

Верификация аппаратных описаний алгебраических блоков в среде MatLab

П.С. Поперечный^{1,2}, А.А. Беляев¹, И.Ю. Колесникова¹

¹*ОАО Научно-производственный центр «ЭЛВИС» (г. Москва)*

²*Институт проблем проектирования в микроэлектронике
Российской академии наук (г. Москва)*

Verification of an Algebraic Block Hardware Description in MatLab Packet

P.S. Poperechny^{1,2}, A.A. Belyaev¹, I.Y. Kolesnikova¹

¹*Open Joint-Stock Company Research and Development
Center «ELVEES», Moscow*

²*Institute of Design Problems in Microelectronics of the Russian Academy of Sciences, Moscow*

Предложен способ построения схем алгебраического кодирования Рида – Соломона с помощью цифровых фильтров. Данный подход позволяет использовать способы цифровой фильтрации (разбиение на фильтры меньшего порядка, конвейеризацию) с целью изменения корректирующей способности кода. Для верификации разработанного блока кодирования предложено сравнивать алгебраические результаты кодирования для случайных данных с имеющимся в библиотеке MatLab эталонным блоком кодера Рида – Соломона. Приведены схемы разработанного кодера для общего случая с возможностью изменять параметры под любой код. Проведен полный цикл верификации со сравнением результатов с эталонным кодером, покрытием тестами, возможностью отладки и корректировки исходного аппаратного описания.

Ключевые слова: кодирование Рида – Соломона; корректирующая способность; регистр с линейной обратной связью; фильтры с конечной/бесконечной импульсной характеристикой; поле Галуа; верификация; покрытие тестами.

The method of building the Reed-Solomon (RS) algebraic coding circuits using the digital filters has been proposed. This approach permits to use the digital filtration methods (partition to the lower order filters, conveyorization) for the purpose to change the code correcting capability. To verify the developed coding block it has been proposed to compare the coding algebraic results for random data with the standard (available in the MatLab library) block of the Reed-Solomon coder. The developed coder circuits for general case with the possibility to change the parameters for any code have been given. The complete cycle of verification with the comparison of the results with the standard coder, the tests coverage, the capability of debugging and error-correction of the initial hardware description has been performed.

Keywords: Cadence; Galois field; coder; cosimulation; coverage; MatLab; Reed-Solomon; simulator; verification.

Введение. Кодирование Рида – Соломона (РС) относится к алгебраическому кодированию [1, 2]. Это означает, что при верной реализации результат кодирования (и декодирования) различными способами должен быть одинаковым. В настоящей работе предложен новый способ верификации и отладки такого рода устройств на основе интерактивной косимуляции в пакетах моделирования MatLab и Cadence. Способ обеспечивает надежную верификацию разработанного блока за счет сравнения его с библиотечным эталоном, ускоряет процессы верификации и проверки устройства на основе широкого набора входных воздействий.

Традиционный способ кодирования РС. Согласно определению кодов РС систематическое кодирование осуществляется следующим образом [3]:

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

где $m(x)$ – входные закодированные 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 $c(x)$ (закодированные данные) представляется как

$$c(x) = m(x)x^{2t} + r(x). \quad (1)$$

Аппаратная реализация выражения (1) выполняется с помощью регистра с линейной обратной связью (РЛОС), представленного на рис.1.

При необходимости изменения требований к корректирующей способности кода РС следует изменить порождающий полином $g(x)$ и, следовательно, схему РЛОС. Предлагается способ построения схемы деления на порождающий полином с возможностью изменять порождающий полином в процессе работы с минимальными затратами.

