УДК 519.725, 681.3

АНАЛИЗ СПОСОБОВ РЕАЛИЗАЦИИ КОДОВ РИДА-СОЛОМОНА, ИСПРАВЛЯЮЩИХ ДВОЙНЫЕ ОШИБКИ

Дяченко Валерий Олегович, Дяченко Олег Николаевич
Донецкий национальный технический университет
г. Донецк, Украина

Аннотация

Выполнен анализ способов построения и практической реализации кодов Рида-Соломона, исправляющих двойные ошибки. Рассмотрены особенности аппаратного декодирования (255, 251) кода Рида-Соломона, корректирующего двойные искаженные байты. Обоснован вывод о рациональности применения для данного типа кодов синдромного метода декодирования.

Ключевые слова: *коды Рида-Соломона, синдромное декодирование, порождающий полином, поле Галуа.*

ANALYSIS OF IMPLEMENTATION DOUBLE-ERROR-CORRECTING REED-SOLOMON CODES

Dyachenko Valery Olegovich, Dyachenko Oleg Nikolaevich
Donetsk national technical university
Donetsk, Ukraine

Abstract

The methods for constructing and implementation of the double-error-correcting Reed-Solomon codes have been analyzed. Features of the hardware decoding for the correcting distorted double bytes using (255, 251) Reed-Solomon code have been considered. The conclusion of the rationality of application the syndrome decoding for this type codes was substantiated.

Keywords*: Reed-Solomon codes, syndrome decoding, generator polynomial, Galois field.*

Введение

Современные тенденции внедрения инновационных технологий во всех областях человеческой деятельности приводят к непрерывному увеличению объема накапливаемой информации. Все большее значение приобретают способы помехоустойчивого кодирования, обеспечивающие требуемую достоверность при передаче, обработке и хранении информационных данных. Одними из наиболее эффективных для исправления ошибок и, в особенности, пакетов ошибок, являются коды Рида-Соломона.

Анализ литературы [1-4] отражает широчайший спектр разработанных и уже используемых на практике кодов Рида-Соломона. Можно привести несколько наиболее известных примеров: (255, 223, 33) код Рида-Соломона для космической связи NASA, укороченные коды Рида-Соломона над полем Галуа $GF(2^8)$ для CD-ROM, DVD и цифрового телевидения высокого разрешения (формат HDTV), расширенный (128, 122, 7) код Рида-Соломона над полем Галуа $GF(2^7)$ для кабельных модемов, (255, 239) код рекомендован в качестве внешнего кода в WiMax.

Кроме того, коды Рида-Соломона можно использовать не только для помехоустойчивого кодирования при передаче данных, а также везде, где есть необходимость в предотвращении искажения информации, например [2]: обнаружение и исправление ошибок в поврежденных или дефектных носителях информации; обнаружение и исправление ошибок при умышленном изменении информационных сообщений с целью дезинформации; обнаружение и исправление модификации информации об авторе или исполняемого кода с целью «взлома» программного обеспечения; защита программного обеспечения или данных от копирования с лицензионного диска; восстановление одного или нескольких томов многотомного архива, искаженных или вообще потерянных при загрузке из сети; обнаружение и исправление ошибок в цепочках ДНК в генной

инженерии. Поэтому вопросы построения и аппаратной реализации кодов Рида-Соломона являются актуальными, учитывая все большую их популярность и востребованность для различных сфер применения.

1. Коды Рида-Соломона – частный случай кодов БЧХ

Задача сравнительного анализа способов построения кодов Рида-Соломона, исправляющих двойные ошибки, появилась после публикации работы [4], в которой рассматриваются вопросы реализации кодера и декодера (255, 251) кода в FPGA. Вместе с тем, в работах [5-7] основное внимание уделяется другим принципам построения подобных кодов, отличающихся более простыми методами декодирования.

Прежде всего, следует отметить, что коды Рида-Соломона, исправляющие одиночные или двойные ошибки, независимо от того, по какому полю Галуа они построены, укорочены, посимвольно перемежены или нет, допускают применение метода синдромного декодирования. Такой метод неприменим для кодов исправляющих большее количество ошибок. Для них типичный декодер основан на блоках вычисления синдрома, буферного регистра, решения ключевого уравнения на основе одного из алгоритмов, (например, Берлекэмпа-Месси, алгоритма Евклида или Питерсона-Горенстейна-Цирлера), поиска корней полиномов локаторов ошибок на основе алгоритма Ченя, расчета значения ошибки (алгоритм Форни), коррекции ошибок. Такой способ построения декодера используется в [4].

