

Impact Factor:

ISRA (India) = 1.344
ISI (Dubai, UAE) = 0.829
GIF (Australia) = 0.564
JIF = 1.500

SIS (USA) = 0.912
PIIHЦ (Russia) = 0.234
ESJI (KZ) = 1.042
SJIF (Morocco) = 2.031

ICV (Poland) = 6.630
PIF (India) = 1.940
IBI (India) = 4.260

SOI: [1.1/TAS](#) DOI: [10.15863/TAS](#)

International Scientific Journal Theoretical & Applied Science

p-ISSN: 2308-4944 (print) e-ISSN: 2409-0085 (online)

Year: 2016 Issue: 10 Volume: 42

Published: 30.10.2016 <http://T-Science.org>

O.A. Aivazian

post graduate student,
department "Information design technologies",
National Polytechnic University of Odessa
ogannes701@gmail.com

SECTION 4. Computer science, computer engineering and automation.

NON-BINARY CODING METHOD BY SEQUENTIALLY DETERMINED INFORMATIONAL BLOCKS

Abstract: Author presents a model and a method for non-binary coding of information. The system that solves the problem of non-binary coding consists of informational blocks determined in the certain consecutively situated blocks and their interrelated vectors, which form angles relatively to the axis. The paper describes the algorithm of coding and decoding of information using sequence of useful bits and added screening information to the angular relations. Paper considers an example of the encoding operation when we apply a random sequence of non-binary symbols to the input of the encoder in a decimal form. Proposed method supposed for further program and hardware non-binary codes application.

Key words: non-binary coding, angles ratios, sequential informational blocks.

Language: Russian

Citation: Aivazian OA (2016) NON-BINARY CODING METHOD BY SEQUENTIALLY DETERMINED INFORMATIONAL BLOCKS. ISJ Theoretical & Applied Science, 10 (42): 70-76.

Soi: <http://s-o-i.org/1.1/TAS-10-42-16> **Doi:**  <http://dx.doi.org/10.15863/TAS.2016.10.42.16>

НЕДВОИЧНОЕ КОДИРОВАНИЕ МЕТОДОМ ПОСЛЕДОВАТЕЛЬНО ЗАДАННЫХ ИНФОРМАЦИОННЫХ БЛОКОВ

Аннотация: Представлены модель и метод недвоичного кодирования информации, где проверочная информация определяется при соотношении последовательно расположенных блоков и их углов относительно оси. Предложен алгоритм кодирования и декодирования информации в котором задана ограниченной последовательности полезных битов и добавлена проверочная информация за счёт векторов, которые образуют угловые соотношения с осью. Рассматривается режим кодирования с примером работы при подаче на вход кодирующего устройства некоторой последовательности недвоичных символов в десятичном виде. Предложенный метод является основой для дальнейших исследований программного и аппаратного применения недвоичных кодов.

Ключевые слова: недвоичное кодирование, угловая мера, последовательность информационных блоков

Введение

Особенности построения системы кодирования и декодирования и размерность системы счисления могут оказать существенное влияние как на функционирование устройств обработки, так и на их технической реализацию. Реализация в ближайшем будущем многоустойчивых элементов выдвинет на передний край проблему разработки прикладной теории многозначных функций алгебры логики, что следует из [1,2]. Проведённый анализ [3-6] показал, что к настоящему времени нет надёжных способов обнаружения и исправления ошибок в недвоичной реализации, вследствие чего откладывается их аппаратная реализация.

Развитие телекоммуникационных технологий делает чрезвычайно актуальной задачу нахождения новых эффективных путей в реализации методов недвоичного кодирования информации. Так, в работе [7] изложен подход кодирования методом однозначно заданных блоков. При разработке новых алгоритмов основное внимание уделяется эффективности и трудоёмкости алгоритмов кодирования и декодирования. Согласно [6,8] такие алгоритмы необходимы при проектировании экспертных систем, принятии сложных логических решений, аналитическом представлении изображений и их обработке, синтезе и анализе дискретных (абстрактных) автоматов в решении задач



Impact Factor:

ISRA (India) = 1.344	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	РИИЦ (Russia) = 0.234	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 1.042	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 2.031	

целочисленной оптимизации, обработки больших многомерных данных, криптографических систем и т.д.

