УДК 004 : 519.673

Белецкий А.Я., Белецкий Е.А., Воливач О.И., Якимчук М.А.
Национальный авиационный университет, Киев

СИНТЕЗ И АНАЛИЗ КОДОВ РИДА-СОЛОМОНА В ПРОСТРАНСТВЕ ИЗОМОРФНОГО ИЗОБРАЖЕНИЯ

Предложен учебный вариант построения кодов Рида-Соломона, значительно упрощающий процесс синтеза кодовых слов и обнаружения-исправления ошибок в искаженных файлах данных. Алгоритм основан на переносе переменных и операций, посредством которых осуществляется синтез и анализ кодов, из пространства оригиналов в пространство изоморфного изображения. В результате предлагаемой замены пространства обработки данных вычислительный процесс становится сведенным к основным операциям в простых полях Галуа и модулярной арифметике над целочисленными операндами, легко реализуемыми средствами компьютерной техники.

Ключевые слова: *коды Рида-Соломона, поля Галуа, модулярная арифметика, изоморфные преобразования*

Введение и постановка задачи

Коды Рида-Соломона являются недвоичными блочными линейными циклическими (n,k) кодами, в которых $n-$ порядок кода, равный общему числу символов в кодовом слове, а $k-$ число информационных символов кода, определяемое соотношение $k=n-2t$, где $t-$ кратность ошибки (допустимое число ошибочных символов в кодовом слове). Кодирование с помощью кода Рида-Соломона (РС-кода) может быть реализовано двумя способами: систематическим и несистематическим [1]. При несистематическом кодировании информационное слово умножается на некий неприводимый полином в поле Галуа. Полученное закодированное слово полностью отличается от исходного и для извлечения информационного слова нужно выполнить операцию декодирования и только потом можно проверить данные на содержание ошибок. Такое кодирование требует больших затрат ресурсов на извлечение информационных данных, при этом они могут быть без ошибок. При систематическом кодировании к информационному блоку из k символов приписываются $2t$ проверочных символов. В этом случае нет затрат ресурсов при извлечении исходного блока, если информационное слово не содержит ошибок. Именно такой (систематический) способ получил наибольшее применение в практике кодирования информации и является предметом рассмотрения в данной статье.

В качестве символов РС-кода используют элементы расширенного поля Галуа $GF(2^m)$, $m\geq 2$, характеристики 2. В силу этого все преобразования (сложения или умножения) над символами кода выполняются по правилам двоичной модулярной арифметики. Любой алфавит, состоящий из множества символов (элементов), должен быть тем или иным способом упорядочен (ранжирован). Поле $GF(2^m)$ определяют над некоторым неприводимым полиномом (НП) $m-$й степени $\varphi(x)$. Множество элементов $GF(2^m)$ содержит один нулевой элемент, состоящий из m нулей, и 2^m-1 ненулевых элементов. Ненулевые элементы поля $GF(2^m)$ обычно представляют в виде степени корня α НП $\varphi_m(x)$, т.е. α^i, $i=\overline{0,2^m-2}$. Такая же форма записи элементов поля Галуа применяется, как правило, в преобразованиях, отображающих процесс кодирования и декодирования в алгоритмах Рида-Соломона, что, как «утяжеляет» сами выражения, так и их восприятие.

В данной статье вместо непосредственного представления элементов α^i РС-кодов предлагается их изоморфное отображение, причем изоморфизм устанавливается простым соответствием

$$\alpha^i \leftrightarrow i. \qquad (1)$$

Естественно, что переход от преобразований элементов поля Галуа в пространстве оригиналов к преобразованиям в пространстве изображений приводит к трансформации, как самих элементов, так и ряда операций над элементами поля. В частности, операция перемножения элементов $\alpha^i \cdot \alpha^j = \alpha^{(i+j)}$ в пространстве оригиналов заменяется операцией сложения показателей степени i и j по $\mathrm{mod}\,(2^m - 1)$ в пространстве изображений.

Изложение материала статьи иллюстрируется конкретными числовыми примерами. В частности, полагается, что длина кодового слова $n = 15$, степень m неприводимого полинома $\varphi(x)$ принята равной четырем, а число информационных символов в кодовом слове $k = 9$, т.е. допустимая кратность ошибочных символов составляет $t = 3$.

Элементы поля Галуа $GF(2^4)$

Выберем в качестве неприводимого полинома, формирующего элементы поля Галуа, примитивный полином (ПрП) четвертой степени

$$\varphi_1(x) = x^4 + x + 1. \qquad (2)$$

Элементами поля $GF(2^4)$, являются всевозможные полиномы третьей степени с коэффициентами $a_i, i = \overline{0,3}$, над $GF(2)$, т.е. $a_i \in \{0,1\}$. Следовательно, существует ровно $2^4 = 16$ различных элементов $GF(2^4)$, которые надлежит соответствующим образом ранжировать (упорядочить) и закодировать. Поле $GF(2^4)$ включает один нулевой (0000) элемент, который обозначим символом X, и $2^4 - 1 = 15$ ненулевых элементов. Почему именно мы выбрали символ X для кодирования элемента 0000, а не символ 0, будет ясно из дальнейших пояснений.

Принятым способом упорядочения ненулевые элементы представляются в виде последовательности степеней *примитивного* (образующего) элемента, формирующего мультипликативную группу максимального порядка. Если в качестве образующего элемента (ОЭ) поля $GF(2^4)$ принять полином первой степени с минимальным весом, т.е. элемент $x = 10$, тогда полином $\varphi(x)$ обязательно должен быть *примитивным*, иначе множество, содержащее степени ОЭ по $\mathrm{mod}\,\varphi$, не будет полным. Это означает, что не все ненулевые элементы поля $GF(2^4)$ будут входить в это множество.

Обратимся к полиному (2). Обозначим α корень этого полинома. *Корнем полинома* является такой элемент α, который, будучи подставленным в полином вместо переменной x, обращает его в нуль, т.е.

$$\varphi_1(\alpha) = \alpha^4 + \alpha^1 + 1 = 0. \qquad (3)$$

Из равенства (3) следует, что

$$\alpha^4 = \alpha^1 + 1,$$

которое перепишем в эквивалентной форме

$$\alpha^4 = \alpha^1 + \alpha^0, \qquad (4)$$

поскольку любое число в нулевой степени равно 1.

Теперь необходимо выбрать удобный способ кодирования элементов $\alpha^i, i = \overline{0,14}$, с помощью которых мы будем представлять все ненулевые элементы $GF(2^4)$. Остановимся на «естественном» способе кодирования, при котором α^0 и α^1 можно записать в виде

четырехбитных кодовых комбинаций 0001 и 0010 соответственно. Для элемента α^0 код 0001 действительно является общепринятым обозначением четырехбитной комбинации числа 1. Также можно считать «естественным» код 0010 для элемента α^1 хотя бы потому, что, как выше было отмечено, элемент $\alpha = \alpha^1$ выбран корнем полинома (2), которым замещается переменная x этого полинома. Но x, в свою очередь, есть минимальный полином первой степени, векторная форма которого в четырехбитной кодовой комбинации «естественно» записывается, как 0010.

Обратим внимание на то, что корень $\alpha = \alpha^1$ равен двум. Умножение на 2 в двоичной системе счисления эквивалентно сдвигу множимого на один разряд влево. «Естественно» при этом, что элемент α^2, равный $\alpha \cdot \alpha^1$, образуется в результате сдвига элемента α^1 на один разряд влево, т.е. для элемента α^2 кодом может служить двоичная комбинация 0100. И, наконец, элемент $\alpha^3 = \alpha \cdot \alpha^2$ должен быть представлен кодом 1000.

