

П. С. Поперечный^{1,2}, А. А. Беляев¹¹ ОАО НПЦ «ЭЛВИС», ² Институт проблем проектирования в микроэлектронике РАН

РЕАЛИЗАЦИЯ КОДЕРА РИДА-СОЛОМОНА С ИСПОЛЬЗОВАНИЕМ ЦИФРОВЫХ ФИЛЬТРОВ

В статье предложен способ построения схем кодирования при помощи цифровых фильтров. Данный подход позволяет использовать способы цифровой фильтрации (разбиение на фильтры меньшего порядка, конвейеризация) с целью изменения корректирующей способности кода. Приведено описание этого способа для применения в кодировании кодов РС (Рида-Соломона), аппаратно реализовано устройство кодера, приведены сравнительные характеристики.

Ключевые слова: коды РС, корректирующая способность, регистр с линейной обратной связью (РЛОС), КИХ- и БИХ-фильтры (фильтры с конечной и бесконечной импульсными характеристиками), поле Галуа.

Введение

Коды РС (Рида-Соломона) относятся к блочному кодированию и широко используются в системах хранения и передачи информации. Эти коды позволяют исправлять множественные ошибки в блоках данных – от нескольких бит до нескольких килобайт (увеличение блока данных приводит к аппаратным сложностям) [1].

В статье предложен метод построения кодера с использованием цифровых фильтров. Хотя арифметика в данном кодировании построена на полях Галуа, основные методы цифровой фильтрации применимы и в конечных полях. Такой подход позволяет разделить схему на последовательно соединенные фильтры первого порядка с возможностью конвейеризации с целью увеличения быстродействия. Также показано, что при таком подходе кодер легко становится реконфигурируемым, а именно включением, отключением определенных фильтров в цепи можно изменять корректирующую способность кода.

В настоящее время эти коды массово используются в таких системах хранения информации, как твердотельные накопители, флэш-память и др. Поэтому при использовании разных накопителей для работы с одним устройством необходимо применение кодов с разной корректирующей способностью соответственно. Например, для того чтобы в блоке данных используемый код позволял исправлять до 16 ошибок, необходимо применение определенного порождающего полинома определенной длины. Однако для того чтобы в этом же блоке данных код давал возможность исправлять, например, 12 ошибок, необходимо применение другого порождающего полинома с меньшей длиной. То есть

для использования одного и того же устройства с разными накопителями необходимо применение разных порождающих полиномов и, как следствие, разных кодеров, это приводит к увеличению аппаратных ресурсов. Однако использование кодера с регулируемой корректирующей способностью (переменный порождающий полином) может также удовлетворить различные требования к корректирующей способности [2].

Традиционный способ кодирования

Следуя определению кодов РС, систематическое кодирование осуществляется следующим образом:

$$\frac{m(x)x^{2t}}{g(x)} = q(x) + \frac{r(x)}{g(x)}, \quad (1)$$

где $m(x)$ – входные незакодированные данные; $g(x)$ – порождающий полином; t – количество исправляемых ошибок; $q(x)$ – частное от деления; $r(x)$ – остаток от деления на $g(x)$.

При этом результирующее кодовое слово (закодированные данные) в систематическом виде представляется как

$$c(x) = m(x)x^{2t} + r(x), \quad (2)$$

где $c(x)$ – кодовое слово.

Таким образом, данные на выходе кодера остаются неизменными, однако к ним добавляются контрольные данные $r(x)$.

Аппаратная реализация выражения (2) выполняется при помощи регистра с линейной обратной связью (РЛОС), представленного на рис. 1. В первые k тактов (k – количество информационных символов) работы схемы данные проходят на выход

схемы неизменными, при этом одновременно поступают на вход РЛОС, где с учетом обратной связи происходит вычисление остатка $r(x)$. После k тактов в схеме РЛОС отключается обратная связь, и значение остатка $r(x)$ фиксируется в сдвиговом регистре. В течение последующих $2t$ тактов из схемы выгружается значение остатка $r(x)$ и поступает на выход схемы.

При необходимости изменения требований к корректирующей способности кода РС следует изменить порождающий полином $g(x)$, что ведет к изменениям схемы РЛОС. В статье предлагается способ построения схемы деления на порождающий полином с возможностью с минимальными затратами изменять порождающий полином в процессе работы.