Целью работы является разработка модели и описание алгоритма недвоичного помехоустойчивого кода, который кодирует информацию, добавляя проверочную часть, затем в процессе декодирования выполняет проверку на возникшие ошибки.

1. Недвоичное представление кода

В [9] даётся правило, что основополагающими в кодировании информации являются два свойства – свойство избыточности, предполагающая использование избыточных символов для каждого сообщения, и свойство усреднения шума, предполагающего зависимость избыточного символа от нескольких информационных символов.

Модель недвоичного кодирования информации можно представить в виде последовательности информационных блоков (ПИБ), каждый из которых имеет одно независимое состояние. ПИБ порядка n представляет собой равномерно квадрированный квадрат [10], в котором задана последовательность кодовых символов от 0 до n^2 . В задачах недвоичного кодирования важно построить быстрые алгоритмы кодирования и декодирования при работе с достаточно

большими многозначными данными. Предположим, на вход кодирующего устройства будет подаваться кодовые символы длиной L . При поступлении информационных слов необходимо задать некоторую избыточную информацию, за счёт которого будет происходить дальнейшее обнаружение и исправление ошибок. При построении ПИБ каждый блок однозначно определён. Для того чтобы задать проверочную информацию относительно каждого информационного блока (ИБ) целесообразно выделить группы блоков с определёнными признаками. Если спроецировать вектор из точки пересечения блоков 0, 1, 2, 3 в направлении любых других блоков, на проекции заданного вектора будут находиться определённые блоки. На рис. 1 представлен ПИБ размерностью $n=8$, и спроецирован вектор над блоками {1, 9, 25, 49}, угловая мера (угол между вектором и осью Ox)

А которой равна 45° . Учитывая, что несколько ИБ могут задаваться одним проверочным значением, такой способ добавления избыточной информации позволит построить более быстрые алгоритмы кодирования и декодирования. При этом ИБ, заданные под одним и тем же углом находятся на больших кодовых расстояниях друг от друга, что позволит более эффективно разработать алгоритм декодирования.

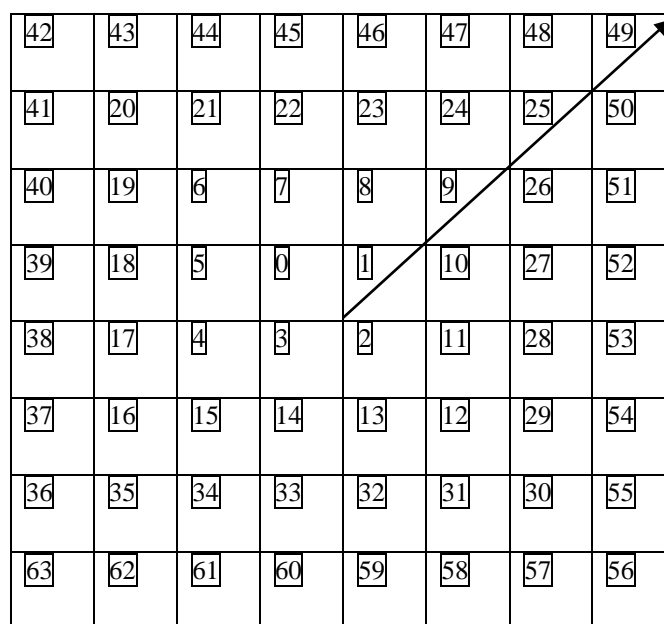


Рисунок 1 – Вектор с $A=45^\circ$ над ПИБ ($n=8$).

В данном методе вектор задаются над теми информационными блоками (ИБ), соответствующие значения S которых поданы на вход кодера.