Но элемент α^4 поля $GF(2^4)$ не может быть получен сдвигом на один разряд влево элемента α^3, поскольку при этом мы выходим за пределы четырех битов. Для разрешения сложившегося «затруднения» воспользуемся равенством (4), согласно которому надлежит произвести поразрядное сложение по правилам двоичной модулярной арифметики элементов α^0 и α^1. В результате сложения приходим к коду элемента α^4, равному 0011, т.е. $\alpha^4 = 0011$.

В общем случае следует придерживаться соотношения $\alpha^{k+1} = \alpha \cdot \alpha^k$. Если при этом код элемента α^k слева содержит 0, то элемент α^{k+1} образуется сдвигом элемента α^k на один разряд влево. Если же код элемента α^k начинается с 1, то необходимо воспользоваться равенством (4).

Следуя изложенному алгоритму формирования элементов поля $GF(2^4)$ над примитивным полиномом (1), сведем эти элементы в табл. 1. При этом для упрощения записи элементов поля $GF(2^4)$ и всевозможных преобразований над ними вместо α^k будем применять их изоморфные отображения (1). В результате предлагаемой замены появляется элемент 0, соответствующий символу α^0. Вот почему элемент 0000 поля $GF(2^4)$ мы обозначили в табл. 1 через X. Будем называть элемент $X = 0000$ «пустым» элементом поля $GF(2^4)$, хотя это, может быть, и не совсем корректно, поскольку в изоморфном отображении появился «нулевой» элемент $0 = \alpha^0$.

Таблица 1.

Множество элементов g поля $GF(2^4)$ над ПрП $\varphi_1(x) = x^4 + x + 1$

g		x^3	x^2	x^1	x^0		g		x^3	x^2	x^1	x^0
X		0	0	0	0		7		1	0	1	1
0		0	0	0	1		8		0	1	0	1
1		0	0	1	0		9		1	0	1	0
2		0	1	0	0		10		0	1	1	1
3		1	0	0	0		11		1	1	1	0
4		0	0	1	1		12		1	1	1	1
5		0	1	1	0		13		1	1	0	1
6		1	1	0	0		14		1	0	0	1

Полученное множество элементов g поля $GF(2^4)$ является полным. *Полноту* множества элементов $GF(2^4)$ следует понимать в том смысле, что не существует какой-либо другой степени корня α примитивного полинома, лежащей вне интервала [0-14],

которое приводило бы к появлению нового ненулевого элемента поля. В самом деле, попробуем сформировать элемент α^{15}, который представим так: $\alpha^{15} = \alpha \cdot \alpha^{14}$. Из табл. 1 имеем $\alpha^{14} = \alpha^3 + \alpha^0$. Следовательно, $\alpha^{15} = \alpha^4 + \alpha^1$. Воспользовавшись соотношением (4), получим $\alpha^{15} = \alpha^0 = 1$. Это означает, в частности, что при любых преобразованиях степеней α их (степени) следует приводить к остатку по модулю 15.

Кроме системы кодирования элементов поля Галуа, которые мы применили для построения табл. 1, существуют и другие способы кодирования элементов. Наиболее часто используют инверсное (по отношению к принятому в табл. 1) кодирование, которое можно проследить по табл. 2.

Таблица 2.

Альтернативное кодирование элементов над ПрП $\varphi_1(x) = x^4 + x + 1$

g		x^0	x^1	x^2	x^3		g		x^0	x^1	x^2	x^3
X		0	0	0	0		7		1	1	0	1
0		1	0	0	0		8		1	0	1	0
1		0	1	0	0		9		0	1	0	1
2		0	0	1	0		10		1	1	1	0
3		0	0	0	1		11		0	1	1	1
4		1	1	0	0		12		1	1	1	1
5		0	1	1	0		13		1	0	1	1
6		0	0	1	1		14		1	0	0	1

И, тем не менее, по нашему мнению кодирование элементов, предлагаемое в табл. 1, более логично по сравнению с кодированием, приведенным в табл. 2. Принципиальное отличие двух рассмотренных способов кодирования элементов одного и того же поля $GF(2^4)$ над ПрП $\varphi_1(x) = x^4 + x + 1$, сведенных в табл. 1 и 2 соответственно, состоит в следующем. Если в табл. 1 элементы поля $GF(2^4)$ представляются полиномами третьей степени $a_3x^3 + a_2x^2 + a_1x^1 + a_0x^0$, $a_i \in \{0,1\}$, $i = \overline{0,3}$, в которых младший моном расположен справа, то в табл. 2 для того же поля Галуа элементы поля отображаются полиномами третьей степени $a_0x^0 + a_1x^1 + a_2x^2 + a_3x^3$, в которых младший моном находится слева.

А теперь вернемся к упоминавшемуся выше (например, в соотношении (2)) условию, согласно которому полином, используемый для построения элементов поля Галуа, должен быть примитивным. И это действительно так, коль скоро мы хотим получить полное множество элементов поля наиболее простым и наглядным способом. Но это вовсе не исключает возможности построения поля Галуа над неприводимым полиномом (НП), не являющимся примитивным. В таком случае в качестве корня α НП следует выбрать примитивный элемент $\omega \neq 10$ поля над этим полиномом.

Рассмотрим пример. Пусть образующим полиномом четвертой степени является НП $\varphi_2(x) = x^4 + x^3 + x^2 + x^1 + x^0$. Данному полиному отвечают восемь примитивных элементов $\omega \in \{3, 5, 6, 7, 9, A, B, E\}$, представленных 16-ричными числами, и в их числе элемент $\omega = 0111$, который мы выберем в качестве образующего элемента поля $GF(2^4)$. Приняв в качестве корня полинома $\varphi_2(x)$ значение $\alpha = \omega = 0111$, построим поле $GF(2^4)$, элементы которого сведены в табл. 3. Элемент $\alpha^{k+1}, k > 1,$ табл. 3 вычисляется по формуле

$$\alpha^{k+1} = (\alpha \cdot \alpha^k) \bmod 11111.$$

Таблица 3.

Элементы поля $GF(2^4)$ *над НП для корня* $\alpha = 0111$

g	x^3	x^2	x^1	x^0	g	x^3	x^2	x^1	x^0
X	0	0	0	0	7	1	1	1	0
0	0	0	0	1	8	1	0	1	1
1	0	1	1	1	9	1	1	1	1
2	1	0	1	0	10	1	1	0	0
3	1	0	0	0	11	0	1	0	1
4	0	1	1	0	12	0	1	0	0
5	1	1	0	1	13	0	0	1	1
6	0	0	1	0	14	1	0	0	1

Над элементами поля $GF(2^n)$ можно выполнять операции сложения и умножения. В табл. 4 показана операция сложения элементов поля $GF(2^4)$ над примитивным полином $\varphi_1(x)$.

Таблица 4.

Таблица сложения для $GF(2^4)$ *над ПрП* $\varphi_1(x) = x^4 + x + 1$

	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	0	X	4	8	14	1	10	13	9	2	7	5	12	11	6	3
1	1	4	X	5	9	0	2	11	14	10	3	8	6	13	12	7
2	2	8	5	X	6	10	1	3	12	0	11	4	9	7	14	13
3	3	14	9	6	X	7	11	2	4	13	1	12	5	10	8	0
4	4	1	0	10	7	X	8	12	3	5	14	2	13	6	11	9
5	5	10	2	1	11	8	X	9	13	4	6	0	3	14	7	12
6	6	13	11	3	2	12	9	X	10	14	5	7	1	4	0	8
7	7	9	14	12	4	3	13	10	X	11	0	6	8	2	5	1
8	8	2	10	0	13	5	4	14	11	X	12	1	7	9	3	6
9	9	7	3	11	1	14	6	5	0	12	X	13	2	8	10	4
10	10	5	8	4	12	2	0	7	6	1	13	X	14	3	9	11
11	11	12	6	9	5	13	3	1	8	7	2	14	X	0	4	10
12	12	11	13	7	10	6	14	4	2	9	8	3	0	X	1	5
13	13	6	12	14	8	11	7	0	5	3	10	9	4	1	X	2
14	14	3	7	13	0	9	12	8	1	6	4	11	10	5	2	X

Произведение символов α^i и α^j в изоморфном отображении

$$\alpha^i \cdot \alpha^j = \alpha^{i+j} \leftrightarrow (i+j)_{15}, \qquad (5)$$

инвариантно к примитивным полиномам $\varphi(x)$, образующим поле $GF(2^4)$. Вследствие чрезвычайной простоты операции умножения (5) в пространстве изоморфного изображения, таблицу, соответствующей этой операции, которая является подобной *циклической* матрице, мы не приводим.

