

УДК 004.274:621.391

**В. А. КУЛАНОВ, А. Е. ПЕРЕПЕЛИЦЫН**

*Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», Харьков, Украина*

## **ПРОЕКТИРОВАНИЕ ГИБКИХ АРХИТЕКТУР ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ НА ПЛИС НА ПРИМЕРЕ РЕАЛИЗАЦИИ КОДОВ РИДА-СОЛОМОНА**

*Рассматриваются особенности реализации гибких архитектур систем на ПЛИС за счет механизма параметризации. Предлагается классификация параметризации проектов на ПЛИС (FPGA) по критерию обеспечения их гибкости в процессе работы системы. Даются рекомендации по использованию предложенных способов обеспечения гибкости вычислительной системы в зависимости от требований. Приводятся результаты анализа возможностей параметризации кодера и декодера кодов Рида-Соломона. Предлагается сопоставление количества ресурсов ПЛИС для различных значений параметров разработанного кодера с указанием значимости отдельных параметров.*

**Ключевые слова:** ПЛИС, FPGA, параметризация, гибкая архитектура, вычислительные системы, коды Рида-Соломона.

### **Введение**

Специализированные вычислительные системы, реализованные на основе технологии ПЛИС, широко применяются для решения разнообразных задач. Зачастую системы с идентичной функциональностью могут отличаться лишь некоторыми особенностями, при том что их разработка весьма ресурсоёмка. В таких случаях могут быть использованы параметризованные IP (Intellectual Property) блоки, универсальность которых обеспечивается внесением функционально значимых констант в список параметров, значения которым могут быть индивидуально заданы для каждого отдельного проектного решения. Эти параметры могут отвечать как за разрядность используемых шин данных, так и за особенность реализации архитектуры вычислительной системы в целом [1].

Описанный способ параметризации предусматривает определение требуемых значений параметров на этапе компиляции, что наиболее приемлемо в случае полного соответствия функциональности IP-ядра условиям решаемой задачи. Однако зачастую существует необходимость реализации аппаратных блоков, функциональность которых частично или полностью зависит от значений сигналов, поступающих на управляющие входы непосредственно во время работы устройства. Так, например, счётчик может предусматривать стартовое значение счёта, загружаемое во время его работы.

Реализация отдельных видов вычислительных систем с возможностью настройки или адаптации их к некоторым внешним условиям в процессе их функционирования является нетривиальной задачей. К их числу можно отнести и аппаратную реализацию ко-

дера и декодера кодов Рида-Соломона [2, 3].

Целью данной статьи является исследование особенностей разработки гибких архитектур вычислительных систем на ПЛИС на примере кодера и декодера кодов Рида-Соломона.

Для достижения поставленной цели необходимо рассмотреть особенности реализации гибких архитектур вычислительных систем на ПЛИС с использованием механизма параметризации проектов, проанализировать возможности параметризации кодера/декодера РС-кодов, а также разработать их и провести оценку полученных параметризуемых решений.

### **1. Реализация гибких архитектур вычислительных систем на ПЛИС**

Построение аппаратных решений средствами ПЛИС (далее подразумевается FPGA) обеспечивает гибкость реализации ввиду возможности перепрограммирования системы даже на этапе ее ввода в эксплуатацию [4].

Гибкая архитектура вычислительной системы на ПЛИС предполагает возможность изменения ее характеристик или функциональности (частичной или полной) в зависимости от установленных ограничений (параметров).

Особенности языков описания аппаратуры и специфика реализации в FPGA позволяют рассматривать гибкость проектов на двух уровнях:

- статическая параметризация;
- динамическая параметризация.

В первом случае гибкость обеспечивается заданием константных значений параметрам проекта,

предусмотренным на этапе его разработки. Такая параметризация позволяет создавать универсальные модули, структура которых определяется лишь на этапе компиляции/синтеза [1]. Количество требуемых ресурсов ПЛИС зависит от значений параметров, а аппаратная реализация, по сути, является частным случаем, эквивалентным проекту без параметризации. Трудоемкость процесса разработки для типовых параметризуемых свойств проекта сравнима с величиной, присущей для проекта без параметризации.

Во втором случае гибкость подразумевает возможность настройки (параметризации) как всей вычислительной системы, так и отдельных ее частей непосредственно в процессе функционирования (работы) – динамическая параметризация. При этом конфигурирование таких проектов осуществляется путем введения дополнительных сигналов управления и управляющих регистров.