Определение 1. Информационный S -блок –

определённый блок информации, принимающий на входе некоторый двоичный код, и преобразующая их в десятичный код с присвоением угловой меры. В обобщённом виде с учётом рис.1 соотношения S -блоков и A можно

Impact Factor:

ISRA (India) = 1.344	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	ПИИЦ (Russia) = 0.234	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 1.042	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 2.031	

выразить в виде (1) и (2):

$$\forall S_1, S_2, \dots, S_L \exists A_i, \quad (1)$$

$$\forall A \exists \min n(S) = 1$$

$$\forall A \exists \max n(S) = \frac{n}{2}, \quad (2)$$

где $\min n(S), \max n(S)$ – минимальное, максимальное количество S соответственно.

Из (1) следует – каждый ИБ обладает своей угловой мерой.

Из (2) следует – каждой A соответствует минимум один S , максимум – $\frac{n}{2} S$.

2. Описание метода кодирования

Для удобства представления кодируемой информации в соответствии с рис. 1 при подаче на вход кодировщика заданную последовательности символов преобразуем в десятичный вид. В случае если на вход кодировщика подаётся двоичная

последовательность, то каждый такт m работы кодировщика будет состоять из восьми символов. Тогда максимально возможная последовательность одного такта m будет иметь

вид 1111 1111, что даёт 255_{10} . С учётом этого и основываясь на (1), (2) обобщённая модель для кодирования ПИБ с $n=16$, в котором максимальное количество ИБ составляет 256 описана в виде определения 2.

Определение 2.

- 1) Углы A образуют кодовые подмножества в которых элементами являются значения информационных S – блоков, которые загружены в память кодирующего устройства. В каждое кодовое подмножество входит заданное количество элементов.
- 2) Минимальное количество ИБ соответствующих некоторому A равно 3 значениям ИБ.
- 3) Одно и то же S не повторяется для разных A . Соответствия множеств значений S для каждого A показаны в (3):

$A = 135 \rightarrow \{0,6,20,42,72,110,156,210\}$; $A = 45 \rightarrow \{1,9,25,49,81,121,169,225\}$;
 $A = 315 \rightarrow \{2,12,30,56,90,132,182,240\}$; $A = 225 \rightarrow \{3,15,35,63,99,143,195,255\}$;
 $A = 175 \rightarrow \{68,105,150,203\}$; $A = 170 \rightarrow \{39,151,204\}$; $A = 165 \rightarrow \{5,18,106\}$;
 $A = 160 \rightarrow \{69,152,205\}$; $A = 155 \rightarrow \{40,107,206\}$; $A = 150 \rightarrow \{19,70,108,153,207\}$;
 $A = 145 \rightarrow \{41,71,154,208\}$; $A = 140 \rightarrow \{109,155,209\}$; $A = 130 \rightarrow \{73,111,157,211\}$;
 $A = 125 \rightarrow \{21,43,158,212\}$; $A = 120 \rightarrow \{74,112,159,213\}$; $A = 115 \rightarrow \{44,113,214\}$;
 $A = 110 \rightarrow \{75,160,215\}$; $A = 105 \rightarrow \{22,114,161\}$; $A = 100 \rightarrow \{7,45,216\}$;
 $A = 95 \rightarrow \{76,115,162,217\}$; $A = 85 \rightarrow \{77,116,163,218\}$; $A = 80 \rightarrow \{8,46,217\}$;
 $A = 75 \rightarrow \{23,117,164\}$; $A = 70 \rightarrow \{78,165,220\}$; $A = 65 \rightarrow \{47,118,221\}$;
 $A = 235 \rightarrow \{34,62,193,253\}$; $A = 60 \rightarrow \{79,119,166,222\}$; $A = 55 \rightarrow \{24,48,167,223\}$;
 $A = 50 \rightarrow \{80,120,168,224\}$; $A = 40 \rightarrow \{122,170,226\}$; $A = 35 \rightarrow \{50,82,171,227\}$;
 $A = 30 \rightarrow \{26,83,123,172,228\}$; $A = 25 \rightarrow \{51,124,229\}$; $A = 20 \rightarrow \{84,173,230\}$;
 $A = 15 \rightarrow \{10,27,125\}$; $A = 10 \rightarrow \{52,174,231\}$; $A = 5 \rightarrow \{85,126,175,232\}$;
 $A = 355 \rightarrow \{86,127,176,233\}$; $A = 350 \rightarrow \{53,177,234\}$; $A = 345 \rightarrow \{11,28,128\}$;
 $A = 340 \rightarrow \{87,178,235\}$; $A = 335 \rightarrow \{54,129,236\}$; $A = 330 \rightarrow \{29,88,130,179,237\}$;
 $A = 325 \rightarrow \{55,89,180,238\}$; $A = 320 \rightarrow \{131,181,239\}$; $A = 310 \rightarrow \{91,133,183,241\}$;
 $A = 305 \rightarrow \{31,57,184,242\}$; $A = 300 \rightarrow \{92,134,185,243\}$; $A = 295 \rightarrow$
 $\{58,135,244\}$; $A = 290 \rightarrow \{93,186,245\}$; $A = 285 \rightarrow \{32,136,187\}$; $A = 280 \rightarrow \{13,59,246\}$;
 $A = 275 \rightarrow \{94,137,188,247\}$; $A = 265 \rightarrow \{95,138,189,248\}$; $A = 260 \rightarrow \{14,60,249\}$;
 $A = 255 \rightarrow \{33,139,190\}$; $A = 250 \rightarrow \{96,191,250\}$; $A = 245 \rightarrow \{61,140,251\}$;