Сведем для удобства дальнейшего применения основные операторы над элементами поля Галуа $GF(2^n)$ в пространстве изоморфного изображения в табл. 5.

Таблица 5.

Основные операторы преобразования элементов $GF(2^n)$

№ оператора	Запись оператора в пр-ве оригиналов	Запись оператора в пр-ве изображений	Функции оператора в пр-ве изображений
1	$\alpha^a + \alpha^b$	$[a+b]$	Сложение операндов по правилам табл. 5
2	$\alpha^a \cdot \alpha^b$	$(a+b)$	Сложение операндов по модулю $2^n - 1$
	Частные варианты операций		
3	0	X	
4	$\alpha^a + \alpha^a = 0$	$[a+a] = X$	
5	$\alpha^a + 0 = \alpha^a$	$[a+X] = a$	
6	$\alpha^a \cdot 0 = 0$	$(a+X) = X$	

В тех случаях, когда это не приводит к неоднозначности трактовки выполняемой операции, квадратные скобки в операторах 1, 4 и 5 из пространства изображений можно убирать. Но круглые скобки в операторах 2 и 6 из пространства изображений убирать не допустимо.

Порождающая и проверочная матрицы РС-кодов

Компактной формой представления полного множества допустимых РС кодовых слов служит *порождающая матрица* $\boldsymbol{G}$. Порождающую $(k \times n)-$матрицу $\boldsymbol{G}$ образуют любое множество, состоящее из k линейно независимых векторов порядка n, являющихся строками матрицы $\boldsymbol{G}$. С точностью до перестановки столбцов любая порождающая матрица эквивалентна матрице, которая в первых k столбцах содержит единичную подматрицу $\boldsymbol{E}$ размером $k \times k$ [2]. Эквивалентную матрицу $\boldsymbol{G}$ можно записать в виде

$$\boldsymbol{G} = [\boldsymbol{E} \vdots \boldsymbol{P}], \qquad (6)$$

где $\boldsymbol{P}$ есть $k \times (n-k)$-матрица, каждая строка которой содержит проверочные символы, соответствующие информационному слову, расположенному в той же строке матрицы $\boldsymbol{G}$. Матрицу (6) называют *порождающей матрицей в систематическом виде*. Соотношение (6) соответствует принятой форме представления порождающей матрицы.

Наиболее естественный способ кодирования информации в любых кодах использует отображение

$$\boldsymbol{C} = \boldsymbol{I} \cdot \boldsymbol{G},$$

где $\boldsymbol{I}-$ информационное слово, представляющее собой $k-$последовательность кодируемых информационных символов, а $\boldsymbol{C}-$ образующая кодовое слово $n-$последовательность.

В дополнении к порождающей матрице $\boldsymbol{G}$ введем *проверочную матрицу* $\boldsymbol{H}$, содержащую n строк и $n-k$ столбцов, и так называемый *синдром* $\boldsymbol{S}$ кодового слова $\boldsymbol{C}$, вычисляемый по формуле

$$\boldsymbol{S} = \boldsymbol{C} \cdot \boldsymbol{H}.$$

Термин «синдром» является медицинским термином и означает группу признаков (симптомов), характеризуя аномальное состояние организма. В теории кодирования данный термин используется в качестве признака наличия (или отсутствия) в кодовом слове искаженных символов. Естественно придать синдрому такие значения. Если в кодовом слове $\boldsymbol{C}$ нет искаженных символов, то его синдром должен быть равен нулю. В том случае, когда $\boldsymbol{C}$ содержит один или более искаженных символов, то синдром этого слова должен быть отличным от нуля. Таким образом, $n-$последовательность $\boldsymbol{C}$ является допустимым

кодовым словом в том и только в том случае, когда синдром, отвечающий $\boldsymbol{C}$, равен нулю, т.е. когда

$$\boldsymbol{C}\cdot\boldsymbol{H}=\boldsymbol{0}. \qquad (7)$$

Поскольку равенство (7) должно выполняться при подстановке вместо $\boldsymbol{C}$ любой строки матрицы $\boldsymbol{G}$, то, согласно [2-5]

$$\boldsymbol{G}\cdot\boldsymbol{H}=\boldsymbol{0}. \qquad (8)$$

Из соотношений (6) и (8) совершенно формально приходим к выражению

$$\boldsymbol{H}=\begin{bmatrix}\boldsymbol{P}\\ \cdots\\ \boldsymbol{E}\end{bmatrix}, \qquad (9)$$

поскольку

$$\boldsymbol{G}\cdot\boldsymbol{H}=[\boldsymbol{E}\vdots\boldsymbol{P}]\cdot\begin{bmatrix}\boldsymbol{P}\\ \cdots\\ \boldsymbol{E}\end{bmatrix}=\boldsymbol{P}+\boldsymbol{P}=\boldsymbol{0}.$$

на том основании, что суммирование элементов матриц $\boldsymbol{P}$, являющихся $m-$битными символами, осуществляется по правилам двоичной модулярной арифметики.

Переходим к конкретным числовым примерам. Для вычисления проверочных символов, являющихся компонентами матрицы $\boldsymbol{P}$, предварительно следует определить генерирующий (образующий) полином $g(x)$ РС-кодов по формуле

$$g(x)=\prod_{i=1}^{2t}(x-\alpha^i), \qquad (10)$$

где $t-$максимально допустимая кратность ошибок в кодовом слове, принятая во введении равной 3. Воспользовавшись отображением (1) и операторами из табл. 5, а также данными табл. 4 и тем, что фактически мы «работаем» в двоичном модулярном пространстве, в котором знак «минус» можно заменить на «плюс», получим

$$\begin{aligned}
g(x)&=\{x+1\}\cdot\{x+2\}\cdot\{x+3\}\cdot\{x+4\}\cdot\{x+5\}\cdot\{x+6\}=\\
&=\left\{x^2+[1+2]\cdot x+(1+2)\right\}\cdot\left\{x^2+[3+4]\cdot x+(3+4)\right\}\cdot\left\{x^2+[5+6]\cdot x+(5+6)\right\}=\\
&=\left\{x^2+5\cdot x+3\right\}\cdot\left\{x^2+7\cdot x+7\right\}\cdot\left\{x^2+9\cdot x+11\right\}=\\
&=\left\{x^4+[5+7]\cdot x^3+\left[7+3+(5+7)\right]\cdot x^2+[(5+7)+(3+7)]\cdot x+(3+7)\right\}\cdot\left\{x^2+9\cdot x+11\right\}=\\
&=\left\{x^4+13\cdot x^3+\left[7+3+12\right]\cdot x^2+[12+10]\cdot x+10\right\}\cdot\left\{x^2+9\cdot x+11\right\}=\\
&=\left\{x^4+13\cdot x^3+6\cdot x^2+3\cdot x+10\right\}\cdot\left\{x^2+9\cdot x+11\right\}=\\
&=\left\{x^6+\left[9+13\right]\cdot x^5+\left[11+\left(13+9\right)+6\right]\cdot x^4+\left[\left(13+11\right)+(6+9)+3\right]\cdot x^3\right\}+\\
&+\left\{\left[\left(6+11\right)+\left(3+9\right)+10\right]\cdot x^2+\left[\left(3+11\right)+\left(10+9\right)\right]\cdot x+\left(10+11\right)\right\}=\\
&=\left\{x^6+10\cdot x^5+\left[11+7+6\right]\cdot x^4+\left[9+0+3\right]\cdot x^3+\left[2+12+10\right]\cdot x^2+\left[14+4\right]\cdot x+10\right\}=\\
&=\left\{x^6+10\cdot x^5+14\cdot x^4+4\cdot x^3+6\cdot x^2+9\cdot x+6\right\}. \qquad (11)
\end{aligned}$$

В соотношениях (11) задействованы фигурные скобки, поскольку круглыми и квадратными скобками обозначены операторы преобразований (табл. 5) в пространстве изоморфного отображения.