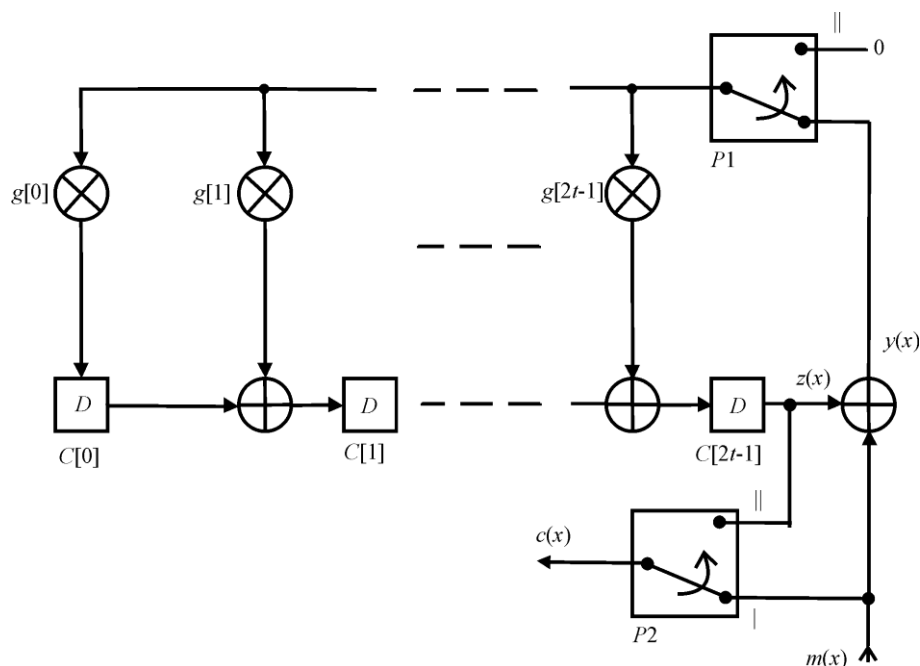


Рис.1. Традиционная схема кодера РС

Способ реализации кодера РС с помощью цифровых фильтров. Для реализации выражения (1) применяется схема РЛОС, представленная на рис.1. Когда переключатели $P1, P2$ находятся в положении I, схема РЛОС становится похожей на схему фильтра с бесконечной импульсной характеристикой (БИХ) 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 I переключателей $P1, P2$. Выход $y(x)$ БИХ-фильтра можно записать следующим образом:

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

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

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

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

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

$$z(x) = \frac{m(x)x^{2t}}{g(x)} + \frac{m(x)x^{2t}}{g(x)} \cdot \frac{g(x)}{x^{2t}}. \quad (3)$$

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

После включения переключателей $P1, P2$ в положение II в схеме на рис.1 отключается цепь обратной связи, а на вход схемы подается ноль. В схеме на рис.3 на вход фильтра с конечной импульсной характеристикой (КИХ) вместо выхода БИХ-фильтра подается ноль. В первые k тактов схемы на выход кодера по определению систематического кодирования должны выдаваться данные без изменения. В следующие $2t$ такта выдаются данные, вычисленные кодером. Можно пренебречь сумматором в схеме на рис.3, добавив переключатель $P3$. В результате получим эквивалентную схему всего кодера, представленную на рис.4.

