$$l_{инт}(z)=\left|\sum_{j=k+1}^{k+r}R_j(z)g_{\ j}(z))\right|^{+}_{P_{кат}(z)}, \qquad (10)$$

где $R_j(z)=\lfloor B_j(z)/\ P_{раб}(z)\rfloor$.Доказательство закончено.

В работе [3] представлена организация сети нейронной логики, реализующей вычисление номера согласно (10) расширенном поле Галуа $GF(2^4)$. С целью сокращения схемных затрат целесообразно перейти к многомерной обработке данных. Тогда

$$\begin{cases} l^{k+1}_{инт}(z)=\left|\sum_{j=k+1}^{k+r}\left|R_j(z)g_{\ j}(z)\right|^{+}_{Pk+1(z)}\right|^{+}_{Pk+1(z)}\Bigg|^{+}_{PK+1(z)} \\ l^{k+2}_{инт}(z)=\left|\sum_{j=k+1}^{k+r}\left|R_j(z)g_{\ j}(z)\right|^{+}_{Pk+2(z)}\right|^{+}_{Pk+2(z)}\Bigg|^{+}_{Pk+2(z)} \\ \text{M} \\ l^{k+r}_{инт}(z)=\left|\sum_{j=k+1}^{k+r}\left|R_j(z)g_{\ j}(z)\right|^{+}_{Pk+r(z)}\right|^{+}_{Pk+r(z)}\Bigg|^{+}_{Pk+r(z)} \end{cases} \qquad (11)$$

Таким образом, применение выражения (11) позволяет осуществить процедуру поиска и коррекции ошибок с использованием нормированного полинома.

Выводы: Из полученных данных наглядно видно, что вычислительное устройство, реализующее алгоритм (11), обеспечивает наибольшую эффективность при контроле и исправлении ошибок, возникающих в процессе функционирования спецпроцессора.

СПИСОК ЛИТЕРАТУРЫ

1. Калмыков И.А., Червяков Н.И., Щелкунова Ю.О., Бережной В.В. Математическая модель нейронных сетей для исследования ортогональных преобразований в расширенных полях Галуа/Нейрокомпьютеры: разработка, применение. №6, 2003. с.61-68.

2. Калмыков И.А. Математические модели нейросетевых отказоустойчивых вычислительных средств, функционирующих в полиномиальной системе классов вычетов/Под ред. Н.И. Червякова. – М.: ФИЗМАТЛИТ, 2005. - 276 с.

3. Калмыков И.А., Червяков Н.И., Щелкунова Ю.О., Бережной В.В. Математическая модель нейронной сети для коррекции ошибок в непозиционном коде расширенного поля Галуа/Нейрокомпьютеры: разработка, применение. №8-9, 2003. С. 10-16.

ОБНАРУЖЕНИЕ И КОРРЕКЦИЯ ОШИБОК В КОДАХ ПОЛИНОМИАЛЬНОЙ СИСТЕМЫ КЛАССОВ ВЫЧЕТОВ НА ОСНОВЕ НУЛЕВИЗАЦИИ

Калмыков И.А.
Северо-Кавказский государственный технический университет, Ставрополь

Применение полиномиальной системы классов вычетов (ПСКВ), в которой в качестве оснований выбираются минимальные многочлены $p_i(z)$ поля $GF(p^n)$, позволяет полином $A(z)$, удовлетворяющий условию:

$$A(z)\in P_{пол},$$

$$P_{пол}=\prod_{i=1}^{k+r}p_i(z)=z^{p^n-1}-1, \qquad (1)$$

представить в виде:

$$A(z)=\left(a_1(z),a_2(z),...,a_{k+r}(z)\right), \qquad (2)$$

где $a_i(z)=rest\left(A(z)\big/p_i(z)\right)$, $i=1,2,...,k+r$.

Выполнение операций над операндами в поле Галуа $GF(p^n)$ производятся независимо по каждому из модулей $p_i(z)$. Независимость обработки информации по основаниям ПСКВ позволяет не только повысить скорость обработки, но так же и обеспечить обнаружение и коррекцию ошибок в процессе функционирования СП. Если на диапазон возможного изменения кодируемого множества полиномов наложить ограничения, то есть выбрать k из n оснований ПСКВ ($k<n$), то это позволит осуществить разбиение полного диапазона $P_{полн}(z)$ поля $GF(p^n)$ на два подмножества, первое из которых называется рабочим диапазоном

$$P_{раб}(z)=\prod_{i=1}^{k}p_i(z), \qquad (3)$$

а второе подмножество $GF(p^n)$, определяемое произведением $r=n-k$ контрольных оснований:

$$P_{конт}(z)=\prod_{i=k+1}^{k+r}p_i(z), \qquad (4)$$

задает совокупность запрещенных комбинаций. Многочлен $A(z)$ будет считаться разрешенным в том и только том случае, если он является элементом нулевого интервала полного диапазона $P_{полн}(z)$, то есть $A(z)\in P_{раб}(z)$.

