DOI: 10.15514/ISPRAS-2022-34(3)-6

Модификация алгоритма обнаружения и локализации ошибки в системе остаточных классов

[1] А.В. Гладков, ORCID: 0000-0002-9454-7618 <agladkov@ncfu.ru>
[2] В.А. Кучуков, ORCID: 0000-0002-1839-2765 <vkuchukov@ncfu.ru>
[1,3] М.Г. Бабенко, ORCID: 0000-0001-7066-0061 <mgbabenko@ncfu.ru>
[3,4,5] А.Н. Черных, ORCID: 0000-0001-5029-5212 <chernykh@cicese.mx>
[1] В.В. Бережной, ORCID: 0000-0002-8327-2735 <vvberezhnoi@ncfu.ru>
[6] А.Ю. Дроздов, ORCID: 0000-0001-5607-2749 <alexander.y.drozdov@gmail.com>

[1] Северо-Кавказский федеральный университет,
355017, Россия, г. Ставрополь, ул. Пушкина, 1
[2] Северо-Кавказский центр математических исследований СКФУ
355017, Россия, г. Ставрополь, ул. Пушкина, 1
[3] Институт системного программирования им. В.П. Иванникова РАН,
109004, Россия, г. Москва, ул. А. Солженицына, д. 25
[4] Центр научных исследований и высшего образования,
Мексика, 22860, Нижняя Калифорния, Энсенада, ш. Тихуана-Энсенада, 3918
[5] Южно-Уральский государственный университет,
454080, Россия, Челябинск, проспект Ленина, 76
[6] Московский физико-технический институт,
141701, Россия, Долгопрудный, Институтский пер., 9

Аннотация. В статье рассмотрена модификация алгоритма обнаружения и локализации ошибки в системе остаточных классов (СОК). Классическая избыточная СОК с одним контрольным основанием позволяет обнаружить ошибку, но не локализовать её. Для локализации одиночной ошибки вводят два контрольных основания. Благодаря накладываемым на основания СОК ограничениям уи разработанному алгоритму дается достичь исправления ошибок при одном контрольном основании, передаваемом по надежному каналу связи. Проведено моделирование классического и предложенного подходов с использованием Verilog на ASIC в среде RTL и физического синтеза Cadence Genus Synthesis Solution Предложенный алгоритм позволяет значительно сократить используемое при аппаратной реализации оборудование, незначительно увеличив время работы. На основе предложенного алгоритма разработана система распределенного хранения данных.

Ключевые слова: избыточная система остаточных классов; обнаружение ошибок; аппаратное моделирование

Для цитирования: Гладков А.В., Кучуков В.А., Бабенко М.Г., Черных А.Н., Бережной В.В., Дроздов А.Ю. Модификация алгоритма обнаружения и локализации ошибки в системе остаточных классов. Труды ИСП РАН, том 34, вып. 3, 2022 г., стр. 75-88. DOI: 10.15514/ISPRAS–2022–34(3)–6.

Благодарности: Работа выполнена при поддержке Российского научного фонда, проект №19-71-10033.

Modified Error Detection and Localization in the Residue Number System

[1] A.V. Gladkov, ORCID: 0000-0002-9454-7618 <agladkov@ncfu.ru>
[2] V.A. Kuchukov, ORCID: 0000-0002-1839-2765 <vkuchukov@ncfu.ru>
[1,3] M.G. Babenko, ORCID: 0000-0001-7066-0061 <mgbabenko@ncfu.ru>
[3,4,5] A.N. Tchernykh, ORCID: 0000-0001-5029-5212 <chernykh@cicese.mx>
[1] V.V. Berezhnoy, ORCID: 0000-0002-8327-2735 <vvberezhnoi@ncfu.ru>
[6] A.Yu. Drozdov, ORCID: 0000-0001-5607-2749 <alexander.y.drozdov@gmail.com>

[1] North-Caucasus Federal University,
1, Pushkin st., Stavropol, 355017, Russia
[2] North-Caucasus Center for Mathematical Research NCFU,
1, Pushkin st., Stavropol, 355017, Russia
[3] Ivannikov Institute for System Programming of the Russian Academy of Sciences,
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia
[4] Centro de Investigación Científica y de Educación Superior,
3918, Ensenada-Tijuana Highway, Ensenada, 22860, Mexico
[5] South Ural State University,
76, Lenin prospekt, Chelyabinsk, 454080, Russia
[6] Moscow Institute of Physics and Technology,
9 Institutskiy per., Dolgoprudny, Moscow Region, 141701

Abstract. This article presents the design of the modified error detection and localization algorithm in the Residue Number System (RNS). Classical redundant RNS with one control modulus can detect one error but not localize it. Two control moduli are used to localize a single error. Presented algorithm can achieve an error correction with a single control modulus transmitted over a reliable communication channel. The proposed approach was verified using Verilog on ASIC in RTL and physical synthesis tool Cadence Genus Synthesis Solution. It significantly reduces the area of the hardware implementation increasing the packing density and more efficient use of silicon resource. It slightly increases the running time compared with the classical algorithm. Distributed data storage was developed to study efficiency of the proposed algorithm.

Keywords: redundant residue number system; error detection; hardware simulation

For citation: Gladkov A.V., Kuchukov V.A., Babenko M.G., Tchernykh A.N., Berezhnoy V.V., Drozdov A.Yu. Modified Error Detection and Localization in the Residue Number System. Trudy ISP RAN/Proc. ISP RAS, vol. 34, issue 3, 2022, pp. 75-88 (in Russian). DOI: 10.15514/ISPRAS–2022–34(3)–6

Acknowledgements. This work was supported in part by the Russian Science Foundation, project number 19-71-10033.

