрать $z_1 = 2$, а для уменьшения количества итераций

$$z_1 = 2^l < k/b < 2^{l+1} . \quad (10)$$

Итерации продолжаются до тех пор, пока выполняется условие:

$$z_{i+1} \neq z_i . \quad (11)$$

При $z_{i+1} = z_i$ вычисления прекращаются, и проверяется условие

$$t = k - b \cdot z_{i+1} - b < 0 . \quad (12)$$

Если это условие верное, то обратная величина

$$\lfloor k/b \rfloor = z_{i+1}, \quad (13)$$

иначе

$$\lfloor k/b \rfloor = z_{i+1} + 1 . \quad (14)$$

Второй этап деления a на b состоит в нахождении частного и остатка из равенств

$$q = \lfloor a \cdot z_{i+1}/k \rfloor , \quad (15)$$

$$r = a - q \cdot b . \quad (16)$$

Если

$$g = r - b > 0 , \quad (17)$$

то

$$\lfloor a/b \rfloor = q + 1 , \; a/b - \lfloor a/b \rfloor = r - b , \quad (18)$$

иначе

$$\lfloor a/b \rfloor = q , \; a/b - \lfloor a/b \rfloor = r . \quad (19)$$

Рассмотрим алгоритм на примере.

Пример 2. Разделим $a = 10304312$ на $b = 8$. Выберем основания СОК с учетом (6), (7), (8): $p_1 = 13, p_2 = 17, p_3 = 19, p_4 = 23, \; p_5 = 29, p_6 = 31, p_7 = 37, p_8 = 41, \; p_9 = 43, p_{10} = 47$ модули СОК и $P = 266186053068611$ - диапазон и $k = 13 \cdot 19 \cdot 29 \cdot 37 \cdot 43 = 11396333$. При этом $2 \cdot k = 22792666$.

Найдем целочисленную обратную величину $\lfloor k/b \rfloor$. Для этого будем выполнять итерации по (9) пока не выполнится (11). При этом начальное приближение выберем из условия (10), то есть $z_1 = 2^{20} < k/b < 2^{21}$.

$$z_1 = 2^{20} = 1048576$$

$$z_2 = \left\lfloor \frac{1048576 \cdot (22792666 - 8 \cdot 1048576)}{11396333} \right\rfloor = 1325316$$

$$z_3 = \left\lfloor \frac{1325316 \cdot (22792666 - 8 \cdot 1325316)}{11396333} \right\rfloor = 1417630$$

$$z_4 = \left\lfloor \frac{1417630 \cdot (22792666 - 8 \cdot 1417630)}{11396333} \right\rfloor = 1424508$$

$$z_5 = \left\lfloor \frac{1424508 \cdot (22792666 - 8 \cdot 1424508)}{11396333} \right\rfloor = 1424541$$

$$z_6 = \left\lfloor \frac{1424541 \cdot (22792666 - 8 \cdot 1424541)}{11396333} \right\rfloor = 1424541$$

Так как $t = 11396333 - 8 \cdot 1424541 - 8 = -3 < 0$, неравенство (12) выполняется и с учетом (13) получим $\lfloor k/b \rfloor = 1424541$.

Определим частное и остаток. Из (15) найдем $q = \left\lfloor \frac{10304312 \cdot 1424541}{11396333} \right\rfloor = 1288038$ и из (16) $r = 10304312 - 1288038 \cdot 8 = 8$. Так как $g = 8 - 8 = 0$, то условие (17) выполняется и согласно (18) получим: $q = 1288039$ - частное и $r = 0$ - остаток.

Для приведенного примера видно, что алгоритм на основе итераций Ньютона содержит 5 итераций, а алгоритм на основе метода спуска Ферма для тех же значений делимого и делителя содержит 16 итераций (рис.2). Это связано с тем, что количество итераций в алгоритме на основе итераций Ньютона вообще не зависит от величины делимого, а зависит только от величины делителя. Поэтому, если в качестве делителя выбрать $b = 8$, а в качестве делимого a выбирать сколь-угодно большие числа, то количество итераций все равно останется равным пяти. Поэтому алгоритм на основе итераций Ньютона в случаях, когда делимое a - большое число, делитель b - относительно малое, а $\bar{b}$ - аппроксимация b имеет явные преимущества перед алгоритмом, основанным на методе спуска Ферма. Однако, если разница между делимым и делителем не велика, например $a = 93$ и $b = 8$, то применение алгоритма на основе итераций Ньютона требует значительно больших вычислительных затрат, чем применение алгоритма на основе метода спуска Ферма. Поэтому для достижения наилучшего результата при целочисленном делении выбор применения одного из методов должен выполняться с учетом диапазона исходных данных.

Литература

1. Галушкин А.И., Червяков Н.И. Нейрокомпьютеры в остаточных классах. – М.: Радиотехника, 2003.

2. Акушский Н.Я., Юдицкий Д.И. Машинная арифметика в остаточных классах. – М.: Сов.

радио, 1968.

3. Червяков Н.И., Лавриненко И.Н., Лавриненко С.В., Мезенцева О.С. Методы и алгоритмы округления, масштабирования и деления чисел в модулярной арифметике // Сборник научных трудов. – Москва, Зеленоград: изд-во Ангстрем, МИЭТ, 2006. – С. 291-311.

4. Szabo N., Tanaka R. Residue arithmetic and its applications to computer technology. – New-York. 1967.

5. Червяков Н.И. Методы масштабирования модулярных чисел, используемые при цифровой обработке сигналов// Инфокоммуникационные технологии. 2006, Т.4, №3. – с. 15-23.

6. Markus A. Hitz and Erich Kaltofen. Integer division in Residue Number Systems// Appears in IEEE Trans. Computers, vol. 44(8), pp. 983-989, 1995.