Для определения местоположения $A(z)=(a_1(z),a_2(z),...,a_{k+r}(z))$ используется метод нулевизации, заключающейся в переходе от исходного полинома к полиному вида:

$$(0,0,...,x_{k+1}(z),...,x_{k+r}(z)), \qquad (5)$$

при помощи последовательных преобразований, при которых не имеет место ни один выход за пределы рабочего диапазона системы. Нулевизация заключается в последовательном вычитании из исходного полинома, представленного в модулярном коде, констант нулевизации, с целью получения

$$A_k(z) = A_{k-1}(z) - M_k(z) = \\ = (0,0,...,x_{k+1}(z),...,x_{k+r}(z)) \quad (6)$$

Если в результате выполнения процедуры нулевизации будет получен нулевой результат, то это свидетельствует, что комбинация *A(z)* не содержит ошибок. В противном случае – модулярный код *A(z)* – содержит ошибки.

Повысить скорость выполнения процедуры нулевизации можно за счет модификации констант нулевизации $M_i(z)$. Оставляя неизменным условие невыхода константы нулевизации $M_i(z)$ за пределы рабочего диапазона $P_{раб}(z) = \prod_{i=1}^{k} p_i(z)$, возьмем в качестве последних значения произведение остатков рабочих оснований на величину ортогональных базисов безизбыточной системы оснований

$$\begin{cases} a_1(z)B_1^*(z) \bmod P_{раб}(z) = \\ = (a_1(z),0,0,...,0,x_{k+1}^1(z),x_{k+2}^1(z),...,x_{k+1}^1(z)); \\ a_2(z)B_2^*(z) \bmod P_{раб}(z) = \\ = (0,a_2(z),0,...,0,x_{k+1}^2(z),x_{k+2}^2(z),...,x_{k+1}^2(z)); \\ \text{M} \\ a_k(z)B_k^*(z) \bmod P_{раб}(z) = \\ = (0,0,0,...,a_k(z),x_{k+1}^k(z),x_{k+2}^k(z),...,x_{k+1}^k(z)). \end{cases} \quad (7)$$

где $B_i^*(z)$ - ортогональный базис, безизбыточной системы оснований; *i=1,2,…,k.*

Тогда если положить условие, что $A(z) \in P_{раб}(z)$, где $P_{раб}(z) = \prod_{i=1}^{k} p_i(z)$, то полином $A(z) = (a_1(z), a_2(z),..., a_k(z))$ согласно китайской теореме об остатках (КТО) можно представить в виде

$$A(z) = (a_1,(z),0,0,...,0) + (0,a_2(z),0,...,0) + \\ + ...(0,0,0,...,a_k(z)) \quad (8)$$

Каждое слагаемое выражения (9) представляет собой:

$$(0,0,...0,a_i(z),0,...,0) = \\ = a_i(z)B_i^*(z) \bmod P_{раб}(z), \quad (9)$$

Подставим выражения (8) в равенство (10). Получаем:

$$A(z) = (a_1(z),0,0,...,0,(z),x_{k+1}^1(z), \\ x_{k+2}^1(z),...,x_{k+r}^1(z)) + (0,a_2(z),0,...,0, \\ x_{k+1}^2(z),x_{k+2}^2(z),...,x_{k+r}^2(z)) + ... + (0,0,0,..., \\ a_k(z),x_{k+1}^k(z),x_{k+2}^k(z),...,x_{k+r}^k(z)). \quad (10)$$

Разность полинома *A(z)* и модифицированных констант нулевизации $M_i(z)$, *i=1, 2,…, k,* псевдоортогональных форм, задаёт величину нормированного следа полинома

$$\begin{cases} x_{k+1}(z) = (a_{k+1}(z) - \sum_{j=1}^{k} x_{k+1}^j(z)) \bmod p_{k+1}(z), \\ \text{M} \\ x_{k+r}(z) = (a_{k+r}(z) - \sum_{j=1}^{k} x_{k+r}^j(z)) \bmod p_{k+r}(z). \end{cases} \quad (11)$$

Рассмотрим ПСКВ, определяемую в поле $GF(2^5)$. В таблице 1 помещены значения рабочих и контрольных оснований ПСКВ, а также динамический диапазон.