1. Введение

Надежное хранение и обработка данных является необходимым условием работоспособности компьютерных систем. Для обеспечения достоверности восстановления информации в литературе предложен ряд подходов: дублирование данных и вычислительных каналов [1], схемы разделения секрета [2], коды обнаружения и коррекции ошибок, такие как коды стирания [3], избыточная система остаточных классов [4, 5]. Все эти коды используют избыточность для обнаружения и исправления ошибок. При этом также важна достоверность арифметической и логической обработки информации.

Дублирование данных. Дублирование является самым простым в реализации и наиболее распространенным методом введения избыточности [6]. Использование дублирования и двойного просчета позволяет за счет сравнения проверить правильность или неправильность вычислений. Тройной просчет за счет мажоритарной обработки уже позволяет исправить

ошибку. Однако такой подход фактически уменьшает производительность вычислительной системы минимум в два раза [7].

Помехоустойчивое кодирование. Теория помехоустойчивого кодирования для каждого конкретного канала позволяет выбрать наиболее эффективный метод обнаружения и исправления ошибок. Существуют два взаимодополняющих метода борьбы с помехами [8].

- Кодирование для исправления ошибок – приемник обнаруживает и исправляет ошибки;
- Кодирование для обнаружения ошибок – приемник распознает ошибки и, в случае необходимости, производит запрос на повторную передачу ошибочного блока.

Среди корректирующих кодов наибольшее распространение получили блочные двоичные коды, т.е. передача двоичного сообщения производится блоками, причем каждый блок содержит двоичных символов. Кодирование и декодирование каждого блока производится независимо от других блоков [8]. Коды стирания преобразуют сообщение из k символов в сообщение из n символов, $k < n$, что исходное сообщение может быть восстановлено по любым k' символам. Простейшим таким кодов является код проверки на четность.

В общем случае коды обнаружения и коррекции ошибок содержат две группы цифр – информационную и контрольную. Однако неарифметичность позиционных кодов не позволяет контролировать результаты арифметических операций контрольных оснований [7], таким образом присутствует неравноправность информационной и контрольной частей кода. Непозиционных системы счисления, такие как система остаточных классов, лишены этого недостатка.

Далее в разд. 2 рассмотрены особенности обнаружения и коррекции ошибок в избыточной системе остаточных классов. В разд. 3 предложена модификация алгоритма обнаружения и исправления ошибки в системе остаточных классов с одним избыточным основанием. В разд. 4 приведены результаты моделирования рассматриваемых алгоритмов на ASIC. В разд. 5 предложен вариант применения алгоритма для системы надежного распределенного хранения. В разд. 6 представлены основные результаты работы и направления дальнейшей работы.

2. Избыточная система остаточных классов

Пусть задана система остаточных классов (СОК) с взаимно простыми основаниями $p_1, p_2, \dots, p_n$.

Тогда рабочим диапазоном системы будет $P = p_1 \cdot p_2 \cdot \dots \cdot p_n$.

Если основания удовлетворяют условию $p_1 < p_2 < \dots < p_n$, то система называется упорядоченной.

Чтобы обеспечить возможность обнаружения и локализации ошибок вводят избыточные основания $p_{n+1}, \dots, p_{n+k}$.

Полным диапазоном СОК будет $\overline{P} = P \cdot p_{n+1} \cdot \dots \cdot p_{n+k}$. В данном случае критерием корректности числа $A = (\alpha_1, \alpha_2, \dots, \alpha_n, \alpha_{n+1})$ будет выполнение условия $A < P$, в противном случае число содержит ошибку.

Для избыточной СОК используют обозначение (k, n), где k – количество рабочих оснований, n – общее количество оснований.

Рассмотрим в общем случае СОК с одним избыточным основанием p_{n+1}, для которой $p_i < p_{n+1}$, $i = 1,2, \dots, n$. Тогда любое искажение цифры по одному какому-либо разряду превращает это число в неправильное и позволяет тем самым обнаружить наличие ошибки [7]. При этом накладывают ограничения, что ошибка возникает только по информационным основаниям.

Одним из методов локализации ошибок модулярного кода является метод проекций. Проекцией A_i числа $A = (\alpha_1, \alpha_2, \dots, \alpha_{n+1})$ по основанию p_i будет число, полученное вычеркиванием цифры α_i в представлении A.

Теорема 1. Если в упорядоченной системе остаточных классов проекция A_i числа $A = (\alpha_1, \alpha_2, \dots, \alpha_i, \dots \alpha_n, \alpha_{n+1})$ по основанию p_i удовлетворяет условию

$$A_i > \frac{\overline{P}}{p_{n+1}},$$

то цифра α_i правильная, если возможна лишь одиночная ошибка [7].

Введение только одного контрольного основания в общем случае не позволяет локализовать ошибку. Для коррекции ошибки можно использовать метод на основе Китайской теоремы об остатках (КТО) и методе проекций. Запишем его в виде алгоритма:

Алгоритм 1. Восстановление числа на основе метода проекций и КТО.

Input: $X' = (x'_1, x'_2, \dots, x'_n, x'_{n+1})$,

$p_1, p_2, \dots, p_{n+1}$, $P = \prod_{i=1}^{n} p_i$, $\overline{P} = p_n \cdot P$

$w_i = \left|\overline{P}_i^{-1}\right|_{p_i} \cdot \overline{P}_i$, где $\overline{P}_i = \overline{P}/p_i$, для всех $i \in [1, n+1]$.

