УДК 004.421.5

ПРОГРАММНАЯ МОДЕЛЬ СТОХАСТИЧЕСКОГО КОДЕКА $(n, k, 2^{64})$-КОДА

К.В. Агиевец[1], М.А. Кондахчан[2]
[1]Студент группы Б21-563 НИЯУ МИФИ, agievets.k.v@gmail.com
[2]Студент группы Б21-503 НИЯУ МИФИ, mikarkon@gmail.com

Аннотация. При передаче данных необходимо решать задачи обнаружения и исправления ошибок, возникающих из-за действия помех в канале связи, обеспечения секретности информации и имитозащиты. Традиционные системы передачи данных эти задачи решают с использованием соответственно помехоустойчивого кодирования, шифрования, формирования (на стороне отправителя) и проверки (на стороне получателя) криптографического контрольного кода целостности (имитовставки). По этой причине они громоздкие и недостаточно эффективные.

Целью данной работы является программная реализация схемы передачи данных, обеспечивающей универсальную защиту передаваемых данных, которая решает перечисленные выше задачи, обеспечивая при этом заданную вероятность правильного приема информации в случае случайных ошибок в канале связи.

Полученные результаты: программно реализован алгоритм работы $(8, 4, 2^{64})$-кода, проведено тестирование процессов кодирования, прямого и обратного стохастического преобразования и декодирования в случае ошибок различной кратности.

Ключевые слова: стохастическое кодирование, генератор псевдослучайных чисел, стохастическое преобразование, преобразованный канал связи.

Введение

Свойства реальных каналов связи многообразны и редко соответствуют модели двоичного симметричного канала. В итоге невозможно рассчитать вероятности наступления событий, приводящих к ошибкам декодирования. Для того, чтобы обеспечить наперед заданную вероятность правильного приема информации, необходимо создать преобразованный (виртуальный) дискретный канал. Ни один классический код не сможет работать при использовании виртуального канала, так как при стохастическом преобразовании теряются свойства кода. Следовательно, необходим новый код, учитывающий факт наличия преобразованного канала связи. Главным результатом применения стохастического метода кодирования данных является решение всех трех задач защиты информации: обнаружения и исправления ошибок, возникающих из-за действия помех в канале связи, обеспечения секретности информации и защиту от навязывания ложных данных (имитозащиту) в рамках одного алгоритма обработки информации [1, 2].

Стохастический (8, 4, 2^{64})-код

Теория кодирования и криптография преследуют противоположные цели. При кодировании сообщение представляется в виде, который допускает в процессе передачи данных появления некоторого количества наиболее вероятных ошибок. Можно считать, что при кодировании ясность сообщения повышается. При шифровании же уменьшается ясность сообщения, скрывается его смысл от злоумышленника. Однако, совмещая теорию кодирования и криптографию, можно решить вышеперечисленные задачи защиты информации, используя один алгоритм.

Использование криптографии позволяет создать Q-ичный (в данной работе $Q = 2^{64}$) симметричный канал передачи данных, при этом используется стохастический код, работающий с Q-ичными символами.

Существует целое семейство стохастических (n, k, Q)-кодов, разработанных С.А. Осмоловским, для каналов с различными вероятностями возникновения помех [1]. В данной работе рассматривается (8, 4, 2^{64})-код, при этом блоки стохастического преобразования реализованы с использованием криптоалгоритма Магма [3, 4].

Принцип работы стохастического кода при обнаружении и исправлении ошибок схож с тем, который используется в современной теории линейных двоичных кодов. Он описывается следующим образом [1, 5, 6]:

- выявление факта искажения информации,
- локализация неискаженных Q-ичных символов,
- стирание искаженных символов,
- их восстановление на основании выполнившихся проверочных соотношений.