Обеспечение возможности настройки систем на ПЛИС с помощью механизма динамической параметризации требует больших затрат людских и материальных ресурсов на разработку, т.к. в этом случае необходимо реализовать весь набор предписанных свойств объекта проектирования и управляющих сигналов для их выбора. При этом количество требуемых аппаратных ресурсов ПЛИС превосходит аналогичную величину для случая ненастраиваемой функциональности, т.е. такая гибкость предполагает избыточность в реализации.

Совместное использование механизмов статической и динамической параметризации позволяет достичь большей гибкости проектов на ПЛИС, нежели каждого из них в отдельности. Так с помощью статических параметров может быть организовано управление количеством используемых ресурсов посредством отсечения избыточных динамических свойств, ненужных для конкретной реализации проекта.

## 2. Возможности обеспечения гибкости в реализации РС-кодов на ПЛИС

К основным достоинствам реализации кодов Рида-Соломона на ПЛИС следует отнести распараллеливание вычислений за счет возможности организации в единую структуру набора универсальных блоков кодирования/декодирования. При этом параллельная реализация предполагает, что создаваемая архитектура, включающая набор параллельных частей системы, в достаточной степени определена, что соответствует решению четко детерминированной задачи [5]. Отметим, что специфика кодов Рида-Соломона позволяет получить детерминированную архитектуру при параллельном способе решения задачи кодирования/декодирования лишь для фиксированных значений параметров кода  $m$ ,  $N$ ,  $K$ .

В рамках РС-кодов значение параметра  $m$  определяет разрядность символа в сообщении;  $N$  – количество символов в сообщении (длина кодового сообщения), где  $N = 2^m - 1$ ;  $K$  – количество символов данных (длина «полезных» данных в кодовом сообщении) [2, 3]. Параметр  $2t$  определяет количество контрольных символов. На рисунке 1 представлена общая структура кодового сообщения с изменяемыми параметрами кода Рида-Соломона ( $m$ ,  $N$ ,  $K$ ) для проектируемой гибкой архитектуры кодера/декодера.

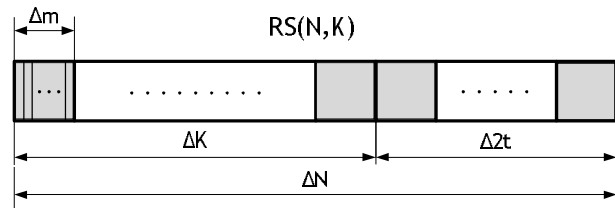


Рис. 1. Структура кодового сообщения с изменяемыми параметрами РС-кода

Отметим, что даже при фиксированном значении параметра  $m$ , существует возможность задания вариативной длины кодируемого сообщения, определяемого величиной  $K$ . При этом если разрабатываемый блок кодирования должен работать не только с кодом РС (255, 223), т.е. с количеством избыточных символов, отличным от 32, то число независимо реализованных умножителей в поле Галуа должно быть не меньшим, чем это значение.

Таким образом, при разработке параметризуемого РС-кодека, предполагающего работу с фиксированными значениями параметров кода Рида-Соломона, достаточно использовать статическую параметризацию [6]. Если же существует необходимость обеспечения гибкости модуля непосредственно во время работы, то это может быть достигнуто дополнительно с помощью механизма динамической параметризации.

## 3. Реализация гибкой архитектуры РС-кодека на ПЛИС

В общем виде в состав РС-кодека с гибкой архитектурой входит кодер и декодер. Очевидно, что при их использовании в составе одной приемопередающей подсистемы параметры кода Рида-Соломона должны совпадать как на передающей, так и на принимаемой стороне.

Отметим, что использование кодера и декодера с гибкой архитектурой возможно отдельно друг от друга как независимых проектных решений в составе других систем.

**Кодер кода Рида-Соломона с гибкой архитектурой.** В общем виде архитектура кодера

РС-кодов для фиксированного набора параметров является тривиальным решением и может быть представлена в виде некоторого набора однотипных вычислительных элементов (сумматоров и умножителей в поле Галуа) и блоков памяти, объединенных в единую цепочку (структуру) [7]. Каждый из элементов в этой цепочке может быть настроен на два режима работы: режим вычисления «первичного элемента» и стандартный режим вычисления.