Способ реализации схемы РЛОС при помощи цифровых фильтров

Для реализации выражения (2) применяется схема РЛОС, представленная на рис. 1. Когда переключатели $P1, P2$ находятся в положении 1, схема РЛОС становится похожей на схему БИХ-фильтра (фильтр с бесконечной импульсной характеристикой) с отличием в том, что выход кодера не совпадает с выходом БИХ-фильтра. А именно выход РЛОС $z(x)$ выведен с регистра, однако в БИХ-фильтре с подобной реализацией выход данных $y(x)$ выведен с сумматора. Опишем математически работу схемы РЛОС при положении 1 переключателей $P1, P2$. Выход $y(x)$ БИХ-фильтра можно записать следующим образом:

$$y(x) = \frac{m(x)x^{2t}}{g(x)}. \tag{3}$$

Однако выход $z(x)$ РЛОС в случае для кодирования отличается, так как выведен до сумматора, а именно:

$$z(x) = \frac{m(x)x^{2t}}{g(x)} + m(x). \tag{4}$$

Эквивалентная схема, описываемая выражением (4), представлена на рис. 2.

Умножив и разделив последнее слагаемое $m(x)$ в выражении (4) на $\frac{x^{2t}}{g(x)}$, получим

$$z(x) = \frac{m(x)x^{2t}}{g(x)} + \frac{m(x)x^{2t}}{g(x)} \cdot \frac{g(x)}{x^{2t}}. \tag{5}$$

Эквивалентная схема, описываемая выражением (5), представлена на рис. 3.

Реализация конвейерного кодера РС при помощи цифровых фильтров

После включения переключателей $P1, P2$ в положение 2 в схеме на рис. 1 отключается цепь

обратной связи, а на вход схемы подается ноль. В эквивалентной схеме на рис. 3 на вход КИХ-фильтра вместо выхода БИХ-фильтра подается ноль. Так как в первые k -тактов схемы на выход кодера по определению систематического кодирования должны выдаваться данные без изменения, а последующие $2t$ тактов выдаются данные, вычисленные кодером, то можно пренебречь сумматором в схеме 3, добавить переключатель $P3$ и получится эквивалентная схема всего кодера, представленная на рис. 4.

Таким образом, в первые k тактов работы схемы на выход кодера поступают данные после БИХ- и КИХ-фильтров (фильтр с конечной импульсной характеристикой), соединенных последовательно, а так как оба фильтра имеют один и тот же полином, то данные на выходе будут поступать без

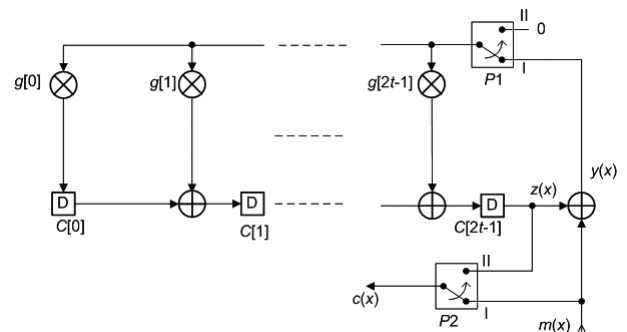


Рисунок 1. Традиционная схема кодера РС

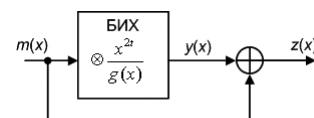


Рисунок 2. Эквивалентная схема РЛОС с применением БИХ-фильтра

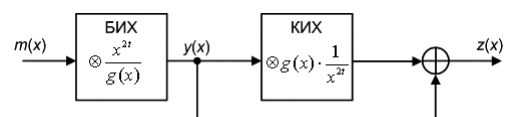


Рисунок 3. Эквивалентная схема РЛОС с применением БИХ- и КИХ-фильтров

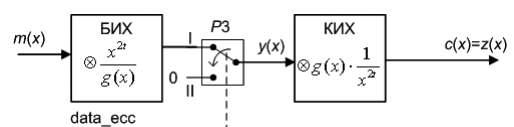


Рисунок 4. Эквивалентная схема кодера РС с применением БИХ- и КИХ-фильтров

изменений, что и требуется по определению кодера систематического кода РС. Однако в этой фазе происходит вычисление, и после k -го такта БИХ-фильтр из цепи отключается переключателем $P3$ и продолжает работать лишь КИХ-фильтр, подавая на выход кодера значения контрольных символов.

Такое построение эквивалентной схемы (рис. 4) позволяет избежать общей цепи обратной связи, охватывающей всю схему РЛОС как в схеме на рис. 1. Порождающий полином $g(x)$ для кодов РС представим в виде

$$g(x) = (x - a)(x - a^1) \dots (x - a^{2^t}), \quad (6)$$

где $a \in GF(2^m)$ – примитивный элемент поля Галуа размерности m .