За основу взят двоичный (8, 4)-код Хэмминга с дополнительной проверкой на четность и минимальным кодовым расстоянием $d = 4$, позволяющий при двоичной реализации одновременно исправлять одну ошибку и обнаруживать две. Формирование проверочных соотношений задается матрицей $\boldsymbol{H}_{8,\,4}$:

$$\boldsymbol{H}_{8,\,4} = \begin{vmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{vmatrix}.$$

Сообщение

$$a = a_1 a_2 a_3 a_4$$

кодируется в кодовое слово

$$x = x_1x_2x_3x_4x_5x_6x_7x_8 = a_1a_2a_3a_4b_1b_2b_3b_4,$$

где a_i и b_i – соответственно информационные и проверочные Q-ичные символы. Принцип формирования избыточных Q-ичных символов:

$$b_1 = a_2 \oplus a_3 \oplus a_4,$$
$$b_2 = a_1 \oplus a_3 \oplus a_4,$$
$$b_3 = a_1 \oplus a_2 \oplus a_4,$$
$$b_4 = a_1 \oplus a_2 \oplus a_3.$$

Ниже представлены четыре проверочных соотношения (1)-(4), задаваемые матрицей $\boldsymbol{H}_{8,\,4}$, и все их линейные комбинации (5)-(15)

$$a_2 \oplus a_3 \oplus a_4 \oplus b_1 = 0, \quad (1)$$
$$a_1 \oplus a_3 \oplus a_4 \oplus b_2 = 0, \quad (2)$$
$$a_1 \oplus a_2 \oplus a_4 \oplus b_3 = 0, \quad (3)$$
$$a_1 \oplus a_2 \oplus a_3 \oplus b_4 = 0, \quad (4)$$
$$a_1 \oplus a_2 \oplus b_1 \oplus b_2 = 0, \quad (5)$$
$$a_2 \oplus a_3 \oplus b_2 \oplus b_3 = 0, \quad (6)$$
$$a_3 \oplus a_4 \oplus a_3 \oplus b_4 = 0, \quad (7)$$
$$a_1 \oplus a_3 \oplus b_1 \oplus b_3 = 0, \quad (8)$$
$$a_1 \oplus a_4 \oplus b_1 \oplus b_4 = 0, \quad (9)$$
$$a_2 \oplus a_4 \oplus b_2 \oplus b_4 = 0, \quad (10)$$
$$a_4 \oplus b_1 \oplus b_2 \oplus b_3 = 0, \quad (11)$$
$$a_3 \oplus b_1 \oplus b_2 \oplus b_4 = 0, \quad (12)$$
$$a_2 \oplus b_1 \oplus b_3 \oplus b_4 = 0, \quad (13)$$
$$a_1 \oplus b_2 \oplus b_3 \oplus b_4 = 0, \quad (14)$$
$$a_1 \oplus a_2 \oplus a_3 \oplus a_4 \oplus b_1 \oplus b_2 \oplus b_3 \oplus b_4 = 0. \quad (15)$$

При декодировании сообщения проверяется выполнение этих 15-ти соотношений. (8, 4, 2^{64})-код как и любой другой (8, 4, Q)-код позволяет исправлять два стирания, т.е. если известно шесть правильных 64-разрядных символов, то остальные символы можно исправить, выразив через правильно принятые символы. Схема алгоритма декодирования представлена на рис. 1.

Пусть в процессе передаче произошла ошибка в символе a_1. Тогда выполняются соотношения под номерами (1), (6), (7), (10), (11), (12), (13), так как в них не содержится символ a_1. Следовательно, при одиночной ошибке $N_c = 7$. Искаженный 64-разрядный символ можно восстановить, воспользовавшись одним из соотношений, куда входит a_1. Например, с помощью соотношения под номером (2):