Impact Factor:

ISRA (India) = 1.344	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	ПИИЦ (Russia) = 0.234	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 1.042	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 2.031	

$$\begin{aligned}
 A = 240 &\rightarrow \{97,141,192,252\}; A = 235 \rightarrow \{34,62,193,253\}; A = 230 \rightarrow \{98,142,194,254\}; \\
 A = 220 &\rightarrow \{100,144,196\}; A = 215 \rightarrow \{36,64,145,197\}; A = 210 \rightarrow \{16,65,101,146,198\}; \\
 A = 205 &\rightarrow \{37,102,199\}; A = 200 \rightarrow \{66,147,200\}; A = 195 \rightarrow \{4,17,103\}; \\
 A = 190 &\rightarrow \{38,148,201\}; A = 185 \rightarrow \{67,104,149,202\}.
 \end{aligned} \quad (3)$$

4) Учитывая заданные зависимости множеств кодовых символов от их угловых значений, можно вычислить минимальное расстояние между кодовыми блоками в выражении (4):

$$1. \quad d_0 = 2\sin(\pi/36) = 0.174 \quad (4)$$

2. Алгоритм кодирования

В литературе [11] дано определение избыточности кода, который рассчитывается по формуле (5):

$$D = k / (k + i), \quad (5)$$

$$Y_k = S_{i1} + \dots + S_{p1} + S_{i2} + \dots + S_{p2} + \dots + S_{ij} + \dots + S_{pj} = v_1(A_1) + v_2(A_2) + \dots + v_m(A_m), \quad (6)$$

где $m = 1..68$, так как всего 68 подмножеств A , возможные значения $p = 3..8$ согласно (3)

Из (6) следует, что при кодировании k -тая контрольная сумма Y равна суммам выбранных S и соответствующим им кодовым подмножествам A .