$w_{i,j} = \left|\overline{P}_{i,j}^{-1}\right|_{p_j} \cdot \overline{P}_{i,j}$, где $\overline{P}_{i,j} = \overline{P}_i/p_j$, для всех $i \in [1, n+1]$ и $j \neq i$.

Output: X

```
1. S=0
2. for i = 1 to n + 1
   2.1. S = S + x'_i · w_i
3. S = S mod P̄
4. if S < P
   4.1. X = S
   4.2. return X
5. else
   5.1. for i = 1 to n + 1
      5.1.1. S_i = 0
      5.1.2. for j = 1 to n + 1
         5.1.2.1. if i ≠ j then
            5.1.2.1.1. S_i = S_i + x'_i · w_{i,j}
      5.1.3.   X = S_i mod P̄_i
      5.1.4.  if X < P then
         5.1.4.1. return X
```

Рассмотрим пример некорректной локализации ошибки. Пусть задана СОК {3,5,7,11,13} с четырьмя информационными основаниями и одним контрольным. Тогда рабочим диапазоном будет $P = 1155$, полный диапазон $\overline{P} = 15015$.

Возьмем число $X = 4 = (1,4,4,4,4)$ и введем ошибку по второму основанию, получим $X' = (1,0,4,4,4) = 6010$. Поскольку $X' = 6010 > P = 1155$ можно сделать вывод, что X' содержит ошибку. Тогда проекции $X'_1 = 1005 = (0,4,4,4)$, $X'_2 = 4 = (1,4,4,4)$, $X'_3 = 1720 = (1,0,4,4)$, $X'_4 = 550 = (1,0,4,4)$, $X'_5 = 235 = (1,0,4,4)$. Очевидно, что проекции X'_1, X'_2, X'_4, X'_5 удовлетворяют условию корректности, откуда следует что одно избыточное основание позволяет только обнаружить ошибку, но не локализовать её.

Для обеспечения возможности коррекции ошибок введем два контрольных основания p_{n+1}, p_{n+2}.

Теорема 2. Если в системе $p_1, p_2, \dots, p_n, p_{n+1}, p_{n+2}$ с двумя контрольными основаниями задано неправильное число $A' = (\alpha'_1, \alpha'_2, \dots, \alpha'_i, \dots, \alpha'_n, \alpha'_{n+1}, \alpha'_{n+2})$, то необходимым и достаточным условием ошибочности цифры α'_i в A' является правильности его проекции A'_i по основанию p_i [7].

Рассмотрим аналогичный пример с двумя контрольными основаниями. Возьмем СОК $\{3,5,7,11,13,17\}$, рабочий диапазон у которой будет $P = 1155$, а полный диапазон $\overline{P} = 255255$. Возьмем число $X = 4 = (1,4,4,4,4,4)$ и введем ошибку по второму основанию, получим $X' = (1,0,4,4,4,4) = 51055$. Поскольку $X' = 51055 > P = 1155$ можно сделать вывод, что X' содержит ошибку. Тогда проекции $X'_1 = 51055$, $X'_2 = 4$, $X'_3 = 14590$, $X'_4 = 4645$, $X'_5 = 11785$, $X'_6 = 6010$. Очевидно, что только $X'_2 = 4 < P$, значит ошибка произошла по второму основанию и исходное число равно 4.

Однако недостатком исправления ошибки с использованием двух контрольных оснований является сложность восстановления чисел с на основе КТО по каждой проекции. Для решения этой проблемы рассмотрим модификацию алгоритма обнаружения и коррекции ошибок.

3. Модификация алгоритма обнаружения и локализации ошибки в системе остаточных классов

Пусть задана система остаточных классов с одним контрольным основанием $p_1 < p_2 < p_3 < \cdots < p_n < p_{n+1}$, при этом считается, что контрольное основание надежное и не может содержать ошибки. Тогда введем алгоритм

Алгоритм 2. Восстановление числа на основе метода проекций и КТО.

Input: $X' = (x'_1, x'_2, \dots, x'_n, x'_{n+1})$,

$p_1, p_2, \dots, p_{n+1}$, $P = \prod_{i=1}^{n} p_i$, $\overline{P} = p_n \cdot P$

$\overline{P}_i = \overline{P}/p_i, i \in [1, n+1]$.

$w_i = \left|\overline{P}_i^{-1}\right|_{p_i}$.

Output: X

1. S=0
2. for $i = 1$ to $n+1$
   2.1. $S = S + x'_i \cdot w_i \cdot \overline{P}_i$
3. $X = S \bmod \overline{P}$
4. if $X < P$
   4.1. return X
5. else
   5.1. $k = 1$
   5.2. $X = S \bmod \overline{P}_1$
   5.3. while $X > P$ AND $k \le n$ do
      5.3.1. $k = k + 1$
      5.3.2. $X = S \bmod \overline{P}_k$
   5.4. return X

Теорема 3. Если $p_{n+1} > p_n \cdot p_{n-1}$, то алгоритм 2 корректен.