Генераторную функцию (10) для принятых параметров преобразования можно следующим образом записать коэффициентами разложения $\{\alpha^k\}$ в пространстве изображений

$$\left\{\alpha^k\right\} \leftrightarrow \{0, 10, 14, 4, 6, 9, 6\}, \qquad (12)$$

здесь левый символ $0 = \alpha^0$ в пространстве изображений отвечает элементу 1 в пространстве оригиналов.

Алгоритм синтеза порождающей матрицы $\boldsymbol{G}$ кодов Рида-Соломона в пространстве изображений подобен алгоритму синтеза порождающих матриц циклических кодов. Разместим коэффициенты генераторной функции (12) в нижней строке матрицы $\boldsymbol{G}$, приписав ей (строке) номер 1, а в ее второй строке запишем те же коэффициенты, предварительно сдвинув их на один разряд влево

							0	10	14	4	6	9	6	
								0	10	14	4	6	9	6

Для того чтобы привести левую часть матрицы $\boldsymbol{G}$ к виду единичной матрицы, следует избавиться от стоящего справа от 0 во второй строке элемента 10. С этой целью сначала следует умножить нижнюю строку $\boldsymbol{G}$ на α^{10}, что эквивалентно сложению непустых элементов нижней строки с цифрой 10 и вычислению от суммы остатка по модулю 15. Имеем

							0	10	14	4	6	9	6	
								10	5	9	14	1	4	1

После этого необходимо найти поразрядную сумму элементов этих двух строк, т.е. вычислить $\alpha^i \oplus \alpha^j$, где i и j есть числа второй и первой строк $\boldsymbol{G}$, воспользовавшись таблицей сложения 5, и разместить значение показателя при α в соответствующем разряде второй строки. Восстановив исходное состояние нижней строки, получим

(2)								0		12	14	8	3	12	1
(1)									0	10	14	4	6	9	6

В левом дополнительном столбце предыдущей таблицы в скобках указаны номера строк матрицы $\boldsymbol{G}$. Продолжив обозначенную схему вычисления, приходим к окончательной форме порождающей матрицы

$$\boldsymbol{G} =$$

0									9	4	8	13	0	3
	0								12	0	13	10	8	13
		0							7	7	13	4	9	10
			0						4	1	4	3	2	10
				0					4	9	9	5	12	14
					0				8	7	0	8	12	7
						0			1	7	9	10	11	3
							0		12	14	8	3	12	1
								0	10	14	4	6	9	6

Пусть

$$\boldsymbol{V} = 5, 12, 0, 7, 10, 4, 2, 11, 3,$$

есть 9-символьное информационное слово, представленное в изоморфной форме. Кодовое слово $\boldsymbol{U}$, отвечающее информационному слову $\boldsymbol{V}$, определяется соотношением

$$\boldsymbol{U} = \boldsymbol{V} \, \square \, \boldsymbol{p}, \qquad (13)$$

где $\boldsymbol{p}$ – вектор проверочных символов, который можно вычислить матричным произведением, $\square$ – знак конкатенации

$$\boldsymbol{p} = \boldsymbol{V} \cdot \boldsymbol{P}. \qquad (14)$$

Матрица $\boldsymbol{P}$ представляет собою прямоугольную (9,6)-матрицу, содержащую элементы проверочной матрицы $\boldsymbol{G}$, т.е.

$$\boldsymbol{P} =$$

9	4	8	13	0	3
12	0	13	10	8	13

7	7	13	4	9	10
4	1	4	3	2	10
4	9	9	5	12	14
8	7	0	8	12	7
1	7	9	10	11	3
12	14	8	3	12	1
10	14	4	6	9	6

Вычисление произведения (14) выполним с помощью ниже следующей таблицы

V		***P***					
5		9	4	8	13	0	3
12		12	0	13	10	8	13
0		7	7	13	4	9	10
7		4	1	4	3	2	10
10		4	9	9	5	12	14
4		8	7	0	8	12	7
2		1	7	9	10	11	3
11		12	14	8	3	12	1
3		10	14	4	6	9	6

Произведение элементов вектора $\boldsymbol{V}$ на элементы матрицы $\boldsymbol{P}$ сводится к нескольким этапам. Сначала следует обычной арифметической операцией сложить числа вектора $\boldsymbol{V}$ с числами этой же строки матрицы $\boldsymbol{P}$. Имеем

V		***P+***					
5		14	9	13	18	5	8
12		24	12	25	22	20	25
0		7	7	13	4	9	10
7		11	8	11	10	9	17
10		14	19	19	15	22	24
4		12	11	4	12	16	11
2		3	9	11	12	13	5
11		23	25	19	14	23	12
3		13	17	7	9	12	9

В матрице $\boldsymbol{P+}$ необходимо привести элементы к остатку по модулю 15, а затем избавиться от одинаковых символов, поскольку их суммы порождают пустые элементы. В результате выполнения рекомендуемых операций получим

V		***P+***					
		(6)	(5)	(4)	(3)	(2)	(1)
5					3		8
12		9	12	10	7		
0		7	7		4		
7		11	8		10		2
10			4		0	7	
4		12	11			1	11
2		3				13	5
11		8	10	4	14	8	12
3		13	2	7	9	12	

Проведем суммирование (в пространстве изображений) элементов столбцов матрицы $\boldsymbol{P}+$. Получим

(6) = [9+7+11+12+3+8+13] = [0+0+13+13] = X.
(5) = [12+7+8+4+11+10+2] = [2+5+14+2] = [5+14]= 12.
(4) = [10+4+7] = [2+7] = 12.
(3) = [3+7+4+10+0+14+9] = [4+2+3+9] = [10+1] = 8.
(2) = [7+1+13+8+12] = [14+3+12] = [0+12] = 11.
(1) = [8+2+11+5+12] = [0+3+12] = [14+12] = 5.

Следовательно,

$$\boldsymbol{p} = X,12,12,8,11,5. \qquad (15)$$

А теперь убедимся в том (табл. 6), что к тому же значению $\boldsymbol{p}$ можно прийти в результате деления полинома $x^6 \cdot V(x)$ на образующий полином $g(x)$.

Таблица 6.