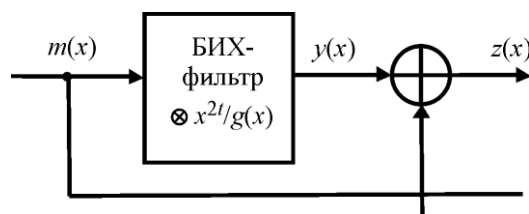


Рис.2. Эквивалентная схема РЛОС с применением БИХ-фильтра

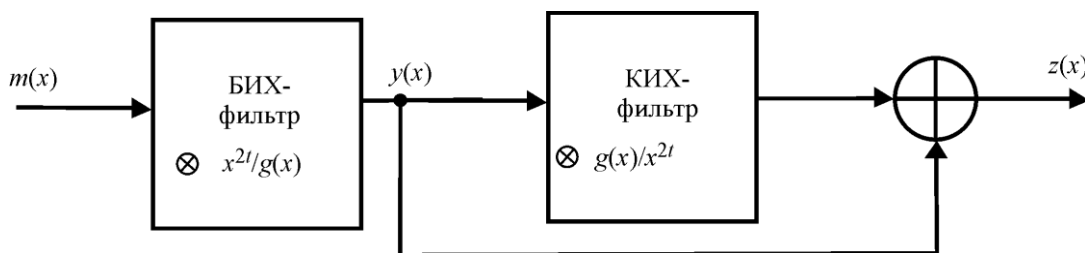


Рис.3. Эквивалентная схема РЛОС с применением БИХ- и КИХ-фильтров

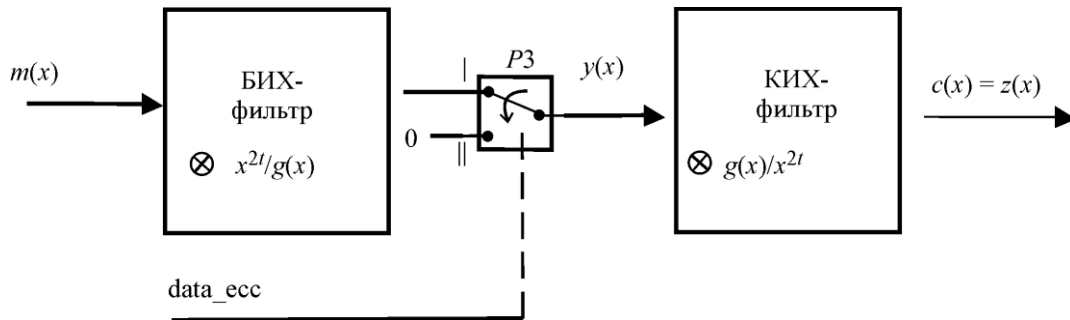


Рис.4. Эквивалентная схема кодера РС с применением БИХ и КИХ-фильтров

В первые k тактов работы схемы на выход кодера поступают данные после БИХ- и КИХ-фильтров, соединенных последовательно. Оба фильтра имеют один и тот же полином, поэтому данные на выходе будут поступать без изменений согласно определению кодера систематического кода РС. 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 (см. 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

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

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

Таким образом, схемы БИХ- и КИХ-фильтров можно разбить на последовательно соединенные фильтры первого порядка. Именно этим предлагаемый способ отличается от представленного в [4].

Верификация разработанного блока кодера. Коды РС относятся к алгебраическим кодам. Это означает, что при кодировании (при заданном примитивном полиноме поля Галуа) кодовое слово должно определяться однозначно. Для функциональной верификации таких блоков рациональнее сравнивать результаты алгебраических операций с эталонным блоком. В качестве эталонного с заданными параметрами (размер данных, примитивный полином и т.д.) предлагается использовать имеющийся в библиотеке MatLab/Simulink кодер РС. В последующей разработке декодера РС можно также использовать эталонный декодер и стандартный блок канала «аддитивный белый гауссовый шум», двоичный симметричный канал, модуляторы, демодуляторы для создания полноценного тестового окружения и снятия характеристик разрабатываемого кодера.

Таким образом, для функциональной проверки алгебраических блоков со стандартными моделями каналов, кодеров, декодеров и пр. наилучшим образом подходит пакет MatLab/Simulink с уже имеющейся библиотекой перечисленных блоков. Можно непосредственно использовать общеизвестные симуляторы аппаратного описания (ModelSim, QuestaSim, NcSim и т.д.). Однако в таком случае необходимо для тестового окружения разрабатывать и стандартные блоки (модели каналов, источники случайных данных и пр.).

Рассмотрим способ моделирования аппаратного описания на примере кодера РС в среде MatLab с интерактивной симуляцией и отладкой в среде NcSim (Cadence) [5].

Для создания схемы тестового окружения в среде MatLab необходимо открыть косимуляционное окно, добавив исходные файлы, например Verilog или VHDL, и установить значения параметров согласно заданным условиям.

В соответствии с входными воздействиями создается тестовое окружение из блоков MatLab для косимуляционного блока. На рис.5 представлена схема сравнения косимуляционного блока (разработанного аппаратного verilog-описания) с эталонным кодером РС (RS Encoder). Результат кодирования сравнивается с блоком подсчета возникновения ошибок (BER).

