

Реконфигурируемый блок помехоустойчивого кодирования для систем на кристалле

П.С. Поперечный

Институт проблем проектирования в микроэлектронике РАН (ИППМ РАН),

Научно-производственный центр «ЭЛВИС», prpoperechny@elvees.com

Аннотация — В статье предложен метод реализации устройства помехоустойчивого кодирования в составе системы на кристалле (СнК) на примере кодера БЧХ (Боуза — Чоудхури — Хоквингема) с параллельной обработкой поступающих данных, а также с возможностью в процессе работы устройства изменять корректирующую способность кода. Выведено аналитическое выражение для параллельной реализации кодера с переменным количеством исправляемых ошибок. Предложено данное устройство кодирования реализовано аппаратно. Приведено подробное описание устройства, а также выполнено сравнение с аналогичными устройствами с различной шириной входной шины данных с помощью современных САПР на базе ПЛИС и СБИС.

Ключевые слова — помехоустойчивое кодирование, корректирующая способность, регистр с линейной обратной связью (РЛОС).

I. ВВЕДЕНИЕ. ПОСТАНОВКА ЗАДАЧИ

Устройства помехоустойчивого кодирования могут использоваться в СнК, в состав которого входят контроллеры памяти. Для работы с твердотельными носителями информации, такими как flash-память, SSD диски, используются коды БЧХ (Боуза — Чоудхури — Хоквингема). Коды БЧХ относятся к блочному кодированию и широко используются в системах хранения и передачи информации. Данные коды позволяют исправлять множественные ошибки в блоках данных от нескольких бит до нескольких килобайт (увеличение блока данных приводит к аппаратным сложностям) [1]. Ввиду использования разных накопителей для работы с одним устройством, необходимо применение кодов с разной корректирующей способностью. Например, для того чтобы в блоке данных используемый код позволял исправлять до 16 ошибок, необходимо применение определенного порождающего полинома. Однако, для того чтобы в этом же блоке данных код позволял 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, что приводит к увеличению аппаратных ресурсов. Однако использование кодера с регулируемой

корректирующей способностью (переменный порождающий полином) может также удовлетворить различные требования к корректирующей способности [2]. При этом для использования данного кодирования в системах хранения данных чаще всего возникает необходимость кодировать данные поступающие параллельно, то есть с шины данных, что делает необходимым использовать параллельный кодер. Существуют способы распараллеливания «разворачиванием» аналитического выражения [3], однако отсутствует единый подход для требуемой ширины шины данных [4].

В данной статье предложен метод построения параллельного кодера с возможностью изменения корректирующей способности кода и выведено аналитическое выражения для аппаратной реализации построения подобных устройств. Разработанное устройство к тому же обладает высоким быстродействием по сравнению с традиционным кодером с последовательной обработкой входного потока данных.

На рис. 1 представлен пример построения СнК, в состав которого входит контроллер flash-памяти. Как правило, данный контроллер включает в себя блок ECC (error-correcting code) – блок кодера для дополнительной проверочной информации, записываемой в память. Таким образом, даже при внесении ошибок в память по нескольким причинам возможно исправление этих ошибок при считывании и, соответственно, декодировании.

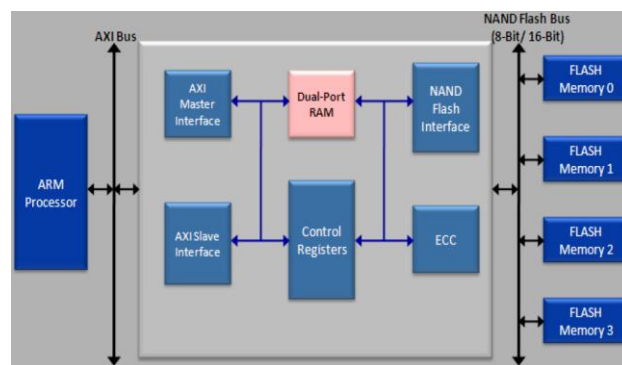


Рис. 1. Пример построения СнК с контроллером памяти

Из определения кодов БЧХ систематическое кодирование осуществляется следующим образом [5]:

$$\frac{u(x) \cdot x^{n-k}}{g(x)} = q(x) + \frac{r(x)}{g(x)}, \quad (1)$$

где $u(x)$ - входные незакодированные данные,

$g(x)$ - порождающий полином,

n - длина кодового слова (длина закодированных данных),

k - длина незакодированных данных,

$q(x)$ - частное от деления,

$r(x)$ - остаток от деления на $g(x)$.

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

$$c(x) = u(x) \cdot x^{n-k} + r(x), \quad (2)$$

где $c(x)$ - кодовое слово.

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

Аппаратная реализация выражения (2) (рис. 2) выполняется при помощи регистра с линейной обратной связью (РЛОС).