К вычислению проверочных символов

$V(x)$																$g(x)$						
5	12	0	7	10	4	2	11	3	*X*	*X*	*X*	*X*	*X*	*X*	0	10	14	4	6	9	6
5	0	4	9	11	14	11			*Частное* ⇒						5	11	11	9	6	3	11
	11	1	0	14	9	9	11								1	14					
	11	6	10	0	2	5	2														
		11	5	3	11	6	9	3													
		11	6	10	0	2	5	2													
			9	12	12	3	6	6	*X*												
			9	4	8	13	0	3	0												
				6	9	8	13	2	0	*X*											
				6	1	5	10	12	0	12											
					3	4	9	7	X	12	*X*										
					3	13	2	7	9	12	9										
						11	11	X	9	X	9	*X*									
						11	6	10	0	2	5	2									
							1	10	7	2	6	2	*X*								
							1	11	0	5	7	10	7								
								14	9	1	10	4	7	*X*							
								14	9	13	3	5	8	5							
							$\boldsymbol{p}$	=	*X*	12	12	8	11	5		⇐*Остаток*					

Выделенный в нижней строке табл. 6 вектор совпадает с вектором (15), что является свидетельством правильно проведенных вычислений. Итак, приходим к следующему значению 15-символьного кодового слова, допускающего исправление до трех ошибочных символов.

$$\boldsymbol{U} = 5,12,0,7,10,4,2,11,3,X,12,12,8,11,5 \qquad (16)$$

Любой корень $\alpha^i, i = \overline{1,15}$, допустимого кодового слова (16) обращает это слово в нуль, т.е.

$$U(x) = \sum_{j=0}^{14} a_j x^j \,|_{x=\alpha^i} = 0. \qquad (17)$$

Проверим соблюдение равенства (17) для минимальной степени i корня α, полагая $\alpha^1 = \alpha$. Заменив в (17) аргумент x на α, получим

$$U(\alpha) = [19+25+12+18+20+13+10+18+9+X+16+0+10+12+5] =$$
$$= [4+13+10+9+1+0] = [11+13+4] = X,$$

т.е. равенство (17) соблюдается. В соответствии с (9) составим проверочную матрицу

$\boldsymbol{H}$ =

9	4	8	13	0	3
12	0	13	10	8	13
7	7	13	4	9	10
4	1	4	3	2	10
4	9	9	5	12	14
8	7	0	8	12	7
1	7	9	10	11	3
12	14	8	3	12	1
10	14	4	6	9	6
0					
	0				
		0			
			0		
				0	
					0

Вычислим синдром $\boldsymbol{S}$ кодового слова (16). С этой целью просуммируем элементы слова $\boldsymbol{U}$ с элементами столбцов матрицы $\boldsymbol{H}$ по модулю 15, уберем пары одинаковых элементов в столбцах матрицы $\boldsymbol{H}$ и вычеркнем строку таблицы для элемента $\boldsymbol{U} = X$, так как в пространстве оригиналов множитель X равен 0. В результате перечисленных преобразований приходим к таблице

$\boldsymbol{U}$		$\boldsymbol{H}$					
		(6)	(5)	(4)	(3)	(2)	(1)
5			9		3		8
12		9		10	7		
0		7	7		4		
7		11	8		10		2
10			4		0	7	
4		12	11			1	11
2		3	9			13	
11		8	10	4	14	8	12
3		13	2	7	9	12	
X							
12							
12				12			
8					8		
11						11	
5							

Просуммируем с помощью табл. 5 оставшиеся элементы столбцов матрицы $\boldsymbol{H}$

(6) = [9+7+11+12+3+8+13] = [0+0+13+13] = X .

(5) = [9+7+8+4+11+9+10+2] = [0+5+2+10+2] = [0+5+10] = [10+10] = X .

(4) = [10+4+7+12] = [2+2] = X .

(3) = [3+7+4+10+0+14+9+8] = [4+4+5+4+8] = [5+4+8] = [8+8] = X .

(2) = [7+1+13+8+12+11] = [14+3+0] = [0+0] = X .

(1) = [8+2+11+12] = [0+0] = X .

Убеждаемся в том, что допустимому кодовому слову соответствует нулевой синдром. Легко проверить, что циклический сдвиг кодового слова также сохраняет нулевым его синдром. Именно по этой причине подобные коды и называются *циклическими.*

Обнаружение-исправление искаженных символов

На основании векторной формы (16) выпишем полином неискаженного кодового слова

$$U(x)=\alpha^5x^{14}+\alpha^{12}x^{13}+\alpha^0x^{12}+\alpha^7x^{11}+\alpha^{10}x^{10}+\alpha^4x^9+\alpha^2x^8+$$
$$+\alpha^{11}x^7+\alpha^3x^6+0x^5+\alpha^{12}x^4+\alpha^{12}x^3+\alpha^8x^2+\alpha^{11}x^1+\alpha^5x^0$$

и пусть

$$e(x)=\alpha^1x^{11}+\alpha^7x^8+\alpha^3x^1, \qquad (18)$$

есть полином вектора ошибок.

Составим полином искаженного кодового слова

$$U^*(x)=U(x)+e(x)=$$
$$=\alpha^5x^{14}+\alpha^{12}x^{13}+\alpha^0x^{12}+[\alpha^7+\alpha^1]\cdot x^{11}+\alpha^{10}x^{10}+\alpha^4x^9+[\alpha^2+\alpha^7]\cdot x^8$$
$$+\alpha^{11}x^7+\alpha^3x^6+0x^5+\alpha^{12}x^4+\alpha^{12}x^3+\alpha^8x^2+[\alpha^{11}+\alpha^3]\cdot x^1+\alpha^5x^0.$$

В соответствии с табл. 5 получим

$$U^*(x)=\alpha^5x^{14}+\alpha^{12}x^{13}+\alpha^0x^{12}+\alpha^{14}x^{11}+\alpha^{10}x^{10}+\alpha^4x^9+\alpha^{12}x^8$$
$$+\alpha^{11}x^7+\alpha^3x^6+0x^5+\alpha^{12}x^4+\alpha^{12}x^3+\alpha^8x^2+\alpha^5x^1+\alpha^5x^0. \qquad (19)$$

Последующая задача вычислений состоит в том, чтобы на основании искаженного кодового слова (19) локализовать и исправить пораженные помехой (18) символы кода. Решение данной задачи разбивается на два этапа.

Этап 1. Локализация (обнаружение) ошибок.

Предварительно необходимо вычислить (в пространстве изоморфного изображения) синдромы $S_i=U^*(\alpha^i)$, число которых совпадает с числом проверочных символов, равных шести, т.е. $i=\overline{1,6}$. С этой целью следует записать (в круглых скобках) сумму показателей степеней в мономах с последующим приведением их по модулю 15. Воспользовавшись обозначением операторов, приведенных в табл. 5, получим

$$S_1=U^*(\alpha^1)=\ [(5+14)+(12+13)+(0+12)+(14+11)+(10+10)+(4+9)+$$
$$+(12+8)+(11+7)+(3+6)+X+(12+4)+(12+3)+(8+2)+(5+1)+(5+0)]=$$
$$=[4+10+12+10+5+13+5+3+9+1+0+10+6+5].$$

Удалив пары одинаковых символов (цифр), получим

$$S_1=\ [4+12+13+3+9+1+0+10+6+5]=[6+8+3+5+9]=[14+11+9]=[10+9]=13.$$

Аналогичным способом вычисляем оставшиеся синдромы

$$S_2=U^*(\alpha^2)=\ [(5+28)+(12+26)+(0+24)+(14+22)+(10+20)+(4+18)+$$
$$+(12+16)+(11+14)+(3+12)+X+(12+8)+(12+6)+(8+4)+(5+2)+(5+0)]=$$
$$=[3+8+9+6+0+7+13+10+0+5+3+12+7+5]=[8+9+6+13+10+12]=$$
$$=[12+0+10+12]=[0+10]=5.$$

$$S_3=U^*(\alpha^3)=\ [(5+42)+(12+39)+(0+36)+(14+33)+(10+30)+(4+27)+$$
$$+(12+24)+(11+21)+(3+18)+X+(12+12)+(12+9)+(8+6)+(5+3)+(5+0)]=$$
$$=[2+6+6+2+10+1+6+2+6+9+6+14+8+5]=[10+1+2+9+6+14+8+5]=$$
$$=[8+11+8+4]=[11+4]=13.$$