$$a_1 = a_3 \oplus a_4 \oplus b_2.$$

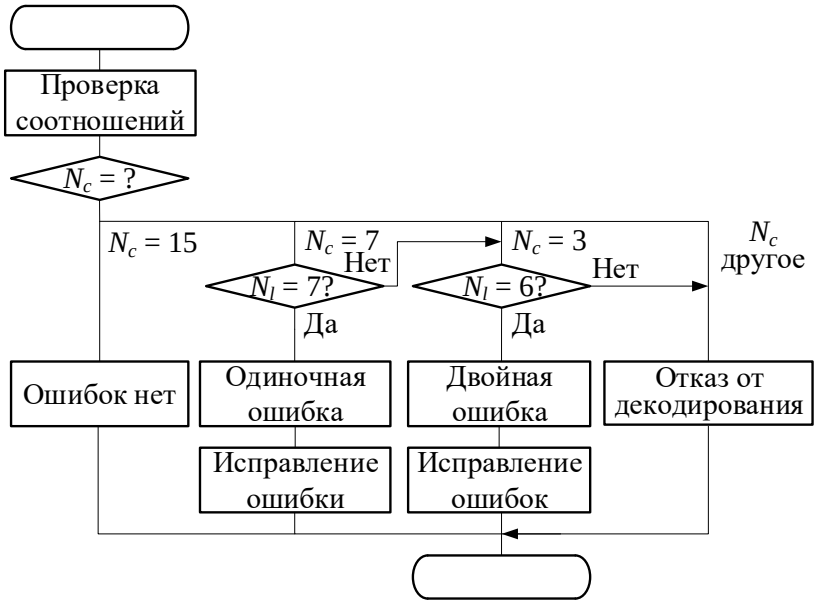

Рисунок 1 – Схема алгоритма декодирования стохастического $(8, 4, 2^{64})$-кода, где N_c – количество выполнившихся проверочных соотношений, N_l – число локализованных символов.

Пусть в процессе передачи произошла двойная ошибка и исказились символы a_1 и a_2. Точно выполняются соотношения под номерами (7), (11), (12). Однако ошибки могут взаимно уничтожиться в соотношениях, в которые a_1 и a_2 входят одновременно: (3), (4), (5), (15), при этом вероятность маскирования ошибок можно рассчитать. Соответственно, при двойной ошибке $N_c \in \{3, 7\}$. Сочетание соотношений (3), (4), (5), (7), (11), (12), (15) позволяет локализовать лишь 6 символов. Но опираться необходимо на выполнение соотношений под номерами (7), (11), (12), потому что в них не содержатся ни a_1, ни a_2. Символ a_1 после стирания можно восстановить с помощью соотношения под номером (2), а символ a_2 – с помощью соотношения под номером (1):

$$a_1 = a_3 \oplus a_4 \oplus b_2,$$
$$a_2 = a_3 \oplus a_4 \oplus b_1.$$

Рассмотрим блоки прямого и обратного стохастического преобразования (соответственно R и R^{-1}), обеспечивающие появление преобразованного канала связи с требуемыми свойствами: когда все преобразованные вектора ошибок Q-ичных символов на выходе блока R^{-1} равновероятны. На входе реального канала связи используется блок R, при этом параметр стохастического преобразования снимается с выхода генератора псевдослучайных чисел (ГПСЧ), схема которого показана на рис. 2. Совокупность блока R и ГПСЧ на стороне отправителя и блока R^{-1} и ГПСЧ на стороне получателя (рис. 2, 3) по сути образуют схему шифрования методом гаммиро-

вания, которая в отличии от традиционной на основе операции XOR не позволяет злоумышленнику вносить предсказуемые изменения в преобразованные данные (по сути шифртекст).

ГПСЧ реализован по схеме Counter Mode с использованием счетчика Q, реализованного на регистре сдвига с линейной обратной связью, и функции E зашифрования криптоалгоритма Магма [3, 4] в качестве функции выхода F_{out} генератора (рис. 2). Блоки R и R^{-1} реализуют функции E и D (соответственно за- и расшифрования) криптоалгоритма Магма (рис. 2, 3). Выбор криптоалгоритма определила требуемая разрядность Q-ичных символов. Имитозащита обеспечивается за счет избыточности, вносимой кодером.

На рис. 3 показана общая схема стохастического кодека, где e – вектор ошибок, действующий в реальном канале связи, e' – преобразованный вектор ошибок.