Пример: Пусть по каналу на вход кодирующего устройства подаётся информационное сообщение $u_1u_2u_3u_4u_5u_6$ в виде {0000 0011 0000 0110 0000 1110 0010 1010 0110 0100 1001 1100}. Закодируем $u_1u_2u_3u_4u_5u_6$ в двоичное сообщение вида $w_1w_2\dots w_m$. Каждый двоичный кодовый символ выражается восьмью двоичными, для этого двоичное сообщение делится на блоки по 8 символов в каждом. Основываясь на (3), (6) запишем метод в виде шагов с примером. *Шаг 1:* с учётом группирования подмножеств (3), формируются двоичные кодовые символы. Всего задействованы 6 двоичных символов, которые принадлежат трём подмножествам A . Вычислим в (7) суммы для каждого v_i и их общую сумму:

$$\begin{aligned}
 v_1 &: 3 + 99 = 102; \\
 v_2 &: 6 + 42 + 156 = 204; \\
 v_3 &: 14;
 \end{aligned} \quad (7)$$

Шаг 2: Вычислим в (8) контрольную сумму по выражению (6):

где k — количество проверочных единиц измерения информации, i — количество информационных единиц измерения информации.

В представленном методе кодирования избыточность для каждого блока может варьироваться в зависимости от того как расположены ИБ.

Обобщённая формула для контрольной суммы m кодовых слов задана в (6):

$$Y_1 = v_1 + v_2 + v_3 = 320; \quad (8)$$

Шаг 3: Зададим в (9) количество кодовых символов в каждом v_i :

$$n_1 = 2, n_2 = 3, n_3 = 1; \quad (9)$$

Шаг 4: Сформируем информационное сообщение вида $w_1w_2\dots w_m$ заданное в виде (10):

$$w_i = v_i A_i n_i, \text{ где } i = \overline{1, m} \quad (10)$$

$n = 1$ не добавляется в кодовое слово. Тогда, в соответствии с примером закодированное сообщение $w_1w_2w_3$ имеет вид:

102	225	2	204	135	3	14	260
v	A	n	v	A	n	v	A

Схематически такой код можно отображён в виде рис. 2.

По выражению (5) вычислим избыточность заданного кода в (11):

$$D = 5 / (5 + 6) = 45\% \quad (11)$$

Вычислим в (12) максимально и минимально возможные избыточности кода по выражению (5), тогда D_{Min} получим в случае $n(v_i) = 8$, D_{Max} — в случае $n(v_i) = 1$:

Impact Factor:

ISRA (India) = 1.344	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	ПИИИ (Russia) = 0.234	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 1.042	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 2.031	