Таблица 1. Основания и динамический диапазон поля $GF(2^5)$

Основания ПСКВ		Рабочий диапазон ПСКВ
Рабочие	Контрольные	
$p_1(z) = z + 1$	$p_6(z) = z^5 + z^2 + 1$	$z^{21} + z^{19} + z^{16} + z^{13} + z^{11} + z^9 + z^8 + z^6 + z^3 + z^2 + z + 1$
$p_2(z) = z^5 + z^3 + 1$	$p_7(z) = z^5 + z^3 + z^2 + z + 1$	
$p_3(z) = z^5 + z^4 + z^2 + z + 1$		
$p_4(z) = z^5 + z^4 + z^3 + z + 1$		
$p_5(z) = z^5 + z^4 + z^3 + z^2 + 1$		

Определим все значения произведений степеней z^j на ортогональные базисы $B_i^*(z)$, учитывая невозможность выхода за пределы рабочего диапазона

$$P_{раб}(z) = z^{21} + z^{19} + z^{16} + z^{13} + z^{11} + z^9 + z^8 + \\ + z^6 + z^3 + z^2 + z + 1.$$

Полученные значения модифицированных констант нулевизации представлены в таблице 2.

Таблица 2. Константы нулевизации для поля GF(2^5)

	$\alpha_1(z)$	$\alpha_2(z)$	$\alpha_3(z)$	$\alpha_4(z)$	$\alpha_5(z)$	$\alpha_6(z)$	$\alpha_7(z)$
$z^0B_1^*(z)$	1	0	0	0	0	z^2	z
$z^0B_2^*(z)$	0	1	0	0	0	1	1
$z^1B_2^*(z)$	0	z	0	0	0	z	z
$z^2B_2^*(z)$	0	z^2	0	0	0	z^2	z^2
$z^3B_2^*(z)$	0	z^3	0	0	0	z^3	z^3
$z^4B_2^*(z)$	0	z^4	0	0	0	z^4	z^4
$z^0B_3^*(z)$	0	0	1	0	0	z^4+z	z^4+1
$z^1B_3^*(z)$	0	0	z	0	0	z^3+z^2+1	z^3+z+1
$z^2B_3^*(z)$	0	0	z^2	0	0	z^4+z^3+z	z^4+z^2+z
$z^3B_3^*(z)$	0	0	z^3	0	0	z^4+1	$z+1$
$z^4B_3^*(z)$	0	0	z^4	0	0	z^2+z+1	z^2+z
$z^0B_4^*(z)$	0	0	0	1	0	z^2	$z+1$
$z^1B_4^*(z)$	0	0	0	z	0	z^3	z^2+z
$z^2B_4^*(z)$	0	0	0	z^2	0	$z^4+z^3+z^2$	z^3+z
$z^3B_4^*(z)$	0	0	0	z^3	0	z^4+1	z^4+z^2
$z^4B_4^*(z)$	0	0	0	z^4	0	z^3+z+1	z^3+z^2+1
$z^0B_5^*(z)$	0	0	0	0	1	z^4+z	z^4
$z^1B_5^*(z)$	0	0	0	0	z	1	z^3+z^2+z+1
$z^2B_5^*(z)$	0	0	0	0	z^2	z	$z^4+z^3+z^2+z$
$z^3B_5^*(z)$	0	0	0	0	z^3	z^2	z^4+z+1
$z^4B_5^*(z)$	0	0	0	0	z^4	z^3	z^3+1

Пусть в поле $GF(2^5)$ задан полином $A(z)=z^6+z^5+z^4+1$. Данный полином принадлежит $P_{раб}(z)$. Представим его в модулярном коде

$A(z)=z^6+z^5+z^4+1=$ $(0,\ z^3+z,\ z^4+z^3+z^2+z+1,\ z^2+z+1,\ z^3+z+1,\ z^4+z^3+z^2+z,\ 0)$.

Проведем процедуру нулевизации.