Доказательство

Пусть ошибка произошла по k-му основанию, где $k \in [1, n]$. Для доказательства теоремы достаточно показать, что для всех $X \in [0, P)$, $j \in [1, n]$ и $j \neq k$, выполняется следующее неравенство:

$$P \le |X + e_k \cdot w_k \cdot \bar{P}_k|_{\bar{P}_j} \tag{1}$$

Вычислим $|X + e_k \cdot w_k \cdot \bar{P}_k|_{\bar{P}_j}$ используя формулу $|a|_m = a - \left\lfloor \frac{a}{m} \right\rfloor \cdot m$, получим:

$$|X + e_k \cdot w_k \cdot \bar{P}_k|_{\bar{P}_j} = X + e_k \cdot w_k \cdot \bar{P}_k - \left\lfloor \frac{X + e_k \cdot w_k \cdot \bar{P}_k}{\bar{P}_j} \right\rfloor \cdot \bar{P}_j \tag{2}$$

Так как $\frac{\bar{P}_k}{\bar{P}_j} = \frac{p_j}{p_k}$, то представим выражение $\left\lfloor \frac{X + e_k \cdot w_k \cdot \bar{P}_k}{\bar{P}_j} \right\rfloor$ в, следующем, виде:

$$\left\lfloor \frac{X + e_k \cdot w_k \cdot \bar{P}_k}{\bar{P}_j} \right\rfloor = \left\lfloor \frac{X}{\bar{P}_j} + \frac{e_k \cdot w_k \cdot \bar{P}_k}{\bar{P}_j} \right\rfloor = \left\lfloor \frac{X}{\bar{P}_j} + \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \tag{3}$$

Оценим значение величины $\frac{X}{\bar{P}_j}$ для всех $j \in [1, n]$ и $j \neq k$, получим

$$\frac{X}{\bar{P}_j} \le \frac{P-1}{\bar{P}_j} = \frac{P}{\bar{P}_j} - \frac{1}{\bar{P}_j} \tag{4}$$

Так как $\frac{P}{\bar{P}_j} = \frac{p_j}{p_{n+1}}$ и по условию теоремы $p_{n+1} > p_n \cdot p_{n-1}$, то $\frac{p_j}{p_{n+1}} < \frac{1}{p_k}$ для всех $j \in [1, n]$ и $j \neq k$, следовательно, Eq. (4) примет вид:

$$\frac{X}{\bar{P}_j} < \frac{1}{p_k} \tag{5}$$

Учитывая неравенство (5), выражение (3) примет вид:

$$\left\lfloor \frac{X + e_k \cdot w_k \cdot \bar{P}_k}{\bar{P}_j} \right\rfloor = \left\lfloor \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \tag{6}$$

Подставляя выражения (2) и (6) в неравенство (1), получим:

$$P \le X + e_k \cdot w_k \cdot \bar{P}_k - \left\lfloor \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \cdot \bar{P}_j \tag{7}$$

Разделим левую часть неравенства (7) на положительное число P, получим:

$$1 \le X + e_k \cdot w_k \cdot \frac{\bar{P}_k}{P} - \left\lfloor \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \cdot \frac{\bar{P}_j}{P} \tag{8}$$

Так как $\frac{\bar{P}_k}{P} = \frac{p_{n+1}}{p_k}$ и $\frac{\bar{P}_j}{P} = \frac{p_{n+1}}{p_j}$, то выражение (8) примет вид:

$$1 \le X + e_k \cdot w_k \cdot \frac{p_{n+1}}{p_k} - \left\lfloor \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \cdot \frac{p_{n+1}}{p_j} \tag{9}$$

Умножим правую и левую часть неравенства (7) на положительное число $\frac{p_k \cdot p_j}{p_{n+1}}$, получим:

$$\frac{p_k \cdot p_j}{p_{n+1}} \le X \cdot \frac{p_k \cdot p_j}{p_{n+1}} + e_k \cdot w_k \cdot p_j - \left\lfloor \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \cdot p_k \tag{10}$$

Так как

$$e_k \cdot w_k \cdot p_j - \left\lfloor \frac{e_k \cdot w_k \cdot p_j}{p_k} \right\rfloor \cdot p_k = \left| e_k \cdot w_k \cdot p_j \right|_{p_k},$$

то формула (10) примет вид:

$$\frac{p_k \cdot p_j}{p_{n+1}} \le X \cdot \frac{p_k \cdot p_j}{p_{n+1}} + \left| e_k \cdot w_k \cdot p_j \right|_{p_k} \tag{11}$$

Учитывая, что $X \in [0, P-1)$ и $1 \le \left| e_k \cdot w_k \cdot p_j \right|_{p_k} \le p_k - 1$, то неравенство (11) выполняется если выполняется для всех $j \in [1, n]$ и $j \neq k$, следующее, условие:

$$\frac{p_k \cdot p_j}{p_{n+1}} \le 1 \tag{12}$$

Так как модули СОК удовлетворяют условию $p_1 < p_2 < p_3 < \cdots < p_n$, то из неравенства (12), следует, что необходимым и достаточным условием выполнения неравенство (1), является: $p_{n+1} \ge p_n p_{n-1}$.

Теорема доказана.

Рассмотрим пример обнаружения и исправления ошибки в СОК с одним контрольным основанием.

Выберем систему оснований СОК, удовлетворяющую теореме 3: $p_1 = 2, p_2 = 3, p_3 = 5, p_4 = 7, n = 4$. Контрольное основание $p_5 = 37 > 7 \cdot 5$. Искомое число $X = 53 = (1,2,3,4,16)$.

Параметры СОК:

$P = 2 \cdot 3 \cdot 5 \cdot 7 = 210$ — рабочий диапазон.

$\overline{P} = p_5 \cdot P = 35 \cdot 210 = 7770$ — полный диапазон СОК.