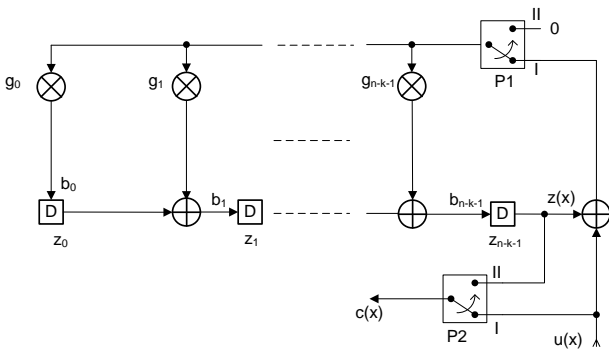


Рис. 2. Традиционная схема кодера БЧХ

В течение первых k тактов работы схемы данные проходят на выход схемы в неизменном виде, при этом одновременно они 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 $n-k$ тактов из схемы выгружается значение остатка $r(x)$, который и поступает на выход схемы.

II. КОДЕР БЧХ С РЕГУЛИРУЕМОЙ КОРРЕКТИРУЮЩЕЙ СПОСОБНОСТЬЮ

Для изменения требований к корректирующей способности кода БЧХ необходимо изменить порождающий полином $g(x)$, что ведет к изменению схемы РЛОС [6, 7]. Предлагаемое устройство включает в себя способ построения схемы РЛОС с возможностью изменять порождающий полином в процессе работы с минимальными затратами.

Для реализации выражения (2) с переменной корректирующей способностью применяется схема РЛОС с настраиваемыми коэффициентами порождающего полинома и узлами управления (рис. 3).

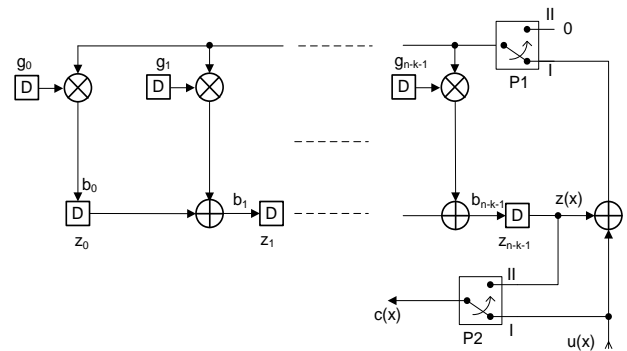


Рис. 3. Кодер БЧХ с регулируемой корректирующей способностью

Первый k тактов работы устройства данные $u(x)$ без изменений проходят на выход схемы и одновременно поступают на вход РЛОС. Переключатели P1 и P2 находятся в положении I. На данном этапе работы устройства происходит вычисление остатка $r(x)$. После k тактов работы переключатели P1 и P2 переходят в состояние II, отключая тем самым петлю обратной связи и подключая РЛОС к выходу схемы. В течение последующих $n-k$ тактов происходит выгрузка остатка $r(x)$ из сдвигового регистра $z_0, z_1, \dots, z_{n-k-1}$. Перед началом кодирования очередного блока данных регистры $z_0, z_1, \dots, z_{n-k-1}$ сброшены в нулевое состояние, и значения $b_0, b_1, \dots, b_{n-k-1}$, поступающие на вход регистров с выходов умножителей, можно описать с помощью следующего итеративного выражения:

$$\begin{cases} b_j = z_{j-1} + (z_{n-k-1} + u_i) \cdot g_j, & j \in [0, n-k-1] \\ b_i = z_i, & i \in [0, k-1] \end{cases} \quad (3)$$

где j - это позиция соответствующего умножителя g_j или регистра z_j ,

i - номер текущего такта работы схемы,

u_i - значение символа входных данных в текущем такте i .

Далее следующие $(n-k)$ тактов схема РЛОС работает без обратной связи, просто как сдвиговый регистр, поэтому значения сигналов $b_0, b_1, \dots, b_{n-k-1}$ на входе регистров можно описать следующим образом:

$$\left\{ \begin{array}{l} b_j = z_{j-1}, \quad j \in [0, n-k-1] \\ i \in [k, n-1] \end{array} \right. \quad (4)$$

Все операции выполняются в поле Галуа $GF(p^m)$. Временная диаграмма работы схемы на рисунке 3 представлена на рис. 4.

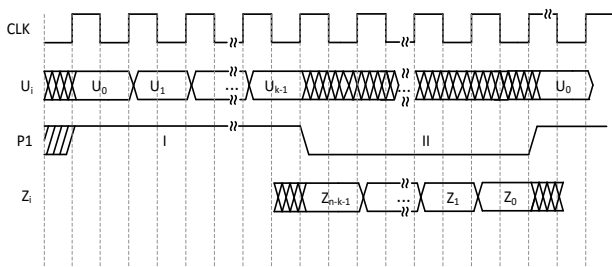


Рис. 4. Временная диаграмма работы последовательного кодера

Для двоичных кодов БЧХ, а именно такие коды наиболее распространены в системах хранения и передачи информации, схема на рис. 2 упрощается, а именно:

- умножители вырождаются в элементы «И»,
- сумматоры - в элементы «исключающее ИЛИ» (полусумматор),
- регистры, хранящие коэффициенты $g_0, g_1, \dots, g_{n-k-1}$ порождающего полинома, становятся одноразрядными.