$$S_4=U^*(\alpha^4)=\ [(5+56)+(12+52)+(0+48)+(14+44)+(10+40)+(4+36)+$$
$$+(12+32)+(11+28)+(3+24)+X+(12+16)+(12+12)+(8+8)+(5+4)+(5+0)]=$$
$$=[1+4+3+13+5+10+14+9+12+13+9+1+9+5]=[4+3+10+14+9+12]=$$
$$=[7+11+8]=[8+8]=X.$$

$$S_5=U^*(\alpha^5)=\ [(5+70)+(12+65)+(0+60)+(14+55)+(10+50)+(4+45)+$$
$$+(12+40)+(11+35)+(3+30)+X+(12+20)+(12+15)+(8+10)+(5+5)+(5+0)]=$$
$$=[0+2+0+9+0+4+7+1+3+2+12+3+10+5]=[9+0+4+7+1+12+10+5]=$$
$$=[7+7+0+3+5]=[0+3+5]=[14+5]=12.$$

$$S_6 = U^*(\alpha^6) = [(5+84)+(12+78)+(0+72)+(14+66)+(10+60)+(4+54)+$$
$$+(12+48)+(11+42)+(3+36)+ X +(12+24)+(12+18)+(8+12)+(5+6)+(5+0)] =$$
$$= [14+0+12+5+10+13+0+8+9+6+0+5+11+5] = [14+0+12+5+10+13+8+9+6+11] =$$
$$= [3+14+9+14+9+11] = [3+11] = 5.$$

Покажем, что к таким же значениям синдрома приходим по формуле

$$S_i = e(\alpha^i),\ i = \overline{1, 6}.$$

Согласно (18) имеем

$S_1 = e(\alpha^1) = [(1+11)+(7+8)+(3+1)] = [12+0+4] = [11+4] =13.$

$S_2 = e(\alpha^2) = [(1+22)+(7+16)+(3+2)] = [8+8+5] = 5.$

$S_3 = e(\alpha^3) = [(1+33)+(7+24)+(3+3)] = [4+1+6] = [0+6] = 13.$

$S_4 = e(\alpha^4) = [(1+44)+(7+32)+(3+4)] = [0+9+7] = [7+7] = X.$

$S_5 = e(\alpha^5) = [(1+55)+(7+40)+(3+5)] = [11+2+8] = [9+8] = 12.$

$S_6 = e(\alpha^6) = [(1+66)+(7+48)+(3+6)] = [7+10+9] = [6+9] = 5.$

Значения синдромов, рассчитанные двумя способами, совпали. А это означает, что синдромы вычислены правильно. Составим далее из синдромов матрицу $\boldsymbol{M}_s$ локатора ошибок. Порядок матрицы совпадает с числом ошибок t, устраняемых РС-кодом ($t = 3$)

$$\boldsymbol{M}_s = \begin{bmatrix} S_1 & S_2 & S_3 \\ S_2 & S_3 & S_4 \\ S_3 & S_4 & S_5 \end{bmatrix}.$$

Имеем

$$\boldsymbol{M}_s = \begin{bmatrix} 13 & 5 & 13 \\ 5 & 13 & X \\ 13 & X & 12 \end{bmatrix}. \qquad (20)$$

Запишем полином локатора ошибок

$$\sigma(x) = \alpha^0 + \sigma_1 x^1 + \sigma_2 x^2 + \sigma_3 x^3. \qquad (21)$$

Коэффициенты σ_i, $i = \overline{1, t}$, находим, решая матричное уравнение

$$\boldsymbol{M}_s \cdot \begin{bmatrix} \sigma_3 \\ \sigma_2 \\ \sigma_1 \end{bmatrix} = \begin{bmatrix} S_4 \\ S_5 \\ S_6 \end{bmatrix}. \qquad (22)$$

С этой целью:

Вычислим матрицу $\overline{\boldsymbol{M}}_s$, обратную матрице $\boldsymbol{M}_s$ по формуле

$$\overline{\boldsymbol{M}}_s = \frac{\boldsymbol{M}_s}{\det[\boldsymbol{M}_s]}, \qquad (23)$$

где $\boldsymbol{M}_s -$ *присоединенная* к $\boldsymbol{M}_s$ матрица, а $\det[\boldsymbol{M}_s] -$ определитель матрицы. Матрицу $\boldsymbol{M}_s$ называют также *взаимной* или *союзной* матрице $\boldsymbol{M}_s$.

Найдем сначала определитель матрицы (20). Принимая во внимание, что элементу X этой матрицы в пространстве оригиналов соответствует нуль, получим

$$\det[\boldsymbol{M}_s] = [(13+13+12)+(13+13+13)+(5+5+12)] = [8+9+7] = [12+7] = 2. \qquad (24)$$

$(i, j)-$й элемент присоединенной матрицы $\boldsymbol{M}_s$, где $i-$номер строки, $j-$номер столбца, равен определителю матрицы $\boldsymbol{M}_s$, в которой удалены $j-$я строка и $i-$й столбец. Следовательно,

$$\boldsymbol{M}_s = \begin{bmatrix} (13+12) & (5+12) & (13+13) \\ (5+12) & [(13+12)+(13+13)] & (5+13) \\ (13+13) & (5+13) & [(13+13)+(5+5)] \end{bmatrix} =$$

$$= \begin{bmatrix} 10 & 2 & 11 \\ 2 & [10+11] & 3 \\ 11 & 3 & [11+10] \end{bmatrix} = \begin{bmatrix} 10 & 2 & 11 \\ 2 & 14 & 3 \\ 11 & 3 & 14 \end{bmatrix},$$

т.е.

$$\boldsymbol{M}_s = \begin{bmatrix} 10 & 2 & 11 \\ 2 & 14 & 3 \\ 11 & 3 & 14 \end{bmatrix}. \qquad (25)$$

Согласно соотношениям (23)-(25), получим

$$\overline{\boldsymbol{M}}_s = \frac{\begin{bmatrix} 10 & 2 & 11 \\ 2 & 6 & 3 \\ 11 & 3 & 6 \end{bmatrix}}{2} = (-2)\cdot\begin{bmatrix} 10 & 2 & 11 \\ 2 & 14 & 3 \\ 11 & 3 & 4 \end{bmatrix} = 13\cdot\begin{bmatrix} 10 & 2 & 11 \\ 2 & 14 & 3 \\ 11 & 3 & 14 \end{bmatrix} =$$

$$= \begin{bmatrix} (10+13) & (2+13) & (11+13) \\ (2+13) & (14+13) & (3+13) \\ (11+13) & (3+13) & (14+13) \end{bmatrix} = \begin{bmatrix} 8 & 0 & 9 \\ 0 & 12 & 1 \\ 9 & 1 & 12 \end{bmatrix}.$$

Таким образом,

$$\overline{\boldsymbol{M}}_s = \begin{bmatrix} 8 & 0 & 9 \\ 0 & 12 & 1 \\ 9 & 1 & 12 \end{bmatrix}. \qquad (26)$$

Проверим тождество $\boldsymbol{M}_s \cdot \overline{\boldsymbol{M}}_s \equiv \boldsymbol{E}$, используя матрицы (20) и (26). Имеем

$$\boldsymbol{M}_s \cdot \overline{\boldsymbol{M}}_s = \begin{bmatrix} 13 & 5 & 13 \\ 5 & 13 & X \\ 13 & X & 12 \end{bmatrix}\cdot\begin{bmatrix} 8 & 0 & 9 \\ 0 & 12 & 1 \\ 9 & 1 & 12 \end{bmatrix} =$$

$$= \begin{bmatrix} [(13+8)+(5+0)+(13+9)] & [(13+0)+(5+12)+(13+1)] & [(13+9)+(5+1)+(13+12)] \\ [(5+8)+(13+0)] & [(5+0)+(13+12)] & [(5+9)+(13+1)] \\ [(13+8)+(12+9)] & [(13+0)+(12+1)] & [(13+9)+(12+12)] \end{bmatrix} =$$

$$= \begin{bmatrix} [6+5+7] & [13+2+14] & [7+6+10] \\ [13+13] & [5+10] & [14+14] \\ [6+6] & [13+13] & [7+9] \end{bmatrix} = \begin{bmatrix} [9+7] & [14+14] & [10+10] \\ X & 0 & X \\ X & X & 0 \end{bmatrix} = \begin{bmatrix} 0 & X & X \\ X & 0 & X \\ X & X & 0 \end{bmatrix}$$

Если полученную матрицу перевести из пространства изображений в пространство оригиналов, то приходим к единичной матрице. А это свидетельствует о том, что обратная матрица (28) вычислена правильно.