$\overline{P}_1 = \frac{\overline{P}}{p_1} = 3885, \overline{P}_2 = \frac{\overline{P}}{p_2} = 2590,$

$\overline{P}_3 = \frac{\overline{P}}{p_3} = 1554, \overline{P}_4 = \frac{\overline{P}}{p_4} = 1110,$

$\overline{P}_5 = \frac{\overline{P}}{p_5} = 210.$

$w_1 = \left|\overline{P}_1^{-1}\right|_{p_1} = 1, w_2 = \left|\overline{P}_2^{-1}\right|_{p_2} = 1,$

$w_3 = \left|\overline{P}_3^{-1}\right|_{p_3} = 4, w_4 = \left|\overline{P}_4^{-1}\right|_{p_4} = 2,$

$w_5 = \left|\overline{P}_5^{-1}\right|_{p_5} = 3.$

Ввведем вектор ошибки $E = (0,0,1,0,0), X' = E + X = (1,2,4,\ 4,16)$.

Вычислим

$$S = \left|\sum_{i=1}^{n+1} w_i \overline{P}_i x_i'\right|_{\overline{P}} =$$

$$= |1 \cdot 3885 \cdot 1 + 1 \cdot\ 2590 \cdot 2 + 4 \cdot 1554 \cdot 4 + 2 \cdot 1110 \cdot 4 + 3 \cdot\ 210 \cdot 16|_{7770} = 6269.$$

Так как $S = 6269 > 210$, следовательно, есть ошибка, вычислим ее:

$|S|_{\overline{P}_1} = 2384 > P$, $|S|_{\overline{P}_2} = 1089 > P$, $|S|_{\overline{P}_3} = 53 < P$, следовательно $X = 53$.

4. Моделирование и анализ

Для анализа и моделирования были выбраны наборы модулей СОК с 4, 5, 6 рабочими основаниями, рабочий диапазон которых покрывает диапазон в 8, 16, 24 и 32 бита

Избыточность данных является важным вопросом. В избыточной (k, n) СОК, используя любые k остатков из n, мы можем восстановить данные. Тогда избыточность можно выразить выражением

$$\frac{\sum_{i=1}^{n} d_i}{\sum_{i=1}^{k} d_i} - 1,$$

где d_i размерность остатков по основанию p_i.

Для моделирования работы алгоритма 1 были выбраны наборы с двумя контрольными основаниями и вычислена их избыточность. Результаты представлены в табл. 1.

Табл. 1. Избыточность ИСОК с двумя контрольными основаниями
Table 1. The redundancy of the RRNS with two control moduli.

Набор оснований	Размерность, бит	Избыточность
{3, 5, 7, 11, 13, 17}	10	0.75
{13, 17, 19, 23, 29, 31}	16	0.53
{59, 61, 67, 71, 73, 79}	24	0.54
{251, 257, 263, 269, 271, 277}	32	0.51
{2, 3, 5, 7, 11, 13, 17}	11	0.69
{5, 7, 11, 13, 17, 19, 23}	16	0.53
{23, 29, 31, 37, 41, 43, 47}	24	0.44
{79, 83, 89, 97, 101, 103, 107}	32	0.40
{2, 3, 5, 7, 11, 13, 17, 19}	14	0.59
{3, 5, 7, 11, 13, 17, 19, 23}	17	0.48
{11, 13, 17, 19, 23, 29, 31, 37}	24	0.39
{31,37, 41, 43, 47, 53, 59, 61}	32	0.34

Аналогично для моделирования работы алгоритма 2 были выбраны наборы с одним контрольным основанием, удовлетворяющие условиям теоремы 3 и вычислена их избыточность. Результаты представлены в табл. 2.

Табл. 2. Избыточность ИСОК с двумя контрольными основаниями
Table 2. The redundancy of the RRNS with one control modulus.

Набор оснований	Рабочий диапазон, бит	Избыточность
{3, 5, 7, 11, 79}	10	0.58
{13, 17, 19, 23, 439}	16	0.47
{59, 61, 67, 71, 4759}	24	0.50
{251, 257, 263, 269, 70753}	32	0.49
{2, 3, 5, 7, 11, 79}	11	0.54
{5, 7, 11, 13, 17, 223}	16	0.42
{23, 29, 31, 37, 41, 1523}	24	0.41
{79, 83, 89, 97, 101, 9803}	32	0.40
{2, 3, 5, 7, 11, 13, 149}	14	0.47
{3, 5, 7, 11, 13, 17, 223}	17	0.38
{11, 13, 17, 19, 23, 29, 673}	24	0.36
{31, 37, 41, 43, 47, 53, 2503}	32	0.34

В среднем избыточность предложенного алгоритма на 14% меньше, чем у СОК с двумя контрольными основаниями.

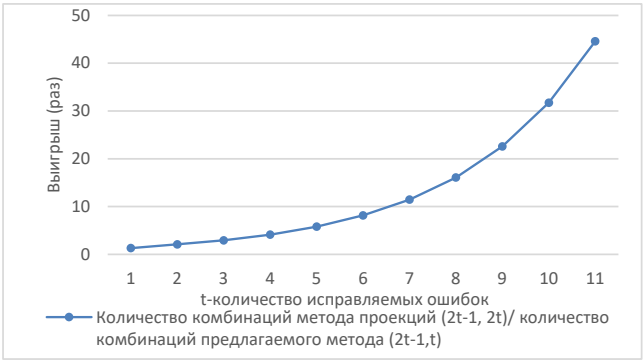

Рис. 1. Анализ количества комбинаций предлагаемого метода по сравнению с методом проекций
Fig. 1. The ratio of the number of combinations of the projection method to the number of combinations of the proposed method.