Схема кодера двоичных кодов БЧХ представлена на рис. 5.

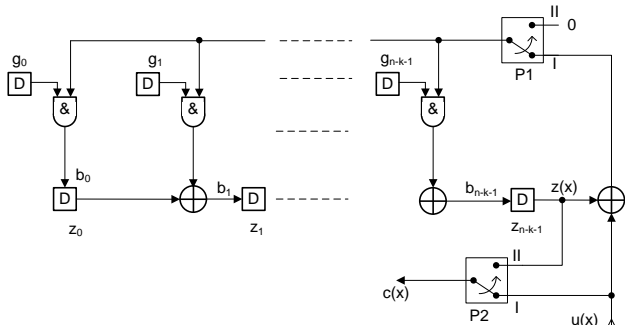


Рис. 5. Кодер двоичных кодов БЧХ с регулируемой корректирующей способностью

Арифметические операции в выражениях (3, 4), а также в схеме на рис. 5 выполняются в расширенном (m -степень поля) поле Галуа $GF(2^m)$. Так как порождающий полином можно задавать любой степени (от максимального количества значащих коэффициентов $(n-k)_{\max}$ до m), кодер является

реконфигурируемым под требуемое количество исправляемых ошибок (от максимального $t_{\max} = \frac{(n-k)_{\max}}{m}$ до 1) для выбранного кода БЧХ. Для

изменения требуемой корректирующей способности кода необходимо перед этапом кодирования перезаписать коэффициенты обновленного полинома. При этом необходимо учитывать, что значащий коэффициент при самой старшей степени полинома должен быть в месте регистра g_{n-k-1} , поэтому если требуемый полином имеет степень меньшую, чем заложено в данной реализации $(n-k)_{\max}$, то необходимо регистры при младших степенях обнулить. В таблице 1 показан для случая, когда порождающие полиномы имеют степени 10 (первая строка) и 8 (вторая строка).

Таблица 1

Пример заданных полиномов со степенями 10 и 8

коэффициенты полинома в соответствующих регистрах							
0	1,2,...,n-k-1	0	n-k-9	n-k-8	n-k-7	n-k-1 (n-k) _{max}
0	0	g[0]	g[1]	g[2]	g[3]	...	g[9] g[10]
0	0	0	0	g[0]	g[1]	...	g[7] g[8]

III. ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ КОДЕРА БЧХ С РЕГУЛИРУЕМОЙ КОРРЕКТИРУЮЩЕЙ СПОСОБНОСТЬЮ

Существуют способы распараллеливания «разворачиванием» аналитического выражения [8-12], однако отсутствует единый подход для требуемой ширины шины данных [13, 14].

Выражение (3) описывающее работу схемы на рис. 3, представляет значения, поступающие на вход регистров. В следующий такт значения с этих регистров используются уже для обработки следующего поступившего бита информации. Если на вход схемы в текущий такт работы поступает сразу несколько бит входных данных, то по аналогии с последовательно схемой можно составить схему из стадий для каждого бита данных. А именно, первый такт обрабатывается первой стадией схемы на рис. 5, и те значения $b_{1,j}$ $j \in [0, n-k-1]$, которые в случае последовательной схемы (рис. 3) поступали бы на вход регистров, попадают на следующую стадию со сдвигом вправо и участвуют в обработке следующего бита данных в шине. Выражение, описывающее первую стадию схемы, аналогично выражению (3):

$$\left\{ \begin{array}{l} b_{0,j} = z_j, \quad j \in [0, n-k-1] \\ b_{1,j} = b_{0,j-1} + (b_{0,n-k-1} + u_{0,i}) \cdot g_j, \quad j \in [0, n-k-1] \end{array} \right.$$

И так далее для каждой следующей стадии пока все биты данных в шине в текущем такте не будут обработаны и после последней стадии вычисленные значения не запишутся в регистры. Таким образом, в один такт работы данной схемы последовательная

схема должна проработать L тактов для обработки того же количества бит информации.

Реализация кодера с параллельным потоком данных [15, 16] представлена на рис. 6.