Таким образом, схемы БИХ- и КИХ-фильтров можно разделить на последовательно соединенные фильтры первого порядка, как показано на рис. 5. А так как в таком построении схемы отсутствует общая цепь обратной связи с выхода на вход кодера, для уменьшения критического пути через цепь последовательно соединенных сумматоров можно конвейеризовать схему дополнительными регистрами, например, $reg1$ и $reg2$ и так далее при необходимости. При этом латентность схемы увеличивается на число тактов, равное количеству дополнительных конвейерных регистров, и быстродействие становится крайне высоким.

В представлении кодера в виде БИХ-фильтров, рассмотренном в [3–5], отсутствует основное преимущество подхода цифровой фильтрации. А именно цифровые фильтры, передаточная функция которых описывается в виде сомножителей, можно представить в виде последовательной цепи фильтров меньшего порядка, как показано на рис. 5 и 6.

Наилучшее применение схемы (рис. 5) получается при использовании дополнительного сигнала разрешения определенных фильтров. На рис. 6 представлена функциональная схема реконфигурируемого кодера РС с возможной конвейеризацией [6].

Схема на рис. 6 наделяет код корректирующей способностью от максимально возможного количества исправляемых ошибок T_{max} до 1. Из выражения (6) известно, что для того чтобы код позволял исправлять t ошибок, необходим порождающий

полином степени $2t$. Поэтому схема реконфигурируемого кодера рассчитана на полином максимальной степени $2T_{max}$. При необходимости кодировать данные кодом с числом исправляемых ошибок меньше максимального T_{max} необходимо с помощью сигналов разрешения отключить определенные фильтры (то есть часть множителей в полиноме (6) заменяется на 1), и результирующий полином получится необходимой степени, а значит, с заданной корректирующей способностью.

Процедура декодирования кодов РС

Декодирование кодов РС структурно показано на рис. 7. Принятые данные (с возможными ошибками) $v(x)$ поступают в схему декодера, одновременно происходит запись этих данных в буфер FIFO. Декодирование поделено на четыре основных этапа [7]. Сначала данные поступают в схему вычисления синдромов (признаки ошибок), в первые n тактов, потому как кодовое слово стало длины n . Дальнейшие вычисления декодер проводит с вычисленными синдромами. Следующий этап – вычисление полинома локаторов ошибок. Например, в алгоритме Берлекэмп-Мэсси (ВМА) без инверсии для данного этапа требуется $2t$ тактов (где t – количество исправляемых ошибок, с которым было закодировано переданное кодовое слово). Далее вычисленные коэффициенты уравнения поступают в схемы поиска позиций ошибок и в схему поиска значений ошибок. После вычисления полинома значений ошибок этот полином вместе с указателями позиций ошибок из схемы поиска позиций ошибок подаются на схему составления вектора ошибок. Эта схема формирует маску $e(x)$ для исправления искаженных данных $v(x)$ при считывании из буфера FIFO. При фиксированном параметре m поля Галуа декодер, реконфигурируемый по количеству исправляемых ошибок, реализуется изменением количества тактов, необходимых для работы алгоритма ВМА и схемы поиска значений ошибок. Таким образом, добавив к схеме декодера управление в зависимости от параметра t (максимальное количество исправляемых ошибок), декодер становится реконфигурируемым в зависимости от корректирующей способности, с которой было закодировано кодовое слово.

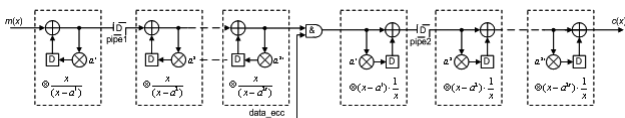


Рисунок 5. Функциональная схема кодера РС с применением БИХ- и КИХ-фильтров с конвейеризацией

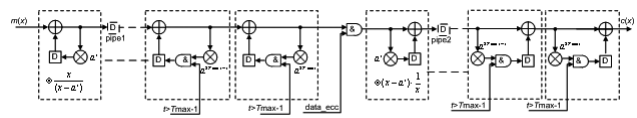


Рисунок 6. Функциональная схема реконфигурируемого кодера РС с применением БИХ- и КИХ-фильтров с конвейеризацией

Результаты экспериментов

На основе предложенной методики построения схемы кодирования при помощи цепи БИХ- и КИХ-фильтров разработано аппаратное описание кодера РС. Проведены сравнения по занимаемой площади (количество логических вентилях), а также по задержкам на критических путях для ПЛИС семейства Altera (Arria V). Ниже представлена таблица полученных характеристик для кодеров в поле Галуа $GF(2^m)$ при $m=8$ (длина закодированных данных не будет превышать $n > 2^8$) в зависимости от количества исправляемых ошибок. Анализ проведен в сравнении с каноническим представлением кодера.