Главное отличие недвоичных кодов Рида-Соломона от двоичных кодов заключается в том, что в качестве символа выступает не двоичный символ (бит), а элемент поля Галуа (несколько битов), на основе которого построен код. Порождающий полином кода Рида-Соломона, исправляющего s ошибок, должен содержать 2s корней:

$$\{\alpha^{j}_{0}, \alpha^{j}_{0}{}^{+1}, \alpha^{j}_{0}{}^{+2}, \ldots, \alpha^{j}_{0}{}^{+2s-1}\},$$

где α - примитивный элемент поля Галуа, j_0 – конструктивный параметр.

При $j_0 = 1$ множество корней приобретает вид: $\{\alpha, \alpha^2, \alpha^3 \ldots \alpha^{2s}\}$.

Для кода Рида – Соломона, исправляющего s ошибок, порождающий полином представляет собой произведение:

$RS(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^3)\ldots(x - \alpha^{2s})$.

Если поле Галуа строится над примитивным полиномом p(z), то возможна другая форма записи порождающего полинома – в качестве примитивного элемента α можно использовать элемент поля z. Такая замена впоследствии дает возможность представления умножителей на константу элементами двоичной логики без необходимости построения поля Галуа.

2. Коды Рида–Соломона, исправляющие двойную ошибку

В качестве полинома p(z) будем использовать примитивный полином. Тогда, после раскрытия скобок и замены α на z, порождающий полином кода Рида-Соломона, исправляющего двойные ошибки, можно представить в следующем виде:

$RS(X)=(X-\alpha)(X-\alpha^2)(X-\alpha^3)(X-\alpha^4)=X^4+X^3(z^4+z^3+z^2+z)+X^2(z^7+z^6+z^4+z^3)+X(z^9+z^8+z^7+z^6)+z^{10}$.

Данный порождающий полином является справедливым для любого кода Рида-Соломона, исправляющего двойные ошибки, и любого поля Галуа.

Для вычисления порождающего полинома для конкретного поля Галуа необходимо вычислить соответствующие коэффициенты при псевдопеременных X. Для этого каждый соответствующий коэффициент в общей форме необходимо разделить на примитивный полином p(z). Остаток от деления и будет представлять собой искомый коэффициент.

Используя таблицу неприводимых полиномов [3], в качестве p(z) можно выбрать первый полином (в этой таблице первый полином всегда

примитивный, причем с наименьшим количеством ненулевых коэффициентов) восьмой степени:

$p(z) = 435_8 = 100\ 011\ 101_2 = z^8+z^4+z^3+z^2+1.$

После приведения по модулю p(z) степень всех коэффициентов порождающего полинома будет не более семи:

$(z^4+z^3+z^2+z) \bmod p(z) = z^4+z^3+z^2+z;$

$(z^7+z^6+z^4+z^3) \bmod p(z) = z^7+z^6+z^4+z^3;$

$(z^9+z^8+z^7+z^6) \bmod p(z) = z^7+z^6+z^5+z^2+z+1;$

$(z^{10}) \bmod p(z) = z^6+z^5+z^4+z^2.$

Таким образом, порождающий полином кода Рида-Соломона для поля $GF(2^8)$:

$RS(X) = X^4 + X^3(z^4+z^3+z^2+z) + X^2(z^7+z^6+z^4+z^3) + X(z^7+z^6+z^5+z^2+z+1) + + (z^6+z^5+z^4+z^2).$

Декодер кода Рида – Соломона аналогичен декодеру кода БЧХ с точностью до обозначений (рис. 1). Такой декодер исправляет два возможных ошибочных байта за 3n тактов: n тактов формируется синдром, кодовое слово заносится в буферный регистр; n тактов выполняется исправление одного из двух возможных ошибочных байтов, синдром модифицируется, кодовое слово заново переписывается в буферный регистр; n тактов выполняется исправление второго ошибочного байта.