Результаты моделирования работы стохастического кодека

Результаты тестирования программной реализации стохастического $(8, 4, 2^{64})$-кода при внесении ошибок в информационные и проверочные Q-ичные символы различной кратности приведены на рис. 4-10.

Заключение

Реализация рассмотренной схемы передачи данных позволяет обеспечить универсальную защиту передаваемой информации в рамках единого алгоритма. Помехозащищенность достигается за счет использования стохастического $(8, 4, 2^{64})$-кода, построенного на основе (8, 4)-кода Хэмминга с дополнительной проверкой на четность. При этом за счет реализации преобразованного канала связи на основе алгоритмов, специфицированных в ГОСТ 34.12-2018, обеспечивается секретность передаваемой информации и фиксация факта умышленных ее искажений.

Перспективными направлениями для развития изложенного в данной работе решения являются:

- реализация и тестирование других стохастических (n, k, Q)-кодов;
- реализация систем передачи данных на основе кодов Рида-Соломона, БЧХ и LDPC с целью сравнения эффективности работы различных кодеков в режиме обнаружения и исправления ошибок, вызванных помехами в каналах связи;
- Light-Weight реализация преобразованного канала связи, ориентированная на использование в RFID- и IoT-системах.

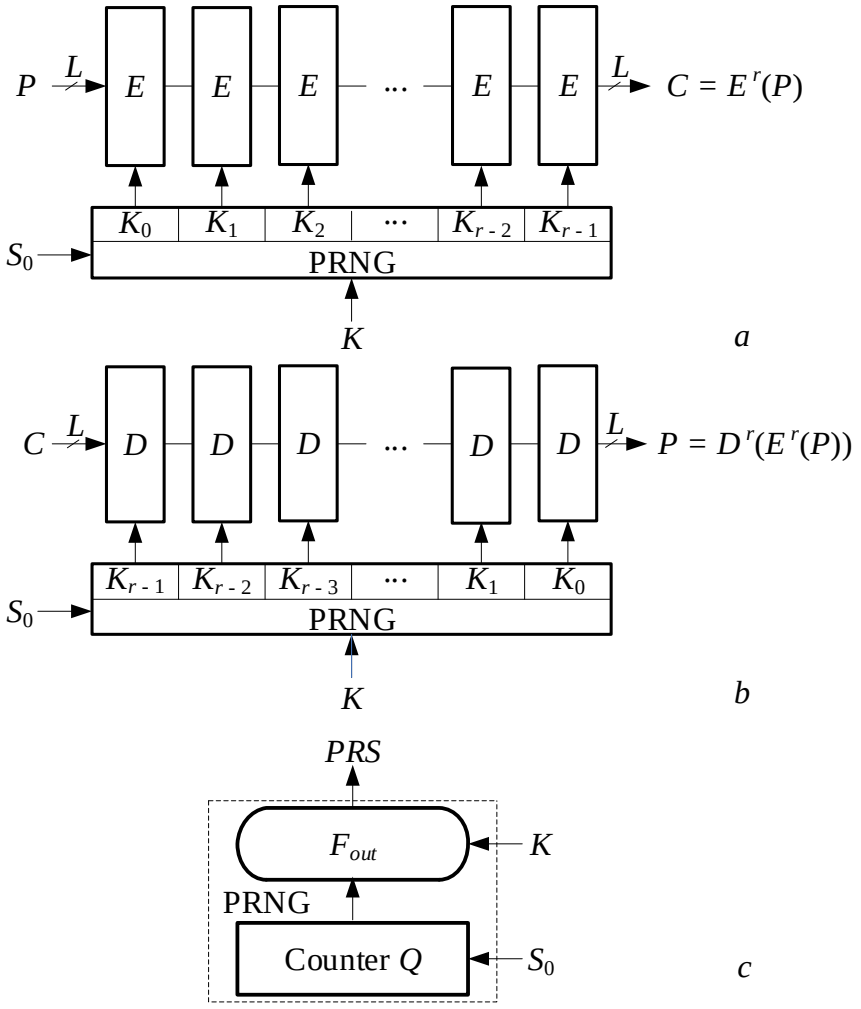

Рисунок 2 – Стохастическое преобразование: *a* – схема прямого преобразования *R*, *b* – схема обратного преобразования R^{-1}; *c* – схема ГПСЧ. PRS – Pseudo-Random Sequence, PRNG – Pseudo-Random Number Generator, *P* – Plaintext, *C* – Ciphertext, *K* – ключ, K_i – раундовые ключи, S_0 – синхропосылка.