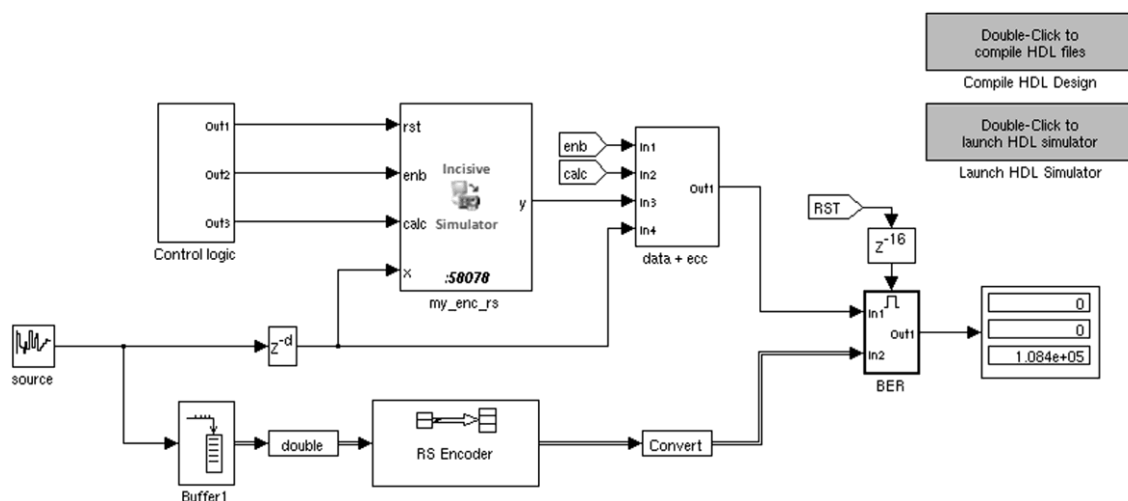


Рис.5. Схема тестового окружения для сравнения разработанного и эталонного кодеров РС

В данной схеме, собранной в пакете MatLab, применяется интерактивный косимуляционный блок Incisive Simulator, который использует инструменты симулятора аппаратных описаний NCSim (Cadence). Для анализа покрытия модели тестами в MatLab через симулятор NCSim при создании косимуляционного блока (см. рис.5) нужно добавить соответствующие опции компиляции.

Заключение. Описанные алгоритмы и способы цифровой фильтрации позволяют преобразовать схему кодера РС в удобную последовательность цифровых фильтров, обеспечивая реконфигурируемость кодера под разную корректирующую способность. Для рассматриваемого алгебраического кодера разработано тестовое окружение в среде MatLab, предложен способ реализации интерактивного взаимодействия пакетов моделирования MatLab и Cadence с использованием лишь исходных файлов.

Рассмотренный подход упрощает и ускоряет процесс разработки тестового окружения. Сравнение результатов кодирования случайных данных для разрабатываемого блока и его библиотечного эталона обеспечивает надежную верификацию аппаратного описания блока.

Литература

1. Блейхут Р. Теория и практика кодов, контролирующих ошибки. – М.: Мир, 1986. – 576 с.
2. Вернер М. Основы кодирования. – М.: Техносфера, 2004. – 288 с.
3. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. – М.: Техносфера, 2005. – 320 с.

4. **Кириллов С.Н., Семин Д.С.** Модифицированный помехозащищенный кодер на основе БИХ-фильтра // Вестник РГРТУ. – 2009. – № 2.

5. The MathWorks.Inc, HDL-Verifier. –
URL: <http://www.mathworks.com/products/hdl-verifier/webinars.html/> (дата обращения: 8 ноября 2015 г.)

Статья поступила
8 декабря 2015 г.

Поперечный Павел Сергеевич – инженер ОАО НПЦ «Элвис» (г. Москва), аспирант ИППМ РАН. *Область научных интересов:* разработка кодеков блочных кодов помехоустойчивого кодирования, способов параллелизации алгоритмов кодирования/декодирования, способов реконфигурирования исправляющей способности кодеков в системе на кристалле (СнК) в процессе работы.
E-mail: poperrechny@elvees.com

Беляев Андрей Александрович – доктор технических наук, начальник лаборатории ОАО НПЦ «Элвис» (г. Москва). *Область научных интересов:* алгоритмы цифровой обработки сигналов (ЦОС), разработка ядра цифрового сигнального процессора в составе СнК, аппаратная реализация алгоритмов видеокompрессии, обработки изображений, видеоаналитики.

Колесникова Ирина Юрьевна – инженер ОАО НПЦ «Элвис» (г. Москва). *Область научных интересов:* функциональная верификация аппаратных описаний алгоритмов ЦОС, разработка программы для автоматической генерации аппаратного описания некоторых операций в полях Галуа для блочного кодирования, разработка тестового окружения в среде MatLab для тестирования аппаратных описаний кодеков.

Информация для читателей журнала «Известия высших учебных заведений. Электроника»

Вы можете оформить подписку на 2016 г. в редакции с любого номера. Стоимость одного номера – 1000 руб. (с учетом всех налогов и почтовых расходов).

Адрес редакции: 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1, МИЭТ, комн. 7231.

Тел.: 8-499-734-62-05. E-mail: magazine@miee.ru

<http://www.miet.ru>