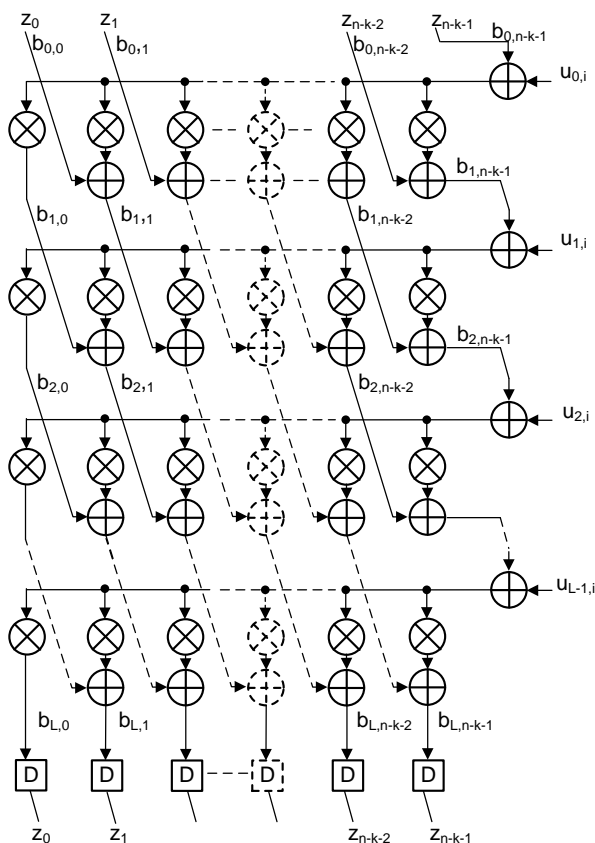


Рис. 6. Параллельный кодер кодов БЧХ

Значения $b_{h,0}, b_{h,1}, \dots, b_{h,n-k-1}$ с выходов умножителей на каждой стадии, а именно на каждой параллельной ступени h потока данных $u_{h,i}$ можно представить следующим образом:

$$b_{0,j} = z_j, \quad j \in [0, n-k-1]$$

$$\left\{ \begin{aligned} b_{h,j} &= b_{h-1,j-1} + (b_{h-1,n-k-1} + u_{h-1,i}) \cdot g_j, \quad j \in [0, n-k-1], \quad (5) \\ &h \in [1, L] \\ &i \in [0, \frac{k}{L} - 1] \end{aligned} \right.$$

где $u_{h,i}$ - значение h -го символа слова данных поступивших в i -ый такт.

Количество тактов работы схемы уменьшилось в L (размерность шины данных) раз с k до k/L , так как данные поступают параллельно. К моменту k/L такта данные заканчиваются, и вычисленные контрольные символы хранятся в регистрах $z_0, z_1, \dots, z_{n-k-1}$.

Временные диаграммы работы схемы представлены на рис. 7.

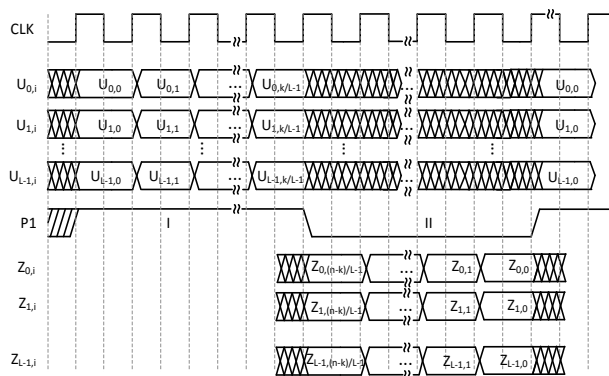


Рис. 7. Временная диаграмма работы параллельного кодера

Данная схема значительно упрощается в кодировании двоичных кодов БЧХ. Для двоичных кодов БЧХ умножители в поле Галуа $GF(p^m)$ в схеме рис. 6 заменяются простым элементом «И», сумматоры выполняют сложение по модулю 2, так как работают в поле Галуа $GF(2^m)$, как показано в схеме на рис. 8.

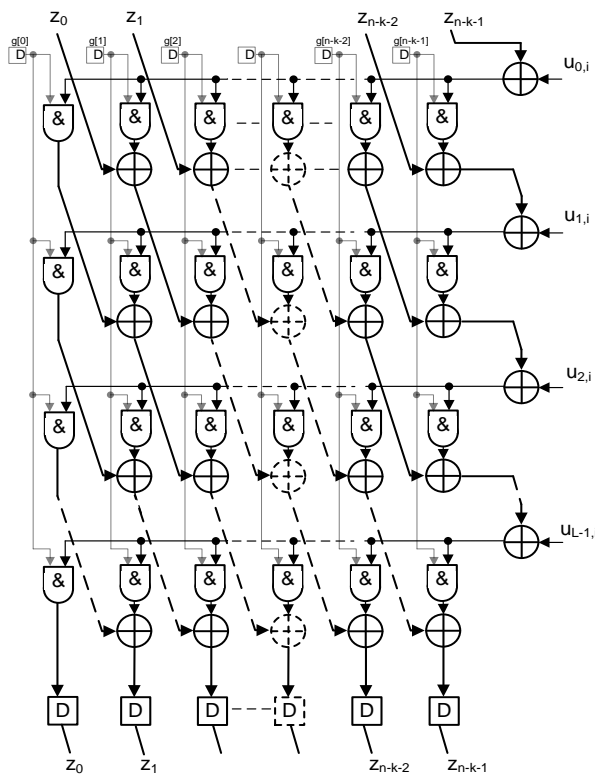


Рис. 8. Параллельный кодер кодов БЧХ с регулируемой корректирующей способностью

Регистры, хранящие коэффициенты $g_0, g_1, \dots, g_{n-k-1}$ порождающего полинома, являются одноразрядными. Алгоритм работы аналогичен для схемы, описанной для рис. 7, однако кодер при этом стал

реконфигурируемым. Простое изменение состояния регистров $g_0, g_1, \dots, g_{n-k-1}$ настраивает корректирующую способность схемы. Критический путь немногим больше пути кодера без возможности реконфигурирования:

$$\tau = 2 \cdot L \cdot \tau_{\oplus} + L \cdot \tau_{\otimes}, \quad (6)$$

где τ_{\oplus} - задержка на элементе сумматора (исключающего «ИЛИ»), τ_{\otimes} - задержка на элементе «И».