Недостаток такого декодера – несогласованность работы кодера и декодера (кодер должен ждать 2n тактов, пока декодер исправит ошибки) – устраняется применением конвейерной реализации.

Для построения декодера на элементах двоичной логики необходимо представить умножители на константу в виде сумматоров по модулю два.

Для упрощения такого преобразования достаточно выполнить только одну операцию умножения и деления и получить в данном случае восемь результатов. Умножение элемента поля в общем виде $a_7z^7 + a_6z^6 + a_5z^5 + a_4z^4 + a_3z^3 + a_2z^2 + a_1z + a_0$ на полином z^7 и деление на p(z) позволяет получить все

остатки от деления R_7, …, R_0 соответствующих произведений элемента поля в общем виде и полиномов z^7 … z^0.

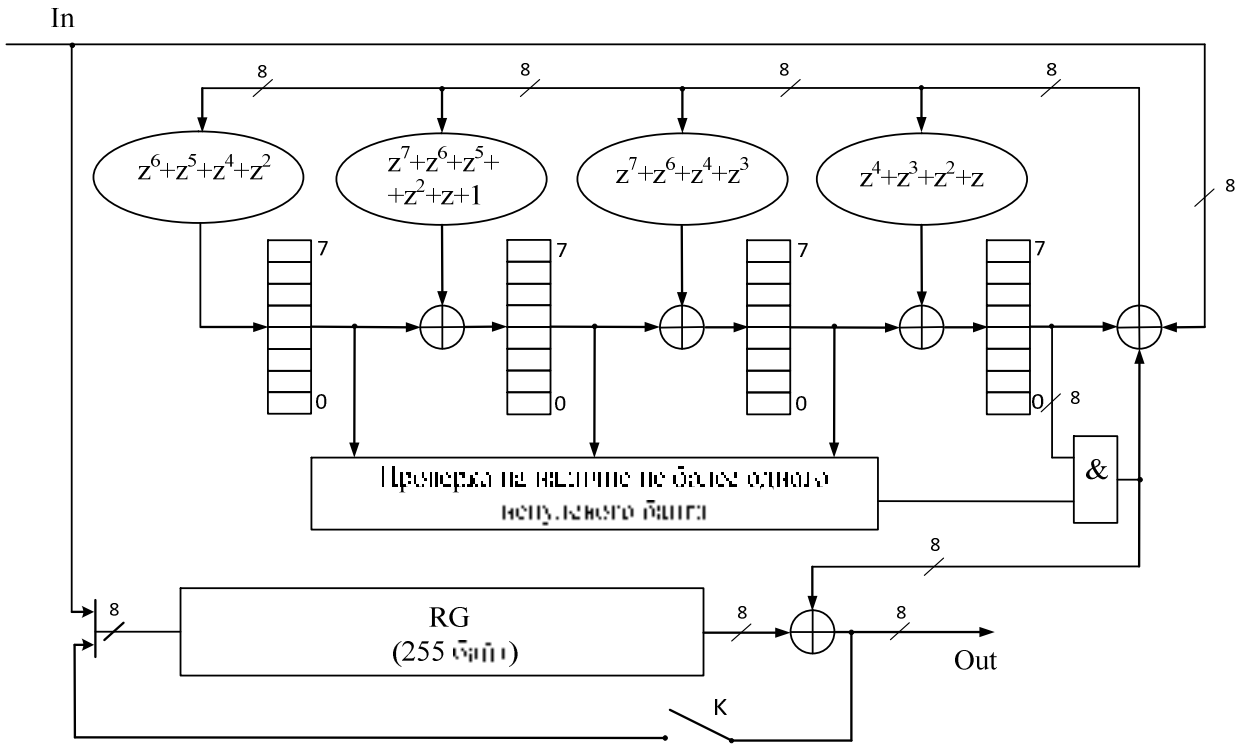

Рисунок 1 - Декодер (255, 251) кода Рида-Соломона, исправляющего двойные ошибочные байты