Например, для $t=16$ (табл.) предложенный кодер будет кодировать данные наиболее типовыми кодами для хранения информации во flash-памяти, с количеством исправляемых ошибок 16, 15, 14 и т.д., согласно используемой flash-памяти, что делает его гораздо более универсальным в сравнении с каноническим кодером.

Результаты синтеза представлены также на рис. 8.

Результаты синтеза логической схемы показали следующее:

- Задержка на критическом пути предложенного кодера растет быстрее канонического в зависимости от количества исправляемых ошибок. Однако эксперимент проведен без использования конвейерных регистров. Очевидно, что с добавлением хотя бы одного конвейерного регистра



Рисунок 7. Структурная схема декодирования кодов РС

Таблица. Ресурсы ПЛИС (число ячеек), задержка на критическом пути в зависимости от количества исправляемых ошибок канонического кодера (const) и кодера на базе КИХ- и БИХ-фильтров (var)

t -число ошибок	Число ячеек		Задержка, нс	
	const	var	const	var
2	43	85	2,19	2,75
4	79	204	2,24	3,23
8	145	464	2,49	4,11
16	274	996	2,76	5,23

(рис. 6) задержка уменьшится в 2 раза и быстродействие увеличится в сравнении с каноническим видом кодера.

- Для конкретного семейства ПЛИС (Arria V) в среднем занимаемая площадь (в единицах логических ячеек) реконфигурируемого кодера в 2,5–3 раза больше кодера с постоянным многочленом.

Заключение

В данной статье описан способ применения цифровой фильтрации для устройств блочного кодирования. Алгоритмы и способы цифровой фильтрации позволили преобразовать схему кодера РС в удобную последовательность цифровых фильтров. Такое построение схемы делает кодер реконфигурируемым под разную корректирующую способность.

На основе предложенного способа разработан кодер, обладающий преимуществами при определенных условиях, например, при заведомо неизвестном количестве исправляемых ошибок. Показано, что разработанное устройство обладает большим быстродействием в сравнении с кодером традиционного вида.

При применении устройства, разработанного предлагаемым способом, для работы с flash-памятью различных типов более 3 делает его выгодным по занимаемой площади. По быстродействию при добавлении конвейерных регистров есть выигрыш в сравнении с устройствами канонического вида, где невозможно сделать конвейеризацию ввиду общей обратной связи [8].

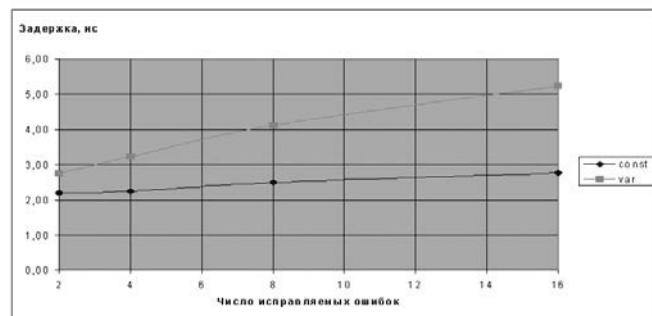
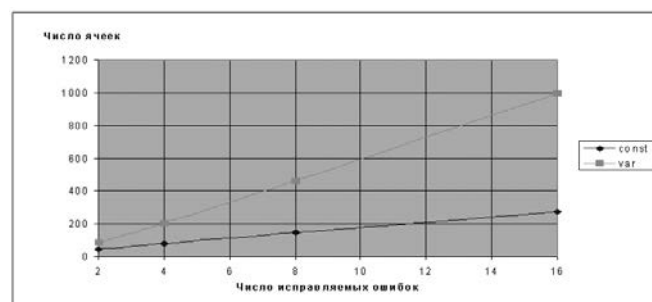


Рисунок 8. Ресурсы ПЛИС (число ячеек), задержка на критическом пути в зависимости от количества исправляемых ошибок канонического кодера (const) и кодера на базе КИХ-БИХ-фильтров (var)