1 этап

$A(z) = (0,\ z^3+z,\ z^4+z^3+z^2+z+1,\ z^2+z+1,\ z^3+z+1,\ z^4+z^3+z^2+z,\ 0)$

$\underline{M_2(z)=(0,\ z^3+z,\ 0,\ 0,\ 0,\ z^3+z,\ z^3+z)}$

$A_2(z)=(0,\ 0,\ z^4+z^3+z^2+z+1,\ z^2+z+1,\ z^3+z+1,\ z^4+z^2,\ z^3+z)$

2 этап

$A_2(z) =(0,\ 0,\ z^4+z^3+z^2+z+1,\ z^2+z+1,\ z^3+z+1,\ z^4+z^2,\ z^3+z)$

$\underline{M_3(z)=(0,\ 0,\ z^4+z^3+z^2+z+1,\ 0,\ 0,\ z^4+z+1,\ z^3+1)}$

$A_3(z)=(0,\ 0,\ 0,\ z^2+z+1,\ z^3+z+1,\ z^2+z+1,\ z+1)$

3 этап

$A_3(z) =(0,\ 0,\ 0,\ z^2+z+1,\ z^3+z+1,\ z^2+z+1,\ z+1)$

$\underline{M_4(z)=(0,\ 0,\ 0,\ z^2+z+1,\ 0,\ z^4,\ z^3+z^2+z+1)}$

$A_4(z)=(0,\ 0,\ 0,\ 0,\ z^3+z+1,\ z^4+z^2+z+1,\ z^3+z^2)$

4 этап

$A_4(z) =(0,\ 0,\ 0,\ 0,\ z^3+z+1,\ z^4+z^2+z+1,\ z^3+z^2)$

$\underline{M_5(z)=(0,\ 0,\ 0,\ 0,\ z^3+z+1,\ z^4+z^2+z+1,\ z^3+z^2)}$

$A_5(z)=(0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0)$

Таким образом, полином $A(z)$ не содержит ошибки.

Пусть ошибка произошла по 1 основанию

$A^*(z)=$ $(1,\ z^3+z,\ z^4+z^3+z^2+z+1,\ z^2+z+1,\ z^3+z+1,\ z^4+z^3+z^2+z,\ 0)$.

В результате проведения процедуры нулевизации получен результат: $A_5(z)=(0,\ 0,\ 0,\ 0,\ 0,\ z^2,\ z)$, что свидетельствует о наличии ошибки в модулярном коде.

СПИСОК ЛИТЕРАТУРЫ

1. Калмыков И.А., Червяков Н.И., Щелкунова Ю.О., Бережной В.В. Математическая модель нейронных сетей для исследования ортогональных преобразований в расширенных полях Галуа/Нейрокомпьютеры: разработка, применение. №6, 2003. с.61-68.

2. Калмыков И.А. Математические модели нейросетевых отказоустойчивых вычислительных средств, функционирующих в полиномиальной системе классов вычетов/Под ред. Н.И. Червякова. – М.: ФИЗМАТЛИТ, 2005. - 276 с.

ПРИМЕНЕНИЕ ИНТЕРВАЛЬНОГО НОМЕРА ДЛЯ КОРРЕКЦИИ ОШИБОК В КОДАХ ПОЛИНОМИАЛЬНОЙ СИСТЕМЫ КЛАССОВ ВЫЧЕТОВ

Калмыков И.А., Петлеваный С.В.,
Чипига А.А., Гахов В.Р.
Северо-Кавказский государственный технический университет, Ставрополь

Проблема исследований: Параллельная обработка данных в вычислительных трактах по модулям системы полиномиальной системы классов вычетов (ПСКВ) может служить базисом в реализации процедур коррекции ошибок. Разработанные алгоритмы поиска и исправления ошибок позволяют повысить эффективность функционирования спецпроцессоров (СП) ПСКВ.

Решение проблемы исследований: При решении многих практических задач цифровой обработки сигналов (ЦОС) необходимо осуществлять ортогональные преобразования. Известно, что использование математической модели ЦОС поля комплексных чисел характеризуется целым рядом недостатков, основные из которых приведены в работе [1]. В работах [2,3] предложена реализация ортогональных преобразований сигналов в полиномиальной системе классов вычетов (ПСКВ) расширенных полей Галуа *GF* (2^v). Основным достоинством такой непозиционной арифметики является возможность организации параллельных вычислений и, следовательно, значительное повышение быстродействия вычислительного устройства ЦОС. Кроме того, применение ПСКВ позволяет сократить аппаратурные затраты необходимы на реализацию вычислительной системы [3].