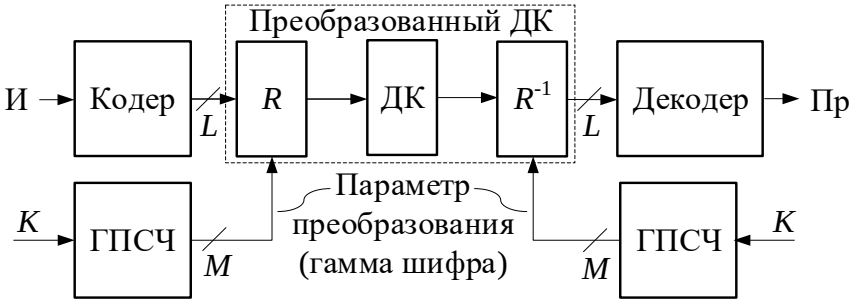

Рисунок 3 – Схема передачи данных по каналу связи с использованием стохастического кодирования. ДК – дискретный канал.
K – ключ, *L* – разрядность *Q*-ичных символов,
$M \geq L$ – разрядность параметра стохастического преобразования.

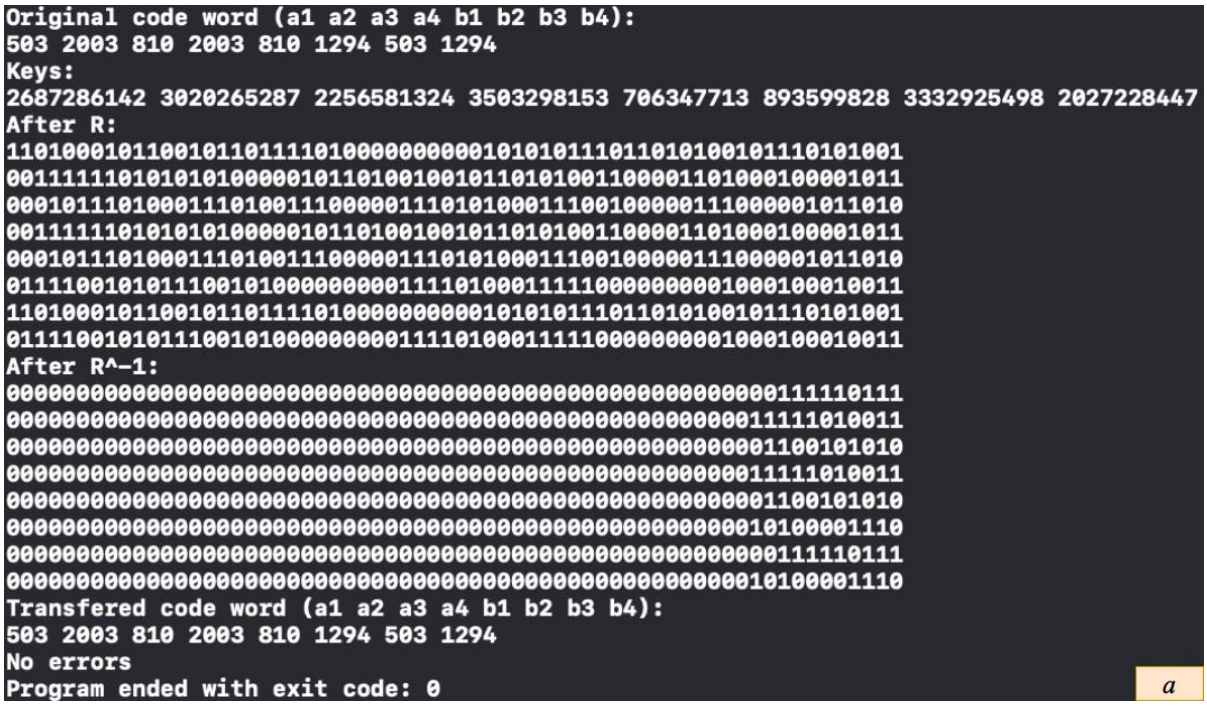

Рисунок 4 – Передача данных без ошибок.