Метод проекций, представленный алгоритмом 1 требует большого количество восстановлений чисел с использованием КТО для каждой проекции. На рис. 1 показано уменьшение количества комбинаций предлагаемого метода по сравнению с методом проекций.

Моделирование предложенных алгоритмов было реализовано на языке Verilog на ASIC в среде RTL и физического синтеза Cadence Genus Synthesis Solution с использованием библиотеки osu018_stdcells.

Все значения, которые могли быть предвычислены записаны в константы. В качестве критериев, по которым проводился анализ, были выбраны площадь (Cell Area) и задержка комбинаторного пути (Arrival). Также был получен показатель количества используемых ячеек (Cell Count), однако он связан с площадью, но в то же время при одинаковом количестве ячеек общая площадь может быть различной ввиду различной сложности ячеек используемой библиотеки.

Результаты моделирования представлены в таблице 3. В среднем предложенный алгоритм 2 выполняется среднем в 1,5 раза дольше за счет последовательности действий 3 и 5 алгоритма, но при этом площадь микросхемы в среднем в 3,5 раза меньше. Это связано с ресурсоемкими вычислениями проекций, которые, однако, могут быть вычислены параллельно. При этом стоит обратить внимание на 32-битный рабочий диапазон. В этом случае для 5 и 6 рабочих основаниях время прохождения сигнала примерно одинаковое, а площадь реализации алгоритма 1 в 6 раз больше.

Табл. 3. Результаты моделирования на ASIC
Table 3. The results of the simulation

Показатель	Количество рабочих модулей	Алгоритм	Покрываемый рабочий диапазон, бит			
			8	16	24	32
Время, пс	4	1	15447	22338	34747	51685
		2	25097	39429	67242	76830
	5	1	15383	23642	30307	51164
		2	21944	36698	59120	44727
	6	1	19083	23047	31911	48572
		2	32558	41193	53634	48738
Площадь	4	1	156978	320430	652211	956196
		2	60759	132332	206485	280213
	5	1	184405	335056	657195	1006255
		2	63763	131767	154479	166433
	6	1	302090	411108	652668	1011104
		2	96815	155115	187973	184302

5. Система распределенного хранения данных

Одним из возможных применений данного алгоритма может быть система распределенного хранения данных, в которой выполняется подготовка исходных файлов для надежного распределенного хранения, посредством перевода в систему остаточных классов, удовлетворяющую теореме 3 и для восстановления полученных файлов, принятых из распределенной среды в случае ошибки или неполучения одной из частей файла по алгоритму 2.

Существует большое количество систем хранения данных, описанных в патентах США, России, Китая [заявка WO2019199288, опубл. 17.10.2019], [заявка US2013173916, опубл. 04.07.2013], [патент RU2656836, опубл. 06.06.2018], [патент CN103957264, опубл. 30.07.2014].

Предложенная система может быть реализована схемой, представленой на рис. 2. Исходное значение X поступает на входы блоков $mod\, p_i$ нахождения остатков по модулю p_i, $i \in [1, n+1]$, которые могут быть выполнены как с использованием вычислительных устройств, например, интегральных схем или FPGA, так и в виде памяти, которая в ответ на значение исходного числа X подает на выход блока $mod\, p_i$ остаток x_i от деления на модуль p_i. При этом модули удовлетворяют условиям $p_1 < p_2 < p_3 < \cdots < p_n < p_{n+1}$ и $p_{n+1} > p_n \cdot p_{n-1}$.

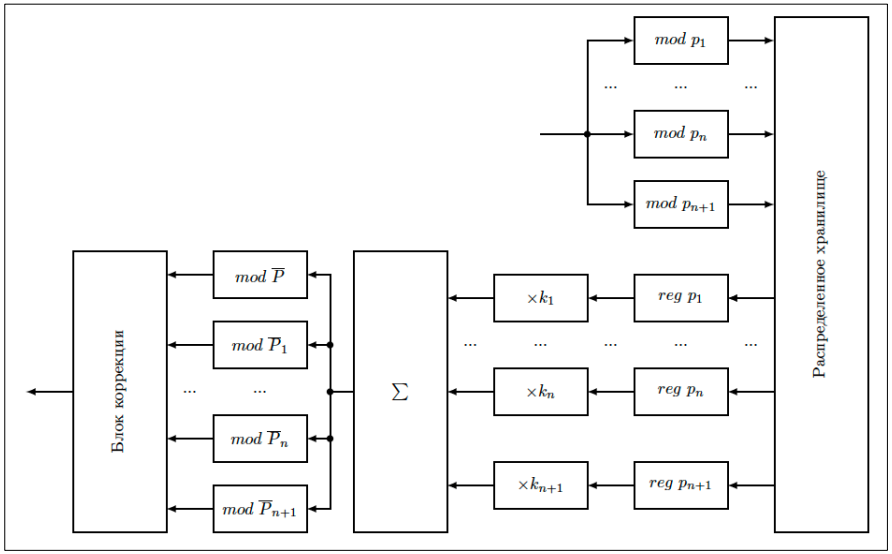

Рис. 2. Схема системы распределенного хранения данных
Fig. 2. Block diagram of the distributed data storage system

Данные с выходов блоков нахождения остатков по модулю p_i передаются в распределенное хранилище, которая может быть использована как для хранения, так и обработки данных, поскольку особенностью системы остаточных классов является возможность выполнения арифметических операций сложения и умножения независимо по каждому модулю. При этом распределенное хранилище может быть представлено одним или несколькими облачными провайдерами, или внутренними частными облаками/хранилищами, что позволит распределить данные при хранении, тем самым повысить надежность восстановления данных в случае выхода из строя одного или нескольких хранилищ за счет избыточной структуры системы остаточных классов.