$$D_{Min} = 2/10 = 20\% \quad (12)$$

$$D_{Max} = 1/2 = 50\%$$

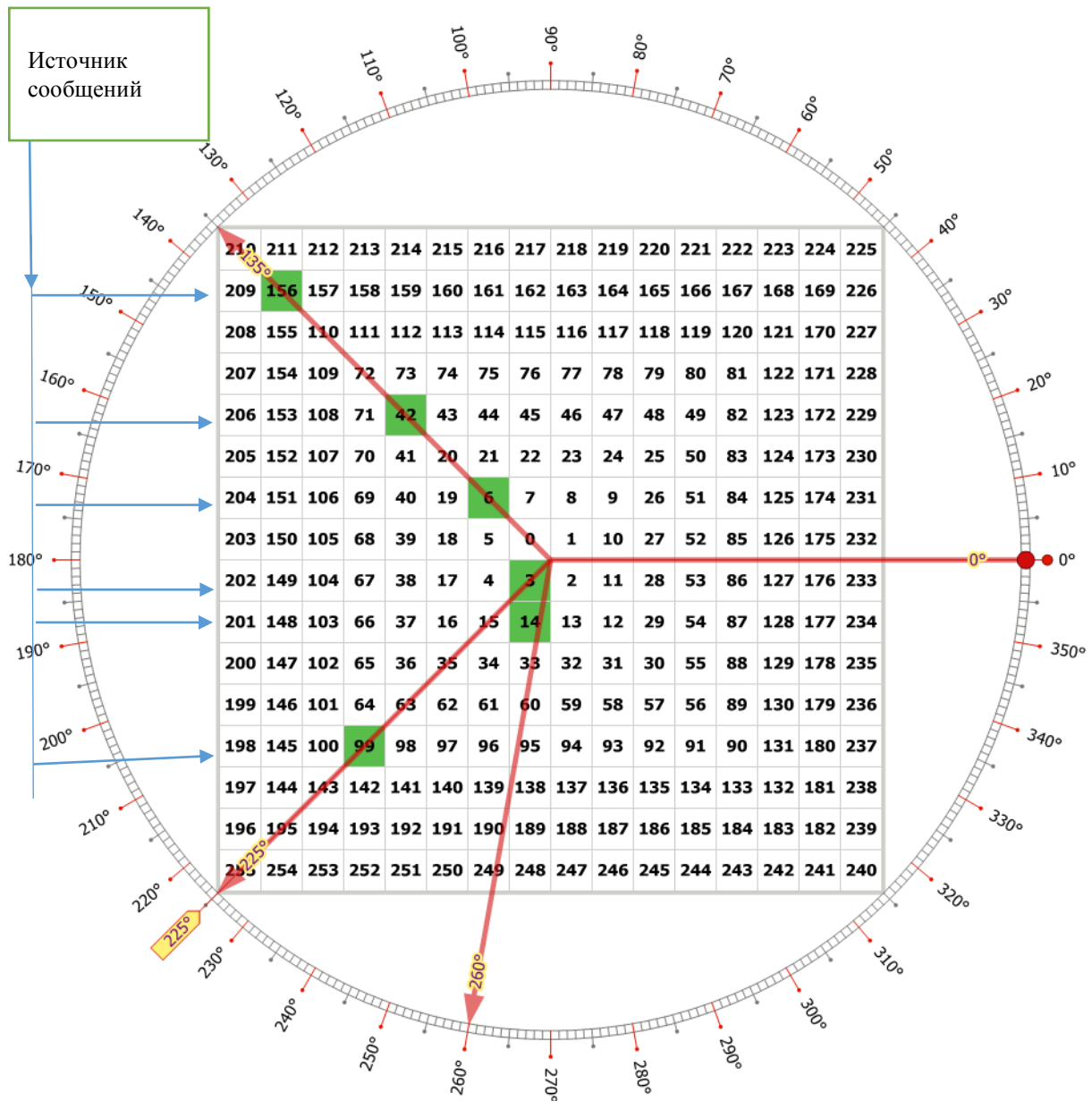


Рисунок 2 – ПИБ (256) с обозначением множества кодируемых S и соответствующим им углом

3. Декодирование и исправление ошибок

Алгоритм декодирования предполагает вычисление каждого V_i из полученного сообщения $w_1 w_2 \dots w_m$. Таким образом, декодирование сводится к задаче о сумме подмножеств. [12,13]. Вычисление искомого S_i из V_i представляет собой задачу о сумме подмножеств, для решения которой используется

множество алгоритмов. С учётом того что скорее всего декодируемое сообщение будет содержать ошибки вычислительные эксперименты, проведённые на ЭВМ позволили установить, что наиболее оптимальным по надёжности является решение с помощью метода brute force, который подробно описан в [12]. С учётом того что каждое подмножество A содержит от 3 до 8 элементов, как показано в (3), удаётся однозначно находить решение.

Impact Factor:

ISRA (India) = 1.344	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	РИИЦ (Russia) = 0.234	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 1.042	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 2.031	