Таблица 1. Коэффициенты остатков от деления $A(z)z^i$ на $p(z)$

	Z^7	z^6	z^5	z^4	z^3	z^2	z^1	z^0
R_7	a_6+a_5+ $+a_4+a_0$	a_5+a_4+ $+a_3$	a_7+a_4+ $+a_3+a_2$	a_7+a_3+ $+a_2+a_1$	a_5+a_4+ $+a_2+a_1$	a_6+a_5+ $+a_3+a_1$	a_7+a_6+ $+a_2$	a_7+a_6+ $+a_5+a_1$
R_6	a_7+a_6+ $+a_5+a_1$	a_6+a_5+ $+a_4+a_0$	a_5+a_4+ $+a_3$	a_7+a_4+ $+a_3+a_2$	a_6+a_5+ $+a_3+a_2$	a_7+a_6+ $+a_4+a_2$	a_7+a_3	a_7+a_6+ $+a_2$
R_5	a_7+a_6+ $+a_2$	a_7+a_6+ $+a_5+a_1$	a_6+a_5+ $+a_4+a_0$	a_5+a_4+ $+a_3$	a_6+a_4+ $+a_3$	a_7+a_5+ $+a_3$	a_4	a_7+a_3
R_4	a_7+a_3	a_7+a_6+ $+a_2$	a_7+a_6+ $+a_5+a_1$	a_6+a_5+ $+a_4+a_0$	a_7+a_5+ $+a_4$	a_7+a_6+ $+a_4$	a_5	a_4
R_3	a_4	a_7+a_3	a_7+a_6+ $+a_2$	a_7+a_6+ $+a_5+a_1$	a_6+a_5+ $+a_0$	a_7+a_5	a_7+a_6	a_5
R_2	a_5	a_4	a_7+a_3	a_7+a_6+ $+a_2$	a_7+a_6+ $+a_1$	a_6+a_0	a_7	a_6
R_1	a_6	a_5	a_4	a_7+a_3	a_7+a_2	a_7+a_1	a_0	a_7
R_0	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0

Строке R_7 соответствует следующее выражение:

$R_7=(a_7z^7 + a_6z^6 + a_5z^5 + a_4z^4 + a_3z^3 + a_2z^2 + a_1z + a_0)z^7 \text{mod} p(z)=$

$=(a_6+a_5+a_4+a_0)z^7+(a_5+a_4+a_3)z^6+(a_7+a_4+a_3+a_2)z^5+(a_7+a_3+a_2+a_1)z^4+$

$+(a_5+a_4+a_2+a_1)z^3+(a_6+a_5+a_3+a_1)z^2+(a_7+a_6+a_2)z+(a_7+a_6+a_5+a_1);$

Используя полученные остатки, можно перейти от умножителей на константы к их реализации на элементах двоичной логики. Например, для псевдопеременной X^3:

$(a_7z^7+a_6z^6+a_5z^5+a_4z^4+a_3z^3+a_2z^2+a_1z+a_0)(z^4+z^3+z^2+z) \text{ mod} p(z)=R_4+R_3+R_2+R_1=$

$=(a_7+a_6+a_5+a_4+a_3)z^7+(a_6+a_5+a_4+a_3+a_2)z^6+(a_7+a_5+a_4+a_3+a_2+a_1)z^5+(a_7+a_6+$

$+a_4+a_3+a_2+a_0)z^4+(a_7+a_4+a_2+a_1+a_0)z^3+(a_7+a_5+a_4+a_1+a_0)z^2+(a_6+a_2+a_5+a_0)z+$

$+(a_7+a_6+a_5+a_4).$

По полученному остатку от деления строится схема умножителя на константу. Остальные элементы кодера и декодера для аппаратной реализации (255, 251) кода Рида-Соломона на элементах двоичной логики не требуют каких-либо специальных расчетов или методов построения.