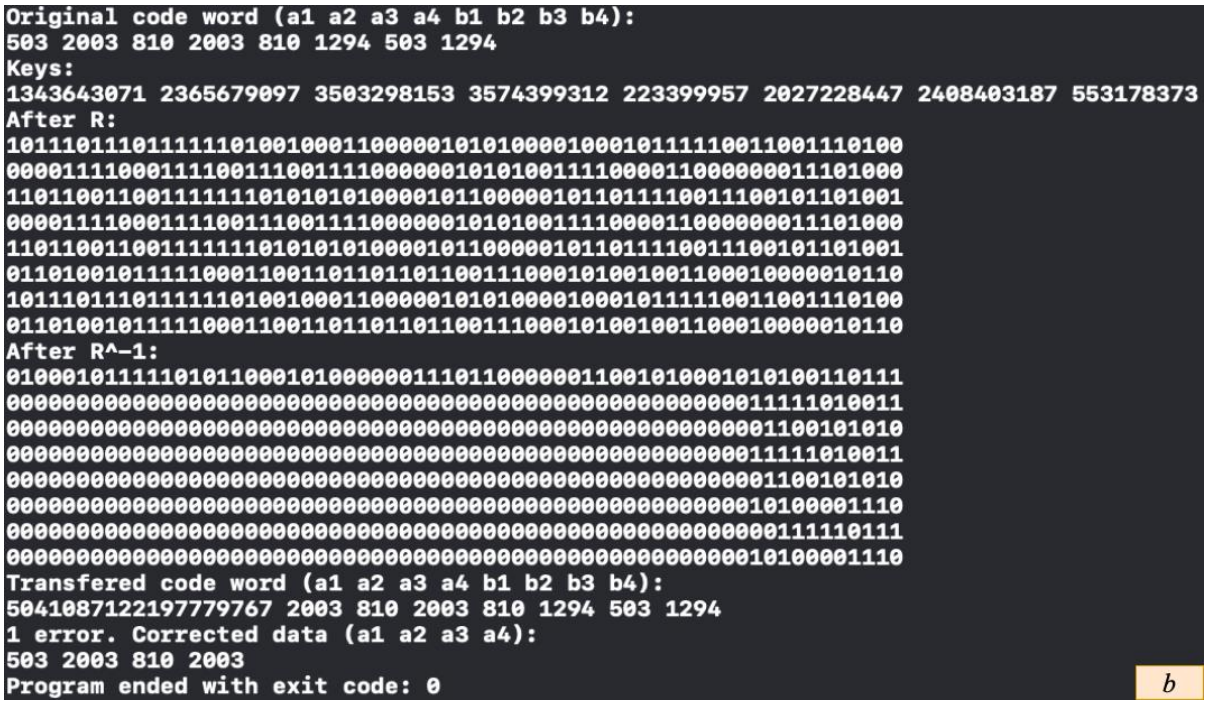

Рисунок 5 – Передача данных с ошибкой.
Исказился один информационный символ. Ошибка исправлена.

Рисунок 6 – Передача данных с ошибками.
Исказились два информационных символа. Ошибки исправлены.