Если выбрать k из n оснований ПСКВ ($k < n$), то это позволит осуществить разбиение полного диапазона $P_{полн}(z)$ расширенного поля Галуа $GF(p^n)$ на два непересекающихся подмножества. Первое подмножество называется рабочим диапазоном и определяется выражением

$$P_{раб}(z) = \prod_{i=1}^{k} p_i(z), \qquad (1)$$

Многочлен $A(z)$ с коэффициентами из поля $GF(p)$ будет считаться разрешенным в том и только том случае, если он принадлежит $P_{раб}(z)$. Второе подмножество, определяемое произведением $r = n - k$ контрольных оснований,

$$P_{конт}(z) = \prod_{i=k+1}^{k+r} p_i(z), \qquad (2)$$

задает совокупность запрещенных комбинаций. Если $A(z)$ является элементом второго подмножества, то считается, что данная комбинация содержит ошибку. Таким образом, местоположение полинома $A(z)$ относительно двух данных подмножеств позволяет однозначно определить, является ли комбинация $A(z) = (a_1(z), \mathbf{K}, a_n(z))$ разрешенной, или содержит ошибочные символы.

Особое место среди методов поиска и коррекции ошибок в процессе вычислений отводится интервальному номеру полинома согласно выражения:

$$l_{инт}(z) = \left\lfloor A(z) / P_{раб}(z) \right\rfloor. \qquad (3)$$

В работе [4] представлено устройство, осуществляющее обнаружение и коррекцию ошибки в модулярном коде на основе вычисления интервального номера. В основу данного алгоритма положен алгоритм

$$l_{инт}(z) = \left| \sum_{i=1}^{k+r} a_i(z) R_i(z) + K^*(z) \right|^{+}_{P_{конт}(z)}, \qquad (4)$$

где ранг определяется выражением

$$K^*(z) = \left\lfloor \sum_{j=1}^{k} a_j(z) B_j^*(z) \Big/ P_{раб}(z) \right\rfloor. \qquad (5)$$

Если $l_{инт}(z) = 0$, то исходный полином $A(z)$ лежит внутри рабочего диапазона и не является запрещенным. В противном случае $A(z)$ – ошибочная комбинация.

Анализ выражения (4) показывает, что применение составного модуля $P_{коит}(z)$, с точки зрения аппаратурных затрат, является не самым оптимальным. Использование изоморфизма, порожденного КТО, позволяет перейти от одномерной обработки к многомерной. Приравнивая соответствующие значения $P_{конт}(z)$ и оснований $p_{k+1}(z), p_{k+2}(z), \ldots, p_{k+r}(z)$, получаем r преобразований

$$\begin{cases} l_{инт}^{k+1}(z) = \left| \sum_{i=1}^{k+r} a_i(z) R_i(z) + K^*(z) \right|^{+}_{p_{k+1}(z)}; \\ \mathbf{M} \\ l_{инт}^{k+r}(z) = \left| \sum_{i=1}^{k+r} a_i(z) R_i(z) + K^*(z) \right|^{+}_{p_{k+r}(z)}. \end{cases} \qquad (6)$$

Основным недостатком предложенного алгоритма является вычисление ранга $K(z)$. Решить данную проблему можно за счёт модификации алгоритма (6). В основу данной модификации положено свойство – отсутствие переноса единицы из младшего разряда в старший при выполнении арифметической операции сложения двух операндов в расширенных полях Галуа *GF* (2^v). Таким образом, величина ранга $K^*(z)$ безизбыточной системы ПСКВ $p_1(z), \ldots, p_k(z)$ определяется значением $a_i(z)$ и $B_i^*(z)$, и никоим образом не зависит от переполнения диапазона $P_{раб}(z)$. Тогда (6) примет вид:

$$\begin{cases} l_{инт}^{k+1}(z) = \left| \sum_{i=1}^{k} (a_i(z) B_i^*(z)) \bmod P_{раб}(z) + \right. \\ \qquad \left. + \sum_{i=k+1}^{k+r} a_i(z) R_i(z) \right|^{+}_{P_{k+1}(z)} \\ \mathbf{M} \\ l_{инт}^{k+r}(z) = \left| \sum_{i=1}^{k} (a_i(z) B_i^*(z)) \bmod P_{раб}(z) + \right. \\ \qquad \left. + \sum_{i=k+1}^{k+r} a_i(z) R_i(z) \right|^{+}_{P_{k+r}(z)} \end{cases} \qquad (7)$$