Умножив обе части матричного уравнения (22) слева на $\overline{\boldsymbol{M}}_s$, получим

$$\begin{bmatrix}\sigma_3\\ \sigma_2\\ \sigma_1\end{bmatrix} = \overline{\boldsymbol{M}}_s \cdot \begin{bmatrix}S_4\\ S_5\\ S_6\end{bmatrix} = \begin{bmatrix}8 & 0 & 9\\ 0 & 12 & 1\\ 9 & 1 & 12\end{bmatrix} \cdot \begin{bmatrix}X\\ 12\\ 5\end{bmatrix}$$

Следовательно

$$\sigma_3 = [(0+12)+(9+5)] = [12+14] = 5.$$
$$\sigma_2 = [(12+12)+(1+5)] = [9+6] = 5.$$
$$\sigma_1 = [(1+12)+(12+5)] = [13+2] = 14.$$

Подставив коэффициенты σ_i, $i = \overline{1,3}$, в (21), приходим к полиному локатора ошибок

$$\sigma(x) = \alpha^0 + \alpha^{14}x^1 + \alpha^5 x^2 + \alpha^5 x^3. \qquad (27)$$

Любой элемент x, который дает $\sigma(x) = 0$, является корнем полинома локатора ошибок. Это позволяет определить расположение ошибки $\beta_i = x_i^{-1}$, $i = \overline{1,t}$, где $x_i -$ то значение аргумента x, которое обеспечивает $\sigma(x) = 0$. Вычислим, используя изоморфное представление локатора, все корни $\sigma(\alpha^i)$, $i = \overline{0,15}$, полинома локатора ошибок (27).

$\sigma(\alpha^0) =$ [0+14+5+5] = [0+14] = 3.

$\sigma(\alpha^1) =$ [0+(14+1)+(5+2)+(5+3)] = [0+0+7+8] = [7+8] = 11.

$\sigma(\alpha^2) =$ [0+(14+2)+(5+4)+(5+6)] = [0+1+9+11] = [4+2] = 10.

$\sigma(\alpha^3) =$ [0+(14+3)+(5+6)+(5+9)] = [0+2+11+14] = [8+10] = 1.

$\sigma(\alpha^4) =$ [0+(14+4)+(5+8)+(5+12)] = [0+3+13+2] = [14+14] = X – корень.

$\sigma(\alpha^5) =$ [0+(14+5)+(5+10)+(5+15)] = [0+4+0+5] = [4+5] = 8.

$\sigma(\alpha^6) =$ [0+(14+6)+(5+12)+(5+18)] = [0+5+2+8] = [10+0] = 5.

$\sigma(\alpha^7) =$ [0+(14+7)+(5+14)+(5+21)] = [0+6+4+11] = [13+13]= X – корень.

$\sigma(\alpha^8) =$ [0+(14+8)+(5+16)+(5+24)] = [0+6+6+14] = [0+14] = 3.

$\sigma(\alpha^9) =$ [0+(14+9)+(5+18)+(5+27)] = [0+8+8+2] = [0+2] = 8.

$\sigma(\alpha^{10}) =$ [0+(14+10)+(5+20)+(5+30)] = [0+9+10+5] = [7+0] = 9.

$\sigma(\alpha^{11}) =$ [0+(14+11)+(5+22)+(5+33)] = [0+10+12+8] = [5+9] = 6.

$\sigma(\alpha^{12}) =$ [0+(14+12)+(5+24)+(5+36)] = [0+11+4+11] = [0+4] = 1.

$\sigma(\alpha^{13}) =$ [0+(14+13)+(5+26)+(5+39)] = [0+12+1+14] = [11+7] = 8.

$\sigma(\alpha^{14}) =$ [0+(14+14)+(5+28)+(5+42)] = [0+13+3+2] = [6+6] = X – корень.

В соответствии с вычисленными корнями находим расположение ошибок

$$\beta_1 = (\alpha^4)^{-1} = \alpha^{-4} = \alpha^{11} = 11.$$
$$\beta_2 = (\alpha^7)^{-1} = \alpha^{-7} = \alpha^8 = 8. \qquad (28)$$
$$\beta_3 = (\alpha^{14})^{-1} = \alpha^{-14} = \alpha^1 = 1.$$

Согласно системе (28) ошибки расположены в полиноме кодового слова со степенями аргумента x, равными 11, 8 и 1, что соответствует выбранному полиному вектора ошибок (18), т.е. положение ошибочных символов локализовано правильно.

Этап 2. Устранение ошибок.

Обозначим e_i, $i = \overline{1,t}$, значение ошибок, которые определяются на основании решения матричного уравнения

$$\boldsymbol{M}_\beta \cdot \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix}, \qquad (29)$$

где

$$\boldsymbol{M}_\beta = \begin{bmatrix} \beta_1 & \beta_2 & \beta_3 \\ \beta_1^2 & \beta_2^2 & \beta_3^2 \\ \beta_1^3 & \beta_2^3 & \beta_3^3 \end{bmatrix}. \qquad (30)$$

Перенесем указатели местоположения ошибок β_i, $i = \overline{1,3}$ из системы (28) в матрицу (30), принимая во внимание, что в пространстве изоморфного изображения возведению элемента β_i в некоторую степень соответствует умножение этого элемента на данную степень. Имеем

$$\boldsymbol{M}_\beta = \begin{bmatrix} 11 & 8 & 1 \\ 22 & 16 & 2 \\ 33 & 24 & 3 \end{bmatrix}. \qquad (31)$$

Элементы матрицы (31) следует привести к остатку по модулю 15.

$$\boldsymbol{M}_\beta = \begin{bmatrix} 11 & 8 & 1 \\ 7 & 1 & 2 \\ 3 & 9 & 3 \end{bmatrix}. \qquad (32)$$

Решение уравнения (29) таково

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \overline{\boldsymbol{M}}_\beta \cdot \begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix} = \overline{\boldsymbol{M}}_\beta \cdot \begin{bmatrix} 13 \\ 5 \\ 13 \end{bmatrix}, \qquad (33)$$

где $\overline{\boldsymbol{M}}_\beta -$ обратная матрица, для вычисления которой следует предварительно найти определитель матрицы (32).

$$\det\left[\boldsymbol{M}_\beta\right] = \det = [(11+1+3)+(8+2+3)+(7+9+1)+(3+1+1)+(9+2+11)+(7+8+3)] =$$
$$= [0+13+2+5+7+3] = [6+1+4] = [11+4] = 13.$$

Затем вычислим матрицу $\boldsymbol{M}_\beta$, присоединенную к матрице $\boldsymbol{M}_\beta$.

$$\boldsymbol{M}_\beta = \begin{bmatrix} [(1+3)+(9+2)] & [(8+3)+(9+1)] & [(8+2)+(1+1)] \\ [(7+3)+(3+2)] & [(11+3)+(3+1)] & [(11+2)+(7+1)] \\ [(7+9)+(3+1)] & [(11+9)+(3+8)] & [(11+1)+(7+8)] \end{bmatrix} =$$
$$= \begin{bmatrix} [4+11] & [11+10] & [10+2] \\ [10+5] & [14+4] & [13+8] \\ [1+4] & [5+11] & [12+0] \end{bmatrix} = \begin{bmatrix} 13 & 14 & 4 \\ 0 & 9 & 3 \\ 0 & 3 & 11 \end{bmatrix}.$$