СПИСОК ЛИТЕРАТУРЫ

1. Блейхут Р. Теория и практика кодов, контролирующих ошибки. М.: Мир, 1986. 576 с.
2. Yoo H., Lee Y. 7.3 Gb/s Universal BCH Encoder and Decoder for SSD Controllers // IEEE Symposium on Foundations of Computational Intelligence (FOCI 2014). Orlando, Florida, USA. 9–12 December 2014, pp. 37–38.
3. Кириллов С. Н., Семин Д. С. Модифицированный помехозащищенный кодер на основе БИХ-фильтра // Вестник РГТУ. 2009. № 2.
4. Patent US8464141 B2. Programmable error correction capability for BCH codes, filing date 13.08.2008, publication date 11.06.2013.
5. Zhang T., Parhi K. K. On the High Speed VLSI Implementation of Errorsand Erasures Correcting ReedSolomon Decoders. Department of Electrical and Computer Engineering, University of Minnesota.
6. Патент RU160960. Реконфигурируемый кодер Рида-Соломона, заявлено 21.07.2015, опубликовано 15.03.2016.
7. Вернер М. Основы кодирования. М.: Техносфера, 2004. 288 с.
8. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М.: Техносфера, 2005. 320 с.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Поперечный Павел Сергеевич, аспирант, инженер, ОАО НПЦ «ЭЛВИС», Москва, Зеленоград, проезд № 4922, д. 4, стр. 2, тел.: 8 (499) 731-75-02, e-mail: yncherepop@elvees.com.

Беляев Андрей Александрович, д.т.н., начальник лаборатории, ОАО НПЦ «ЭЛВИС», Москва, Зеленоград, проезд № 4922, д. 4, стр. 2, тел.: 8 (499) 731-75-02, e-mail: bel@elvees.com.

*For citation: Radiopromyshlennost. – 2016. – № 3. – P. 18–22.
P. Poperechnyi, A. Belyaev*

REED-SOLOMON CODER APPLICATION USING DIGITAL FILTERS

The article describes a method for establishing coding schemes using digital filters. This approach allows to use digital filtering methods (splitting into smaller mode filters, pipelining) for the purpose of changing the code correction capacity. The description of this method has been made for application in the coding of RS codes (Reed-Solomon), hardware encoder device implemented, comparative characteristics provided.

Keywords: Reed-Solomon codes, error-correcting capability, register with linear feedback (RLOS), FIR and IIR filters (filters with a finite/infinite impulse characteristics), Galois field.

REFERENCES

1. Bleykhut R. *Teoriya i praktika kodov, kontroliruyushchikh oshibki* [Theory and Practice of codes controlling Errors]. М.: Mir, 1986. 576 p.
2. Yoo H., Lee Y. 7.3 Gb/s Universal BCH Encoder and Decoder for SSD Controllers. IEEE Symposium on Foundations of Computational Intelligence (FOCI 2014). Orlando, Florida, USA. 9–12 December 2014, pp. 37–38.
3. Kirillov S. N., Semin D. S. Modifitsirovanny pomekhozashchishchenny koder na osnove BIKh-fil'tra [Modified noises protected coder based on an infinite impulse response filter]. *Vestnik RGTU*, 2009, no. 2.
4. Patent US8464141 B2. Programmable error correction capability for BCH codes.
5. Zhang T., Parhi K. K. On the High Speed VLSI Implementation of Errorsand Erasures Correcting ReedSolomon Decoders. Department of Electrical and Computer Engineering, University of Minnesota.
6. Patent RU160960. Rekonfiguriruemy koder Rida-Solomona [Patent RU160960. reconfigurable Reed-Solomon encoder].
7. Verner M. *Osnovy kodirovaniya* [Basics of coding]. М.: Tekhnosfera, 2004. 288 p.
8. Morelos-Saragosa R. *Iskusstvo pomekhoustoychivogo kodirovaniya. Metody, algoritmy, primeneniye* [Art of noise-proof coding. The methods, algorithms, applications]. М.: Tekhnosfera, 2005. 320 p.

AUTHORS

Poperechnyi Pavel, Postgraduate, Engineer of Open Joint-Stock Company Research and Development Center «ELVEES», Moscow, tel.: +7 (499) 731-7502, e-mail: yncherepop@elvees.com.

Belyaev Andrey, PhD, Head of DSP-lab of Open Joint-Stock Company Research and Development Center «ELVEES», Moscow, tel.: +7 (499) 731-7502, e-mail: bel@elvees.com.