Рассмотрим необходимое условие точного исправления ошибок. Тогда отклонение β от закодированной суммы v_i должно быть меньше минимального элемента S из некоторого подмножества A_i . Таким образом, порог исправления ошибок β задаётся неравенством (13):

$$\beta < \min S_i(A_i), \quad (13)$$

Опишем метод декодирования с внесением ошибок удовлетворяющих неравенству (13) в виде шагов. *Шаг 1.* Декодер получает сообщение вида $w_1 w_2, \dots, w_m$, как задано в конечном *Шаге 4* алгоритма кодирования. *Шаг 2.* Выразим внесение ошибки в v_i , в результате чего получим t_i в (14):

$$v_i \pm \beta_i = t_i, \quad (14)$$

В качестве примера основываясь на (7) и (13) внесём ошибки $\beta=1$ или 2 в v_1 и промоделируем на ЭВМ средствами программирования процесс декодирования. Выразим результат в (15):

$$t_1 = 3 + 99 + 2 = 104 \Rightarrow 3 + 99$$

$$t_1 = 3 + 99 + 1 = 103 \Rightarrow 3 + 99 \quad (15)$$

Шаг 3. Из декодированных кодовых слов v_i можно обратно получить сообщение в двоичном виде u_i , каждое из которых выражено восьмью двоичными.

Заключение

Предложен метод недвоичного помехоустойчивого кодирования с представлением информации в виде последовательно заданных информационных блоков. Представлен простой и алгоритм кодирования с расчётом избыточности, где результат кодирования – сумма элементов из одного кодового подмножества. Описан метод быстрого декодирования, основывающийся на задаче о сумме подмножеств. Внесённые ошибки позволили установить, что в соответствии со способом декодирования, максимальный порог ошибки равен минимальному элементу из подмножества. Алгебраическое представление кода и его модернизация являются предметом дальнейших исследований.

Предложенный метод может быть использован как для повышения эффективности разработки недвоичных кодов, так и для тестирования в криптографии, аппаратной и дальнейшей программной реализации.

References:

1. Kuharev GA (1990) Algorithms and systological processors for many-valued data processing / G.A. Kuharev, V.P. Shmerko, Y. N. Zaytseva – Minsk: Navuka i tehnika, 1990
2. Korolev AI (2002) Codes and impairments steady devices coding / A.I. Korolev – Minsk: 2002 – pp. 286
3. Blahut R (1986) Theory and practice of error control codes / R. Blahut — Moscow: Mir, 1986 – 576 p.
4. Verner M (2004) Information and Codes / M. Verner — Moscow : Tekhnosfera, 2004 — 286 p.
5. MacWilliams FJ (1979) Theory of Error-Correcting Codes / F.J. MacWilliams, N.J.A. Sloane. — Moscow: Svjaz', 1979. — pp. 745.
6. Sklar B (2003) Digital communications fundamentals and applications / B. Sklar – Moscow: Vilyams, 2003 —1104 p.
7. Aivazian O (2016) Non-binary codes complex modeling provided in the squares form and rotation vector / O.A. Aivazian, Y. D. Ivanov: Proceedings of modern information and electronic technologies — Odessa, 2016
8. Proakis J (2000) Digital Communication / J. Proakis, M. Salehi — M.: Radio i svjaz', 2000 – 800 p. — ISBN 5-256-01434-X.
9. Clark G (1987) Jr. / Error-Correction coding for digital communications / G. C. Clark, Jr., J. B. Cain — Moscow: Radio i svjaz', 1987. — 392 p.
10. Honsberger R (2015) Squaring the square [Electronic Resource] / University of Waterloo, Faculty of Mathematics, – 2015 –



Impact Factor:

ISRA (India) = 1.344	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	PIHHI (Russia) = 0.234	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 1.042	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 2.031	

<http://www.math.uwaterloo.ca/navigation/ideas/articles/honsberger2/index.shtml>

- Piterson W (1976) Error-correcting codes / W. W. Peterson, E. J. Weldon — M.: Mir, 1976 — 583 p.
- Cormen T (2009) Chapter 35.5: The subset-sum problem / Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest: Introduction to Algorithms, 3rd Edition. — MIT Press, 2009. — ISBN 978-0-262-03384-8.
- Garey M (1979) Computers and Intractability: A Guide to the Theory of NP-Completeness. / Michael R. Garey and David S. Johnson — W.H. Freeman, 1979. — ISBN 0-7167-1045-5.