Обратная матрица

$$\overline{\boldsymbol{M}}_\beta = \frac{\boldsymbol{M}_\beta}{\det} = (15 - \det) \cdot \boldsymbol{M}_\beta = 2 \cdot \boldsymbol{M}_\beta.$$

Имеем

$$\overline{\boldsymbol{M}}_\beta = 2 \cdot \begin{bmatrix} 13 & 14 & 4 \\ 0 & 9 & 3 \\ 0 & 3 & 11 \end{bmatrix} = \begin{bmatrix} (13+2) & (14+2) & (4+2) \\ (0+2) & (9+2) & (3+2) \\ (0+2) & (3+2) & (11+2) \end{bmatrix}.$$

И в окончательном виде

$$\overline{\boldsymbol{M}}_{\beta}=\begin{bmatrix}0 & 1 & 6\\ 2 & 11 & 5\\ 2 & 5 & 13\end{bmatrix}. \qquad (34)$$

Перемножим прямую (32) и обратную (34) матрицы

$$\boldsymbol{M}_{\beta}\cdot\overline{\boldsymbol{M}}_{\beta}=\begin{bmatrix}11 & 8 & 1\\ 7 & 1 & 2\\ 3 & 9 & 3\end{bmatrix}\cdot\begin{bmatrix}0 & 1 & 6\\ 2 & 11 & 5\\ 2 & 5 & 13\end{bmatrix}=$$

$$=\begin{bmatrix}[(11+0)+(8+2)+(1+2)] & [(11+1)+(8+11)+(1+5)] & [(11+6)+(8+5)+(1+13)]\\ [(7+0)+(1+2)+(2+2)] & [(7+1)+(1+11)+(2+5)] & [(7+6)+(1+5)+(2+13)]\\ [(3+0)+(9+2)+(3+2)] & [(3+1)+(9+11)+(3+5)] & [(3+6)+(9+5)+(3+13)]\end{bmatrix}=$$

$$=\begin{bmatrix}[11+10+3] & [12+4+6] & [2+13+14]\\ [7+3+4] & [8+12+7] & [13+6+0]\\ [3+11+5] & [4+5+8] & [9+14+1]\end{bmatrix}=\begin{bmatrix}[14+3] & [6+6] & [14+14]\\ [4+4] & [9+7] & [0+0]\\ [5+5] & [8+8] & [4+1]\end{bmatrix}=\begin{bmatrix}0 & X & X\\ X & 0 & X\\ X & X & 0\end{bmatrix}.$$

Таким образом, произведение прямой и обратной матриц равно единичной матрице, что является подтверждением правильно вычисленной матрицы (34).

Подставив (34) в (33), имеем

$$\begin{bmatrix}\boldsymbol{e}_1\\ \boldsymbol{e}_2\\ \boldsymbol{e}_3\end{bmatrix}=\overline{\boldsymbol{M}}_{\beta}\cdot\begin{bmatrix}13\\ 5\\ 13\end{bmatrix}=\begin{bmatrix}0 & 1 & 6\\ 2 & 11 & 5\\ 2 & 5 & 13\end{bmatrix}\cdot\begin{bmatrix}13\\ 5\\ 13\end{bmatrix}=$$

$$=\begin{bmatrix}[(0+13)+(1+5)+(6+13)]\\ [(2+13)+(11+5)+(5+13)]\\ [(2+13)+(5+5)+(13+13)]\end{bmatrix}=\begin{bmatrix}[13+6+4]\\ [0+1+3]\\ [0+10+11]\end{bmatrix}=\begin{bmatrix}[0+4]\\ [4+3]\\ [5+11]\end{bmatrix}=\begin{bmatrix}1\\ 7\\ 3\end{bmatrix}. \qquad (35)$$

Полученные в (35) значения ошибок $e_1=1$, $e_2=7$ и $e_3=3$ совпадают с теми, которые определены заданием, что в целом завершает процедуру синтеза и анализа РС-кодов.

Выводы.

Коды Рида-Соломона являются эффективными и наиболее распространенными кодами, используемыми во многих областях науки и техники, связанной с помехоустойчивым преобразованием цифровой информации. С момента его появления (1960) и до настоящего времени описание РС-кода, включая синтез кодовых слов, а также локализация и устранение ошибок, базируется на использовании формальных элементов, которыми являются корни генераторных, информационных и проверочных полиномов. Перегруженность алгебраических структур такими корнями является в определенной мере «балластом», не только усложняющим процесс вычислений, но и создающим определенные затруднения при изучении алгоритма кодирования. В связи с этим в работе предложен вариант построения РС-кодов, значительно упрощающий как освоение самого алгоритма, так и процесс обнаружения-исправления ошибок в искаженных данных. Предложения основаны на переносе преобразований из пространства оригиналов в пространство изоморфного изображения. В результате предлагаемой замены вычислительный процесс оказывается сведенным к простым операциям модулярной арифметики над целочисленными операндами, легко реализуемыми средствами компьютерной техники.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Электронный ресурс: http://ru.wikipedia.org/wikiКод_Рида_—_Соломона
2. Блейхут Р. Теория и практика кодов, контролирующих ошибки. – М.: Мир, 1986. – 576 с.

3. Колесник В.Д. Кодирование и декодирование сообщений. Уч. пособие. С-П., 2005-2006.
4. Рахман П.А. Кодирование информации с применением кодов Рида-Соломона. Электронный ресурс: http://bugtraq.ru/library/crypto/.keep/rscodes.pdf
5. Мак-Вильямс Ф.Дж., Слоэн Н. Дж. А. Теория кодов, исправляющих ошибки.: Пер.с англ. – М.: Связь, 1979. – 744 с.

Стаття надійшла до редакції 23.08.13

А.Я. Білецький, Е.А. Білецький, О.І. Воливач, М.А. Якимчук
Національний авіаційний університет, Київ

СИНТЕЗ І АНАЛІЗ КОДІВ РІДА-СОЛОМОНА У ПРОСТОРІ ІЗОМОРФНОГО ЗОБРАЖЕННЯ

Запропоновано навчальний варіант побудови кодів Ріда-Соломона, що значно спрощує процес синтезу кодових слів і виявлення-виправлення помилок у спотворених файлах даних. Алгоритм заснований на перенесенні змінних і операцій, за допомогою яких здійснюється синтез і аналіз кодів, з простору оригіналів в простір изоморфного зображення. В результаті запропонованої заміни простору обробки даних обчислювальний процес стає зведеним до основних операцій в простих полях Галуа і модулярної арифметики над цілочисельними операндами, легко реалізованими засобами комп'ютерної техніки.

Ключові слова: коди Ріда-Соломона, поля Галуа, модулярная арифметика, ізоморфні перетворення

Beletsky A.J., Beletsky A.A., Volyvach O.I., Yakymchuk M.A.
National Aviation University, Kyiv

SYNTHESIS AND ANALYSIS OF REED-SOLOMON CODES IN ISOMORPHIC SPACE OF IMAGES

A training variant of Reed-Solomon codes, which greatly simplifies the process of synthesis of codewords and detection-fixes for corrupted data files. The algorithm is based on the transfer of variables and operations, in which the synthesis and analysis of codes from the original space into isomorphic space of image. As a result, the proposed replacement of data space computational process is reduced to the basic operations in a simple and modular Galois field arithmetic on integer operands, easily implemented by means of computer technology.

Keywords: Reed-Solomon codes, Galois fields, modular arithmetic, isomorphic transformation