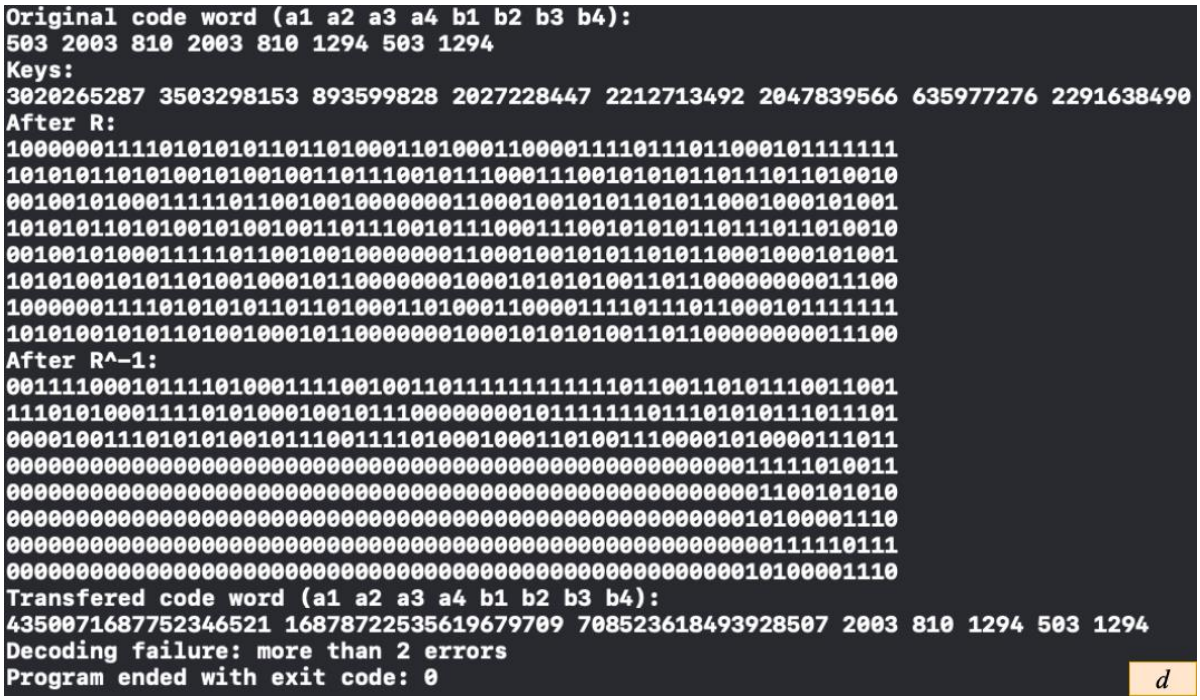

Рисунок 7 – Передача данных с ошибками,
Исказилось более двух информационных символов.
Отказ от декодирования.

```
Original code word (a1 a2 a3 a4 b1 b2 b3 b4):
503 2003 810 2003 810 1294 503 1294
Keys:
2583874451 1412695426 3332925498 2408403187 4095679132 635977276 1145819245 394827685
After R:
0000110010101110010111001001000001101010110000100011111111000000
0010100011011110010000111010000001001001010110010001111000010111
1011110001001001110010100011001101000100000101110000100011110110
0010100011011110010000111010000001001001010110010001111000010111
1011110001001001110010100011001101000100000101110000100011110110
0000011101110101011010000010101101101100011001100011011010100111
0000110010101110010111001001000001101010110000100011111111000000
0000011101110101011010000010101101101100011001100011011010100111
After R^-1:
0000000000000000000000000000000000000000000000000000111110111
0000000000000000000000000000000000000000000000000000111111010011
0000000000000000000000000000000000000000000000000000001100101010
0000000000000000000000000000000000000000000000000000111111010011
0110000011011001011111010100001111101001011011100101111111110111
0000000000000000000000000000000000000000000000000000010100001110
0000000000000000000000000000000000000000000000000000111110111
0000000000000000000000000000000000000000000000000000010100001110
Transfered code word (a1 a2 a3 a4 b1 b2 b3 b4):
503 2003 810 2003 6978746828219899895 1294 503 1294
b error, a correct
Program ended with exit code: 0
```