Реализация гибкой архитектуры предполагает внесение избыточности не только на уровне дополнительных сигналов управления и настроечных регистров, которые определяют текущий режим работы кодера, а также и дополнительных вычислительных ячеек (ресурсов). При этом количество этих ячеек определяет как максимально допустимое значение параметров РС-кода, так и допустимые границы этих параметров.

Особенность формирования структуры кодера с гибкой архитектурой для параметров кода Рида-Соломона РС (15, K) представлено на рисунке 2. Здесь, например, для фиксированного значения  $m = 4$  параметр K может находиться в диапазоне от 1 до 13. При этом в зависимости от значения K будет меняться позиция ячейки, которая работает в режиме «первичного элемента».



Рис. 2. Формирование структуры кодера с гибкой архитектурой для параметра кода РС (15, 11)

Таким образом, структура параметризуемого кодера РС-кода при фиксированном значении  $m = 4$  и «плавающего» значения параметра K включает в себя 15 вычислительных ячеек.

При этом в случае статической параметризации, когда параметры кода определяются на этапе компиляции проекта, количество вычислительных ячеек для параметра РС (15, 11) составляет 4.

Обеспечение возможности изменения длины кодового слова с поддержкой механизма динамической параметризации посредством задания требуемого значения  $m$  существенно усложняет архитектуру вычислительной ячейки. В этом случае должны быть реализованы гибкие умножители в поле Галуа, допускающие динамическое указание разных значений  $m$ .

**Умножитель в поле Галуа с гибкой архитектурой.** Реализация операции умножения в поле Га-

луа для фиксированных значений  $m$  может быть выполнена как в последовательном, так и в параллельном стиле описания [8]. Структура модуля для разных  $m$  имеет отличия, допускающие универсальное описание средствами статической параметризации. К их числу можно отнести потребность в различных неприводимых полиномах. Однако эти отличия существенно затрудняют реализацию общей архитектуры для разных значений  $m$ .

Одним из возможных способов организации динамической параметризации является мультиплексирование независимо реализованных архитектур для разных значений  $m$ , которые могут находиться в диапазоне от 2 до 16. При этом средствами статической параметризации может быть предоставлен выбор конкретных значений из указанного диапазона, задание которых должен поддерживать умножитель уже во время работы.

Таким образом, совместное использование статической и динамической параметризации может позволить снизить количество аппаратных ресурсов, требуемых для реализации универсального модуля, за счёт отсеечения реализации неиспользуемых режимов.

**Декодер кода Рида-Соломона с гибкой архитектурой.** Процесс декодирования РС-кодов включает в себя несколько шагов, на каждом из которых существует потребность в умножителях и сумматорах в поле Галуа. Для сообщений различной длины количество последовательных операций с зависимостью по данным будет различно. Специфика алгоритмов декодирования предполагает вариативность даже для фиксированных параметров кода, что обусловлено зависимостью набора действий от закодированного сообщения. Если в поступившем сообщении нет ошибок, то нет необходимости выполнять операции по их исправлению.

Учёт изложенных особенностей обуславливает возможность использования для реализации декодера с гибкой архитектурой конечного автомата, состояния которого отвечают за реализацию отдельных частей алгоритмов. В этом случае, как и для кодера, количество умножителей с гибкой архитектурой определяется максимально возможной величиной  $2t$ . Кроме того, алгоритмы декодирования требуют наличия ещё одного дополнительного умножителя. Данные между итерациями могут быть сохранены в промежуточных регистрах, разрядность которых определяется максимально допустимой величиной  $m$ .

Описанная реализация декодера также требует вычисления значений логарифма в поле Галуа, что может быть реализовано таблично для каждого значения  $m$  и коммутироваться аналогично умножителю.

#### 4. Результаты реализации РС-кодека с гибкой архитектурой на ПЛИС

Результаты имплементации кодека кодов Рида-Соломона с гибкой архитектурой – количество используемых ресурсов ПЛИС для разных значений поддерживаемых параметров приведены в таблице 1. В качестве оцениваемых ресурсов рассматривается количество занимаемых логических элементов (Logic Element) ПЛИС класса FPGA компании Altera Cyclone II (EP2C35F672C6).