Критический путь в L раз больше пути в кодере последовательного потока данных (рис. 5), однако следует учитывать, что каждый такт на входе параллельного кодера (рис. 8) поступает L бит с шины данных, что не должно сказаться на пропускной способности.

IV. ПРОЦЕДУРА ДЕКОДИРОВАНИЯ БЧХ КОДОВ

Декодирование кодов БЧХ структурно показано на рис. 9. Принятые данные (с возможными ошибками) $v(x)$ поступают в схему декодера, одновременно происходит запись этих данных в буфер FIFO. Декодирование поделено на три основных этапа [17].

1. Сначала данные поступают в схему вычисления синдромов (признаки ошибок) в течение первые n/L тактов, так как кодовое слово стало длины n . Дальнейшие вычисления декодер проводит с вычисленными синдромами.

2. Следующий этап – вычисление полинома локаторов ошибок. Например, в алгоритме Берлекэмп – Мэсси (ВМА) без инверсии для данного этапа требуется t тактов (где t – количество исправляемых ошибок в кодовом слове).

3. Далее вычисленные коэффициенты уравнения поступают в схему поиска позиций ошибок, в этот же момент происходит считывание данных с буфера, а схема поиска позиций ошибок выдает маску, при сложении с которой искаженные данные исправляются и поступают на выход схемы. При фиксированном параметре m поля Галуа декодер, реконфигурируемый по количеству исправляемых ошибок, реализуется изменением количества тактов, необходимых для работы алгоритма ВМА.

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.

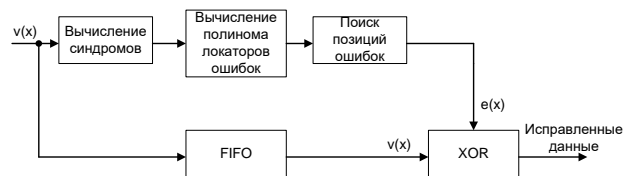


Рис. 9. Структурная схема декодирования БЧХ кодов

V. ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

Проведены сравнения по занимаемой площади (количество логических вентилях), а также по задержкам на критических путях, для ПЛИС семейства Altera (Arria II), а также для СБИС технологии 28 нм (Silterra). В таблицах 2-3 представлены полученные характеристики для кодеров с максимальной степенью полинома $R=208$. А именно, степень полинома:

$$R = n - k = m \cdot t, \quad (7)$$

то есть для поля $GF(2^m)$ при $m=13$ длина закодированных данных не будет превышать $n < 2^{13}$, а количество исправляемых ошибок, исходя из выражения (7), будет не более $t_{\max} = \frac{R}{m} = \frac{208}{13} = 16$.

Данный размер полинома позволяет кодировать наиболее типовыми кодами для хранения информации во flash-памяти, с количеством исправляемых ошибок 16, 15, 14 и т.д. согласно используемой flash-памяти [6].

Результаты синтеза представлены также на графике (рис. 10).

Результаты экспериментов по логическому синтезу показали следующее:

- Задержка на критическом пути не зависит от степени используемого полинома, а зависит лишь от ширины шины данных, что и подтверждает выражение (6).
- Занимаемая площадь (единицы логических ячеек) реконфигурируемого кодера в 5-5.5 раз больше кодера с постоянным многочленом.
- Задержка на критическом пути мало зависит от возможности реконфигурирования кодеров, в основном от ширины шины данных.
- При увеличении ширины шины данных увеличивается критический путь, однако пропускная способность при этом также увеличивается, что делает возможным подавать данные с внешней шины с более низкой частотой.

Таблица 2

Ресурсы ПЛИС, задержка на критическом пути, пропускная способность в зависимости от ширины шины данных кодеров (R=208) с постоянным многочленом и с переменным

L	кол-во ячеек		задержка, нс		пропускная способность, Мбит/с	
	const	var	const	var	const	var
1	120	224	1,15	1,37	869	729
2	163	226	1,25	1,76	1600	1136
4	202	453	1,56	2,66	2564	1503
8	246	1128	2,80	3,60	2857	2222
16	392	2250	5,59	7,07	2862	2263
24	642	3139	8,77	9,80	2736	2448
32	837	4037	11,21	12,27	2854	2607
40	1017	4932	15,15	15,18	2640	2635
48	1123	5829	18,65	18,78	2573	2555
56	1305	6500	21,23	20,83	2637	2688
64	1500	7395	24,60	24,50	2601	2612

Таблица 3

Площадь (28 нм), задержка на критическом пути, пропускная способность в зависимости от ширины шины данных кодеров (R=208) с постоянным многочленом и с переменным

L	Площадь, мкм ²		задержка, нс		пропускная способность, Мбит/с	
	const	var	const	var	const	var
1	800	1000	0,30	0,65	3333	1538
4	970	1800	0,40	0,80	10000	5000
8	1330	3650	0,50	1,00	16000	8000
16	2300	7400	0,80	1,10	20000	14545
32	3250	14800	1,20	1,90	26666	16842
48	4100	21400	2,00	2,70	24000	17777
64	5500	28100	2,50	3,50	25600	18285

Результаты синтеза представлены также на графике рис. 11.

