

Ставропольский Военный Институт Связи Ракетных Войск

ПРЯМОЕ ПРЕОБРАЗОВАНИЕ ИЗ ДВОИЧНОГО КОДА В КОД ПСКВ

Основным достоинством полиномиальной системы классов вычетов является сравнительно простота модульных операций, что позволяет существенно повысить скорость вычислительных устройств цифровой обработки сигнала.

Для реализации вычислительного процесса с использованием полиномиальной системы классов вычетов необходимо осуществить преобразование из позиционного кода в модулярный и обратно.

Представление операнда в позиционном счислении определяется следующим образом

$$A(z) = a_r z^r + a_{r-1} z^{r-1} + \dots + a_1 z + a^0, \quad (1)$$

где a_j – элементы поля $GF(2)$; $j = 0, \dots, r$.

Для перевода из позиционной системы счисления в непозиционную, заданную взаимно простыми основаниями $p_1(z), p_2(z), \dots, p_n(z)$, необходимо выполнить операции деления на модули $p_i(z)$, $i = 1, 2, \dots, n$. Образование остатка $a_i(z)$ в этом случае осуществляется следующим образом

$$a_i(z) = A(z) - [A(z) / p_i(z)] p_i(z), \quad (2)$$

где $[A(z) / p_i(z)]$ – наименьшее целое от деления $A(z)$ на основание $p_i(z)$; $i = 1, 2, \dots, n$.

Таким образом, целью данной статьи является систематизация методов перевода из позиционного двоичного кода в коды системы классов вычета и выделение наиболее перспективных из них.

Все множество методов перевода из позиционной системы счисления в систему классов вычетов можно свести к трем основным группам. В основу методов, образующих первую группу, положен метод понижения разрядности числа, не содержащий операцию деления.

В основу данного метода положен следующий принцип: вычисление остатка осуществляется с помощью итерационного алгоритма. Для этого необходимо определить

остатки от деления на p_i степеней основания, которые дадут набор чисел C_i , $i=1, 2, \dots, r$. Если остаток от деления степени основания C_i превосходит половину модуля p_i , то в качестве значения C_i необходимо взять число, дополняющее до значения p_i , со знаком минус. Значения C_i можно знать заранее и они являются константами для выбранной системы счисления. Количество разрядов C_i определяется разрядностью исходного числа A . Затем цифры исходного числа умножаются на соответствующие числа C_i , полученная сумма определяется

$$A_i = A_k \cdot C_k + \dots + A_1 \cdot C_1 + A_0 \cdot C_0 < A_k \cdot S^k + \dots + A_1 \cdot S^1 + A_0 \quad (3)$$

По значению A_i можно узнать каков остаток от деления числа A на p_i . Если A_i имеет количество разрядов больше, чем p_i , то вновь цифры числа A_i необходимо умножить на числа C_i , причем полученная сумма будет $A_2 < A_i$.

По значению A_2 можно узнать каков остаток от деления числа A на p_i . Этот продолжается до тех пор, пока не получится число A_i , разрядностью равной или меньшей разрядности p_i . По данному числу и определяется остаток от деления числа A на p_i .

Несмотря на простоту реализации, данный метод преобразования чисел по модулю, построенных по принципу рекуррентной редукции, значительно сужают область применения модулярной арифметики.

Основу второй группы составляют методы, обеспечивающие пространственное распределение вычислительного процесса – перевода из ПСС в ПСКВ. Отказ от обратных связей в нейронных сетях позволяет устранить указанные выше недостатки и реализует обработку исходных данных на сети прямого распространения. В настоящее время предложена математическая модель нейронной сети, реализующей прямое преобразование позиционного двоичного кода в код классов вычетов, на основе сети прямого распространения. Число слоев в такой сети определяется количеством итераций l , необходимых для преобразования входных данных, а количество нейронов в каждом слое – разрядностью обрабатываемых данных на каждой из итераций. В этом случае итеративный алгоритм преобразования A по модулю p определяется выражением

$$A(l+1) = \sum_{i=0}^{ord A(l)} \left| 2^i \right|_p^+ \cdot \left[\left[\frac{A(l)}{2^i} \right]_2^+ \right], \quad (4)$$

где $l=0, 1, 2, \dots$ – число итераций.

Замена обратных связей в нейронных сетях на прямые позволяет повысить скорость обработки данных, так как в такой сети одновременно обрабатывается несколько отсчетов и в каждом такте работы сети на входе формируются преобразованные данные.

Вычислительные процессы третьей группы методов перевода чисел из ПСС в непозиционную систему реализуют различные варианты метода непосредственного суммирования.

Перевод из позиционного двоичного кода в ПСКВ осуществляется в соответствии с выражением

$$a_i \equiv A(z) \bmod p_i(z) = \sum_{l=0}^k a_l(z) \cdot z^l \bmod p_i(z), \quad (5)$$

где $i=1, 2, \dots, n$.

Для получения $A(z)$ в системе классов вычетов с основаниями $p_1(z), p_2(z), \dots, p_n(z)$ необходимо получить в этой системе значения $a_l(z) \cdot z^l \bmod p_i(z)$. В этом случае остаток по модулю $p_i(z)$ определяется

$$a_i(z) = \left\lfloor \sum_{l=0}^k (a_l^i \cdot z^l) \bmod p_i(z) \right\rfloor_2^+, \quad (6)$$

где $a_l^i = a_l \bmod p_i(z)$, $i = 1, 2, \dots, n$.

В соответствии с (6), перевод $A(z)$ из позиционной системы счисления в непозиционную сводится к суммированию по модулю два величин $(a_l^i(z) \cdot z^l) \bmod p_i(z)$ в соответствии с заданным полиномом $A(z)$.

Таким образом, очевидно, что модификация и реализация метода непосредственного суммирования для полиномиальной системы классов вычетов позволяет разрабатывать высокоскоростные преобразователи кодов для вычислительных структур реального масштаба времени.