e

Рисунок 8 – Передача данных без ошибок в информационных символах. Исказился один проверочный символ.

```
Original code word (a1 a2 a3 a4 b1 b2 b3 b4):
503 2003 810 2003 810 1294 503 1294
Keys:
2365679097 3574399312 2027228447 553178373 2543909104 2291638490 394827685 1737984142
After R:
1010001100001000101001001001010011000001100010110100110111111011
1101111000000010101101010111000111001010110101110010000010010010
1010111011001101101101010100011111100111000011001011010011010111
1101111000000010101101010111000111001010110101110010000010010010
1010111011001101101101010100011111100111000011001011010011010111
1011010110101100000110100010110000111110110110111011110000110000
1010001100001000101001001001010011000001100010110100110111111011
1011010110101100000110100010110000111110110110111011110000110000
After R^-1:
0000000000000000000000000000000000000000000000000000111110111
0000000000000000000000000000000000000000000000000000111111010011
0000000000000000000000000000000000000000000000000000001100101010
0000000000000000000000000000000000000000000000000000111111010011
1101101100101100101000010100110001010110101100101000110010100
0001011111110010011001110010010100011110010110001010111011001010
0000000000000000000000000000000000000000000000000000111110111
0000000000000000000000000000000000000000000000000000010100001110
Transfered code word (a1 a2 a3 a4 b1 b2 b3 b4):
503 2003 810 2003 15793174628833841556 1725555016356900554 503 1294
b error, a correct
Program ended with exit code: 0
```

f

Рисунок 9 – Передача данных без ошибок в информационных символах. Исказились два проверочных символа.

```
Original code word (a1 a2 a3 a4 b1 b2 b3 b4):
503 2003 810 2003 810 1294 503 1294
Keys:
2256581324 893599828 2669322631 2047839566 3300722679 1579310740 3475968284 2842138530
After R:
0100000110001101001111011010101011010110001011111101010001011000
0000001011001011101101001100011001001000011001111110101001100001
0100001100011010001110010001111111011001001011110101110000111111
0000001011001011101101001100011001001000011001111110101001100001
0100001100011010001110010001111111011001001011110101110000111111
1101011110010000000110100000000010001100000011000010010011110111 0
0100000110001101001111011010101011010110001011111101010001011000
1101011110010000000110100000000010001100000011000010010011110111 0
After R^-1:
0000000000000000000000000000000000000000000000000000000111110111
0000000000000000000000000000000000000000000000000000001111010011
1110000010011101010100100011001011111000011010010100010011110000
0000000000000000000000000000000000000000000000000000001111010011
0000000000000000000000000000000000000000000000000000001100101010
1000100101001101101011111010011101100101111100110000010000000100
0000000000000000000000000000000000000000000000000000000111110111
0000000000000000000000000000000000000000000000000000010100001110
Transfered code word (a1 a2 a3 a4 b1 b2 b3 b4):
503 2003 16185183014708921584 2003 810 9893757280178504708 503 1294
1 error in a and 1 error in b. Corrected data (a1 a2 a3 a4):
503 2003 810 2003
Program ended with exit code: 0
```

Рисунок 10 – Передача данных с ошибками. Исказились один информационный и один проверочный символ. Ошибка исправлена.

Список литературы

1. Осмоловский С.А. Стохастические методы защиты информации. – М.: Радио и связь, 2003.
2. Иванов М.А. Способ обеспечения универсальной защиты информации, пересылаемой по каналу связи // Вопросы кибербезопасности. 2019, № 3 (31), с. 45-50.
3. ГОСТ 28147-89. Система обработки информации. Защита криптографическая. Алгоритм криптографического преобразования.
4. ГОСТ 34.12-2018. Информационная технология. Криптографическая защита информации. Блочные шифры.
5. Сагалович Ю.Л. Введение в алгебраические коды. (2011). [Электронный ресурс]. http://iitp.ru/upload/content/790/algebcodes.pdf (Дата обращения: 30.07.2024).
6. Блейхут Р. Теория и практика кодов, контролирующих ошибки. – М.: Мир, 1986.