После хранения данные поступают из распределенного хранилища на входы регистров хранения остатков по модулю p_i, выходы которых соединены со входами соответствующих блоков умножения на $k_i = w_i \bar{P}_i$, выходы которых подключены к входами сумматора произведений, значение суммы поступает на входы блока нахождения остатков по модулю $\bar{P}$ и блоков нахождения остатков по модулю $\bar{P}_i$, результаты с которых поступают на вход блока коррекции ошибки, выход которого является выходом системы.

На рис. 3 показана структурная схема блока коррекции ошибки. Значение $|S|_{\bar{P}}$ поступает на блок сравнения $|T|_{\bar{P}}$ с рабочим диапазоном, в котором проверяется логическое выражение $|S|_{\bar{P}} < P$, если оно истинно, то сигнал с выхода блок сравнения $|S|_{\bar{P}}$ с рабочим диапазоном поступает на управляющий вход мультиплексора, пропуская корректное значение $|S|_{\bar{P}}$ через первый информационный вход на выход блока коррекции ошибки. Второй информационный вход мультиплексора $|S|_{\bar{P}}$ подключен к выходу мультиплексора $|S|_{\bar{P}_1}$.

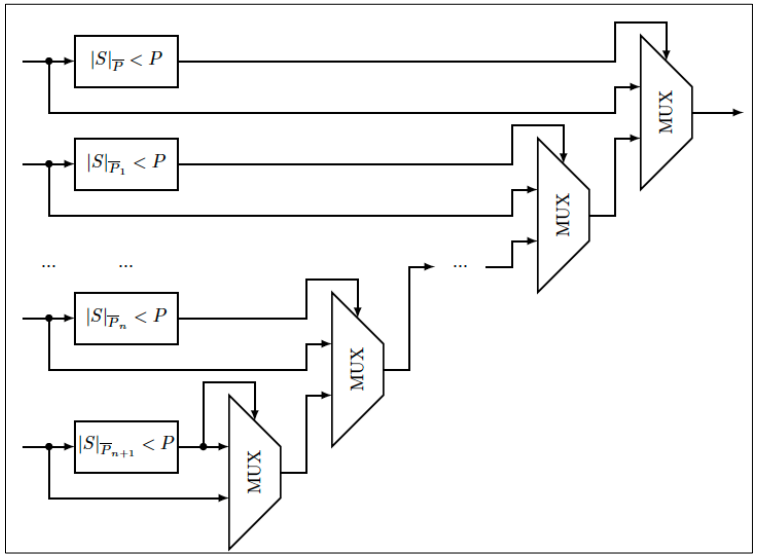

Рис. 3. Структурная схема блока коррекции
Fig. 3. Block diagram of the correction unit

Значения $|S|_{\bar{P}_i}$ с выходов блоков нахождения остатков по модулю $\bar{P}_i$ поступают на входы соответствующих блоков сравнения $|S|_{\bar{P}_i}$ с рабочим диапазоном, в которых проверяются логические выражения $|S|_{\bar{P}_i} < P$, если оно не выполняется, т.е. равно 0, то сигнал 0 с выхода блок сравнения $|S|_{\bar{P}_i}$ с рабочим диапазоном поступает на управляющий вход мультиплексора $|S|_{\bar{P}_i}$, на первый информационный вход которого подается значения $|S|_{\bar{P}_i}$ с выхода блока нахождения остатков по модулю $\bar{P}_i$. Выход мультиплексора $|S|_{\bar{P}_i}$ подключен ко второму информационному входу мультиплексора $|S|_{\bar{P}_{i-1}}$. Второй информационный вход мультиплексора $|S|_{\bar{P}_{n+1}}$ подключен к выходу блок сравнения $|S|_{\bar{P}_{n+1}}$.

Блок коррекции ошибки осуществляет вывод первого значения $|S|_{\bar{P}}$ или $|S|_{\bar{P}_i}$, которое меньше рабочего диапазона.

6. Заключение

Преимуществом данного алгоритма является снижение аппаратных издержек коррекции ошибок модулярных чисел, полученных из систем распределенного хранения данных, что связано с отсутствием необходимости вычисления $S = \sum_{i=1}^{n+1} k_i x_i'$ для каждой проекции $\bar{P}_i$, S вычисляется один раз, и в дальнейшем необходимо вычисления только остатка от деления.

Реализация всей системы возможна с использованием программируемых логических интегральных схем (ПЛИС), специализированных интегральных схем, а также в виде алгоритма работы ЭВМ и может использоваться как отдельное устройство, так и как сопроцессор для выполнения подготовки файлов к надежному распределенному хранению данных.

При этом предложенный алгоритм в среднем выполняется в 1,5 раза дольше за счет последовательной архитектуры некоторых операций, но при этом площадь микросхемы в среднем в 3,5 раза меньше.

При этом стоит обратить внимание на 32-битный рабочий диапазон. В этом случае для 5 и 6 рабочих основаниях время прохождения сигнала примерно одинаковое, а площадь реализации алгоритма 1 в 6 раз больше.