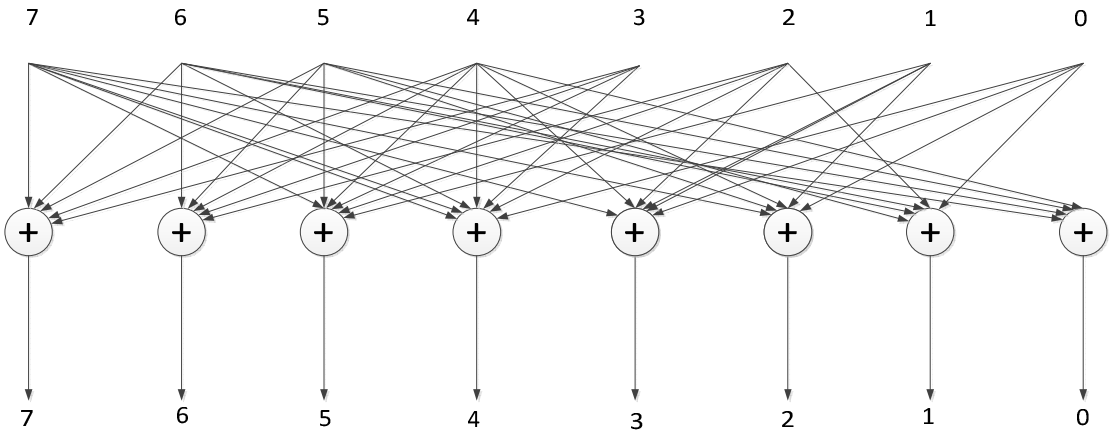

Рисунок 2 - Реализация умножителя на константу для псевдопеременной X^3 на элементах двоичной логики

Основная задача рисунка 2 – дать общее представление реализации умножителя на сумматорах по модулю два.

Рассмотренные способы декодирования и преобразования умножителей могут быть применимы не только для данного кода, но также для любых

кодов Рида-Соломона, исправляющих одиночные или двойные ошибки, посимвольно перемеженных и/или укороченных. И, наконец, следует отметить, что для генератора синдрома любого аппаратно реализуемого кода Рида-Соломона необходима замена умножителей на константу на основе примитивных элементов α сумматорами по модулю два элементами двоичной логики.

Выводы

Проведен анализ способов построения кодов Рида-Соломона, исправляющих двойные искаженные байты при параллельном и пакеты ошибок при последовательном способе применения. Рассмотрены детали аппаратной реализации декодирования на примере (255, 251) кода Рида-Соломона. Результаты показали, что для данного типа кодов рационально использовать синдромный метод декодирования. Дальнейшую работу планируется направить на проведение экспериментальных исследований кодов Рида-Соломона на FPGA, для чего будут использоваться отладочные платы фирм Xilinx и Altera, имеющиеся в распоряжении FPGA-лаборатории ДонНТУ [8].

Литература

1. Richard E.Blahut. Algebraic Codes for Data Transmission/ Cambridge University Press, 2012. – 498 p.

2. Рахман П.А. Основы защиты данных от разрушения. Коды Рида-Соломона/ Интернет-ресурс. – Режим доступа: URL http://icc.mpei.ru/documents/00000885.pdf Загл. с экрана.

3. Питерсон У., Уэлдон Э. Коды, исправляющие ошибки. – М.: Мир, 1976. – 595 с.: ил.

4. Anindya Sundar Das, Satyajit Das and Jaydeb Bhaumik. Design of RS (255, 251) Encoder and Decoder in FPGA// International Journal of Soft

Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January 2013, PP. 391–394.

5. Возовик К.П., Дяченко О.Н. Особенности аппаратной реализации и корректирующих возможностей кодов Рида-Соломона, исправляющих одиночные ошибки// Інформатика та комп'ютерні технології: Тез. доп. міжнародної наук.-техн. конф. – Донецьк, 15-16 грудня 2005. – С. 315-316.

6. Дяченко О.Н. Аппаратная реализация и корректирующие возможности кодов Рида-Соломона// Наукові праці Донецького національного технічного університету. Серія "Проблеми моделювання та автоматизації проектування динамічних систем" (МАП-2007). Випуск: 6 (127) – Донецьк: ДонНТУ. – 2007. – С. 113-121.

7. Зинченко Е.Ю., Дяченко О.Н. Сравнительный анализ способов укорачивания кодов Рида-Соломона// Збірка праць VII міжнародної науково-технічної конференції студентів, аспірантів та молодих науковців – 22-23 листопада 2011 р., Донецьк, ДонНТУ. – 2011. У 2-х томах, Т. 1 – С. 48-52.

8. Зинченко Ю., Калашников В., Хайдук С., Дяченко О. и др. FPGA-технологии проектирования и диагностика компьютерных систем/ Сборник научных трудов VI Междунар. научн.-практ. конф. «Современные информационные технологии и ИТ-образование». – Москва: МГУ, 2011. – Т. 1. 787 – С. 422-429.