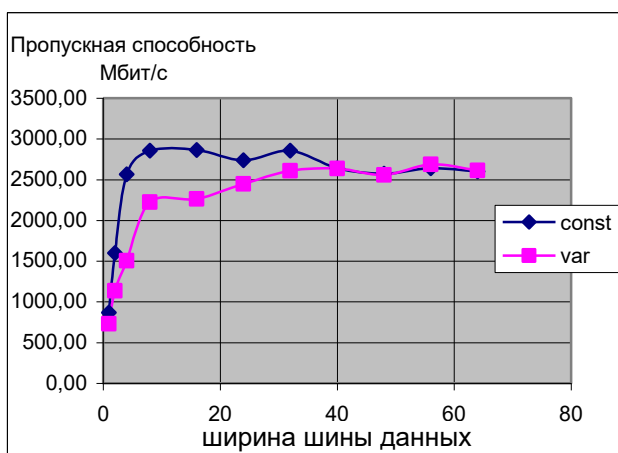
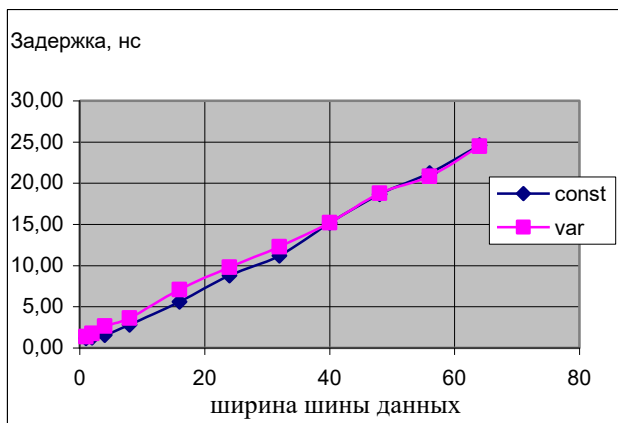
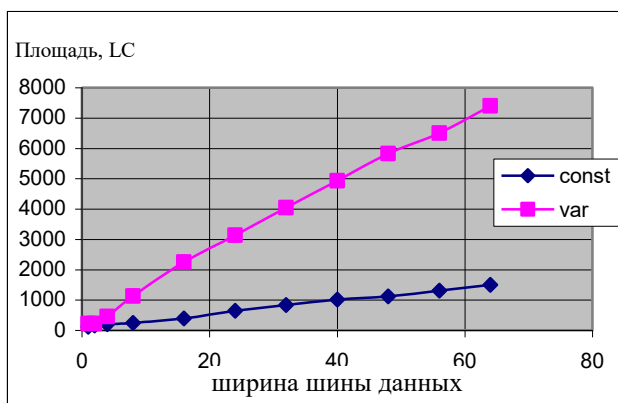


Рис. 10. Ресурсы ПЛИС, задержка на критическом пути и пропускная способность в зависимости от ширины шины данных кодеров (R=208) с постоянным многочленом и с переменным

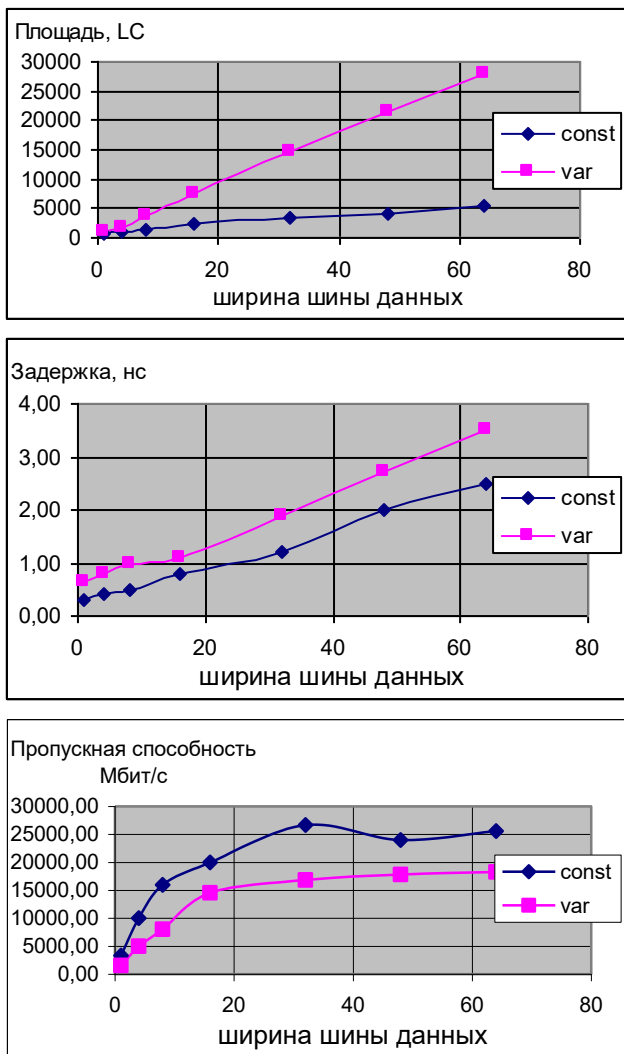


Рис. 11. Площадь (28 нм), задержка на критическом пути и пропускная способность в зависимости от ширины шины данных кодеров (R=208) с постоянным многочленом и с переменным