Направлением дальнейших исследований может стать реализация предложенного алгоритма для криптографических алгоритмов с размерностью 64, 128, 256 бит.

Список литературы / References

[1]. Ghemawat S., Gobioff H., Leung S.T. The Google file system. In Proc. of the 18th ACM Symposium on Operating Systems Principles, 2003, pp. 29-43.
[2]. Gomathisankaran M., Tyagi A., Namuduri K. HORNS: A homomorphic encryption scheme for Cloud Computing using Residue Number System. In Proc. of the 45th Annual Conference on Information Sciences and Systems, 2011, pp. 1-5.
[3]. Lin H.Y., Tzeng W.G. A secure erasure code-based cloud storage system with secure data forwarding. IEEE transactions on parallel and distributed systems, vol. 23, issue 6, 2011, pp. 995-1003.
[4]. Celesti A., Fazio M. et al. Adding long-term availability, obfuscation, and encryption to multi-cloud storage systems. Journal of Network and Computer Applications. vol. 59, 2016, pp. 208-218.
[5]. Chervyakov N., Babenko M. et al. AR-RRNS: Configurable reliable distributed data storage systems for Internet of Things to ensure security. Future Generation Computer Systems, vol. 92, 2019, pp. 1080–1092.
[6]. Li W., Yang Y., Yuan D. A Novel Cost-Effective Dynamic Data Replication Strategy for Reliability in Cloud Data Centres. In Proc. of the IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, 2011, pp. 496–502.
[7]. Акушский И.Я., Юдицкий Д.И. Машинная арифметика в остаточных классах. М., Советское радио, 1968 г., 440 стр. / Akushsky I.Ya., Yuditsky D.I. Computer arithmetic in residual classes. Moscow, Soviet Radio, 1968, 440 p. (in Russian).
[8]. Вернер М. Основы кодирования. М., Техносфера, 2004 г., 288 стр. / Werner M. Information und Codierung. Vieweg+Teubner Verlag, Wiesbaden, 2002, 213 p.

Информация об авторах / Information about authors

Андрей Владимирович ГЛАДКОВ – младший научный сотрудник. Сфера научных интересов: высокопроизводительные вычисления, система остаточных классов.

Andrei Vladimirovich GLADKOV – Research Assistant. His research interests include high-performance computing, residue number systems.

Виктор Андреевич КУЧУКОВ – младший научный сотрудник. Сфера научных интересов: высокопроизводительные вычисления, система остаточных классов, нейронные сети, цифровая обработка сигналов.

Viktor Andreevich KUCHUKOV – Research Assistant. His research interests include high-performance computing, residue number systems, neural networks, digital signal processing.

Михаил Григорьевич БАБЕНКО – кандидат физико-математических наук. Сфера научных интересов: облачные вычисления, высокопроизводительные вычисления, система остаточных классов, нейронные сети, криптография.

Mikhail Grigoryevich BABENKO - PhD in Physics and Mathematics. His research interests include cloud computing, high-performance computing, residue number systems, neural networks, cryptography.

Андрей Николаевич ЧЕРНЫХ получил степень доктора наук в Институте системного программирования РАН. Он является профессором Центра научных исследований и высшего образования в Энсенаде, Нижняя Калифорния, Мексика. В научном плане его интересуют многоцелевая оптимизация распределения ресурсов в облачной среде, проблемы безопасности, планирования, эвристики и метаэвристики, интернет вещей и т.д.

Andrei Nikolaevitch TCHERNYKH received his PhD degree at Ivannikov Institute for System Programming of the Russian Academy of Sciences. He is holding a full professor position in computer science at CICESE Research Center, Ensenada, Baja California, Mexico. He is interesting in grid and cloud research addressing multiobjective resource optimization, both, theoretical and experimental, security, uncertainty, scheduling, heuristics and meta-heuristics, adaptive resource allocation, and Internet of Things.

Виктор Васильевич БЕРЕЖНОЙ – кандидат технических наук, доцент. Сфера научных интересов: модулярная арифметика, параллельные вычислительные системы, нейронные сети.

Victor Vasilyevich BEREZHNOY – PhD in Technical Sciences, Associate Professor. Research interests: modular arithmetic, parallel computing systems, neural networks.

Александр Юльевич ДРОЗДОВ. Доктор технических наук, профессор. Главный научный сотрудник, руководитель лаборатории, заместитель заведующего кафедр РЭПИ, ФРТК МФТИ. Основатель и руководитель лаборатории моделирования и проектирования архитектур специальных вычислительных систем МФТИ. Основатель и руководитель конструкторского цента Микроэлектроники факультета радиотехники и кибернетики МФТИ. Научные интересы: системы автоматизации проектирования, вычислительные машины, комплексы и компьютерные сети, системы автоматизации проектирования, принципы работы современных архитектур микропроцессоров, компилятора и системного ПО.

Alexander Yulievich DROZDOV. Doctor of Technical Sciences, Professor. Chief Researcher, Head of the Laboratory, Deputy Head of the Departments of REPI, FRTK MIPT. Founder and head of the Laboratory for Modeling and Designing Architectures of Special Computing Systems at MIPT. Founder and head of the Microelectronics Design Center of the Faculty of Radio Engineering and Cybernetics of the Moscow Institute of Physics and Technology. Scientific interests: design automation systems, computers, complexes and computer networks, design automation systems, operating principles of modern microprocessor architectures, compiler and system software.