VI. ЗАКЛЮЧЕНИЕ

В данной статье предложен путь построения устройств кодирования для параллельной обработки данных, описан псевдокод для реализации как в аппаратуре, так и в программной модели. Предложен метод реконфигурирования корректирующей способности кода, который также применим для параллельной обработки любой шины данных.

Предложенный способ построения параллельного реконфигурируемого кодера, основанный на традиционной схеме кодера (рис. 2). Именно она фигурирует во многих источниках как базовая при объяснении кодов BCH [1, 5, 17], с ней же и приводится сравнение по быстродействию и площади. В статье подробно описан переход к реконфигурируемой и параллельной реализации кодера. В экспериментальной части показано, что реконфигурируемость не влияет на быстродействие,

однако обладает универсальностью. При этом именно параллельность приводит к выигрышу по быстродействию. Из таблиц 2-3 видно, что пропускная способность последовательного кодера достигает величины 3.3 Гб/с в СБИС, однако даже такую пропускную способность сложно обеспечить ввиду ограничений контактной площадки микросхемы. Однако использование предлагаемой параллельной (L=64) схемы кодера может обеспечить пропускную способность 25 Гб/с при тактовой частоте $25/64 = 390$ МГц, что соизмеримо с кодерами [6, 18]. Из рисунков 10-11 видно, что для любой ширины шины данных реконфигурируемость практически не влияет на быстродействие, но параллельность приводит к большей пропускной способности. Так, при увеличении ширины шины данных (то есть количества бит, обрабатываемых за такт) в L раз критический путь растет менее чем в L раз.

Применение реконфигурируемого кодера для работы с flash-памятью различных типов (более 5) в составе СнК делает его выгодным по занимаемой площади по сравнению с набором нескольких (то есть более 5) кодеров с постоянными многочленами без перенастройки корректирующей способности. Это видно из таблиц 2,3, количество ячеек кодера с переменным многочленом примерно в 5-6 раз больше количества ячеек кодера с постоянным многочленом.

ЛИТЕРАТУРА

- [1] Блейхут Р. Теория и практика кодов, контролируемых ошибок. Пер. с англ. М.: Мир, 1986, 576 с.
- [2] Yoo H., Lee Y, 7.3 Gb/s Universal BCH Encoder and Decoder for SSD Controller // IEEE 978-1-4799-2816-3. 2014. pp. 37-38.
- [3] Keshab K.Parhi, Eliminating the fanout bottleneck in parallel long BCH encoders // IEEE Communication Society. - 2004., -pp.2611-2615.
- [4] Zhang Jun, Wang Zhi-Gong, Hu Qing-Sheng, Xiao Jie, Optimized design for high-speed parallel BCH encoder // IEEE Int.Workshop VLSI Design&Video Tech.. -2005. -pp. 97-100.
- [5] Вернер М., Основы кодирования. М., «Техносфера», - 2004. -288с.
- [6] Gomes M., Falcao G., Silva V. Scalable and Parallel Codec Architectures for the DVB-S2 FEC System // IEEE 978-1-4244-2342-2/08, 2008. – P 1506-1509.
- [7] Патент US8812940 B2 – Programmable Error Correction Capability for BCH Codes
- [8] Патент CN102761340 Broadcast channel (BCH) parallel coding circuit.
- [9] Патент CN101227194 Circuit, encoder and method for encoding parallel BCH.
- [10] Патент CN102820892 Circuit for parallel BCH (broadcast channel) coding, encoder and method.
- [11] Aiswariya R., Parameshwaran R. Loop Unrolling for Second Order Recursive Digital Filter to Achieve High Throughput // Contemporary Engineering Sciences, vol.7.№8, 2014. – P. 357-362.
- [12] Zhang X., Parhi K.K. High-speed Architecture for Parallel Long BCH Encoder // GLSVLSI, 2004.- P.26-28.

- [13] Chuan-Sheng Lin, Kuang-Yuan Chen, Yu-Hsian Wang A NAND Flash Memory Controller for SD/MMC Flash Memory Card // IEEE 1-4244-0395, 2006. - P. 1284-1287.
- [14] Koorapati S., Prakash S. Design of Any Codeword Length Parallel Long BCH Encoder with the help of An Efficient C-Utility // International Conference on VLSI Systems, Architecture, Technology and Application , 2015.
- [15] Поперечный П.С. Разработка параллельного кодера БЧХ с регулируемой корректирующей способностью // Известия ЮФУ. Технические науки. -2015. -№7. -с.19-31.
- [16] Патент RU157943. Параллельный реконфигурируемый кодер БЧХ кодов. 21.07.2015.
- [17] Морелос-Сарагоса П., Искусство помехоустойчивого кодирования. Методы алгоритмы, применение. М., «Техносфера», 2005.
- [18] Lee Y., Yoo H., Jung J., Jo J., Park I. 17.7-Gb/s Iterative Concatenated-BCH Decoder in 65 nm CMOS for NAND Flash Memory // IEEE Journal of Solid-State Circuits, vol.48, №10, 2013.- P.2531-2540.

Adjustable error-correcting encoder for Systems on Chip

P.S. Poperechny

Postgraduate of Institute for Design Problems in Microelectronics RAS,

Engineer of Research and Development Center "ELVEES", ppoperechny@elvees.com

Keywords — Error-correction coding, encoder, linear feedback shift register (LFSR), SoC.

ABSTRACT

Nowadays a BCH (Bose–Chaudhuri– Hocquenghem) error-correcting coding is widely used in different technical fields, such as telecommunication, data storing systems etc. To use an error correcting codec in many these applications on a System-on-Chip (SoC) two main requirements should be met: a parallelization for data bus stream and universality for different error-correcting capability. Traditional serial encoder consists of linear feedback shift register (LFSR) and switches. To overcome many error-correcting coding capabilities one SoC should have a lot of encoders with different capabilities. So, this approach suppose very large chip square.

This article offers one universal decision for different requirements. Special registers store generating polynomial coefficients, which can be changed in working process. Also, the parallelization approach is offered too. This technique allows getting data from input bus. There are in-common mathematical equations for both software and hardware implementations for different data bus. The comparing results of logical hardware synthesis are presented. The proposed encoder has bigger throughput comparing with traditional approach. The universality decreases chip square consumption if usage of different error-correction capabilities is needed.

REFERENCES

- [1]] Blejht R. Theory and practice of error-controlling codes. M.: Mir, 1986, 576 p. (in Russian)
- [2] Yoo H., Lee Y, 7.3 Gb/s Universal BCH Encoder and Decoder for SSD Controller // IEEE 978-1-4799-2816-3. 2014. pp. 37-38.
- [3] Keshab K.Parhi, Eliminating the fanout bottleneck in parallel long BCH encoders // IEEE Communication Society. - 2004., -pp.2611-2615.
- [4] Zhang Jun, Wang Zhi-Gong, Hu Qing-Sheng, Xiao Jie, Optimized design for high-speed parallel BCH encoder // IEEE Int.Workshop VLSI Design&Video Tech.. -2005. -pp. 97-100.
- [5] Verner M., Coning basics. M., «Tehnosfera», -2004. -288p. (in Russian)
- [6] Gomes M., Falcao G., Silva V. Scalable and Parallel Codec Architectures for the DVB-S2 FEC System // IEEE 978-1-4244-2342-2/08, 2008. – P 1506-1509.
- [7] Patent US8812940 B2 – Programmable Error Correction Capability for BCH Codes
- [8] Patent CN102761340 Broadcast channel (BCH) parallel coding circuit.
- [9] Patent CN101227194 Circuit, encoder and method for encoding parallel BCH.
- [10] Patent CN102820892 Circuit for parallel BCH (broadcast channel) coding, encoder and method.
- [11] Aiswariya R., Parameshwaran R. Loop Unrolling for Second Order Recursive Digital Filter to Achieve High Throughput // Contemporary Engineering Sciences, vol.7.№8, 2014. – P. 357-362.
- [12] Zhang X., Parhi K.K. High-speed Architecture for Parallel Long BCH Encoder // GLSVLSI, 2004.- P.26-28.
- [13] Chuan-Sheng Lin, Kuang-Yuan Chen, Yu-Hsian Wang A NAND Flash Memory Controller for SD/MMC Flash Memory Card // IEEE 1-4244-0395, 2006. - P. 1284-1287.
- [14] Koorapati S., Prakash S. Design of Any Codeword Length Parallel Long BCH Encoder with the help of An Efficient C-Utility // International Conference on VLSI Systems, Architecture, Technology and Application , 2015.
- [15] Poperechny P.S. Design of parallel BCH encoder with tunable correcting level // Izvestija JuFU. Tehnicheskie nauki. -2015. -№7. -pp.19-31 (in Russian).
- [16] Patent RU157943. Parallel reconfigurable BCH encoder. 21.07.2015 (in Russian).
- [17] Morelos-Saragosa P., Error-correction cofing art. Methods, alghoritms, applications. M., «Tehnosfera», 2005. (in Russian)
- [18] Lee Y., Yoo H., Jung J., Jo J., Park I. 17.7-Gb/s Iterative Concatenated-BCH Decoder in 65 nm CMOS for NAND Flash Memory // IEEE Journal of Solid-State Circuits, vol.48, №10, 2013.- P.2531-2540.