Следует отметить, что для полученных проектных решений количество занимаемых ресурсов для вариантов 1 и 4, а также вариантов 2, 3 и 5 существенно не отличается (табл. 1). Это объясняется особенностью реализации архитектуры кодера и декодера. Здесь параметр  $m$  оказывает меньшее влияние на количество задействованных ресурсов, чем параметр  $2t$  (является одинаковым для рассматриваемых вариантов).

Таким образом, наиболее критичным по отношению к количеству используемых ресурсов ПЛИС является параметр  $2t$  (поддерживаемое количество проверочных символов в сообщении), отвечающий также за число реализуемых гибких умножителей в поле Галуа.

#### Выводы

В статье рассмотрены особенности разработки гибких архитектур вычислительных систем на ПЛИС на примере кодера/декодера РС-кодов за счет механизма параметризации проектов. Предложено рассматривать параметризацию на двух уровнях: статическая и динамическая. Основным отличием между ними является возможность адаптации различных проектных решений в зависимости от устанавливаемых ограничений без необходимости перекомпиляции и повторной «прошивки» кристалла ПЛИС (динамическая параметризация). При этом, вносимая структурная избыточность может оказывать значительное влияние на количество занимаемых проектом ресурсов ПЛИС. Проведен анализ возможностей параметризации кодера/декодера РС-кодов. Установлено, что при разработке параметризуемого кодека, предполагающего работу с фиксированными значениями параметров кода Рида-Соломона, достаточно использовать статическую параметризацию. Если же существует необходимость обеспечения гибкости модуля непосредственно в процессе его работы, то это может быть обеспечено динамической параметризацией.

Таблица 1  
Занимаемые ресурсы ПЛИС Altera Cyclone II (EP2C35F672C6) для разных значений параметров кодека РС-кодов

№ Варианта	Параметры РС-кодека			Используемые ресурсы ПЛИС, Altera Logic Elements (EP2C35F672C6 – 33216 LEs)	
	Поддерживаемое количество битов в символе сообщения ( $m$ )	Поддерживаемое количество проверочных символов в сообщении ( $2t$ )	Поддерживаемые параметры РС-кода $PC(N, K)$	Количество LEs	%, LEs
1	4	14	$PC(15, K), K \in [1; 13]$	14375	$\approx 43\%$
2	8	16	$PC(255, K), K \in [239; 253]$	18816	$\approx 57\%$
3	10	16	$PC(1023, K), K \in [1007; 1021]$	19608	$\approx 59\%$
4	4, 8	14	$PC(15, K), K \in [1; 13]$ $PC(255, K), K \in [241; 253]$	16356	$\approx 49\%$
5	8, 10	16	$PC(255, K), K \in [239; 253]$ $PC(1023, K), K \in [1007; 1021]$	21117	$\approx 64\%$
6	4, 8, 10	6	$PC(15, K), K \in [9; 13]$ $PC(255, K), K \in [249; 253]$ $PC(1023, K), K \in [1017; 1021]$	7955	$\approx 25\%$
		16	$PC(15, K), K \in [1; 13]$ $PC(255, K), K \in [239; 253]$ $PC(1023, K), K \in [1007; 1021]$	21584	$\approx 65\%$
		22	$PC(15, K), K \in [1; 13]$ $PC(255, K), K \in [233; 253]$ $PC(1023, K), K \in [1001; 1021]$	30398	$\approx 92\%$

Разработана гибкая архитектура РС-кодека. Установлено, что в значительной степени на количество занимаемых ресурсов ПЛИС оказывает влияние параметр  $2t$ , который определяет количество проверочных символов в сообщении.

### Литература

1. Kulanov, V. *Parameterized IP Infrastructures for Fault-Tolerant FPGA-Based Systems: Development, Assessment, Case-Study [Text]* / V. Kulanov, V. Kharченко, A. Perepelitsyn // *Proceedings of IEEE East-West Design & Test Symposium*. – 2009. – P. 322–325.
2. Семенов, Ю. А. *Алгоритмы телекоммуникационных сетей. Ч.1: Алгоритмы и протоколы каналов и сетей передачи данных [Текст] : учеб. пособие.* / Ю. А. Семенов. – М. : Интернет – Университет Информационных Технологий : БИНОМ. Лаборатория знаний, 2007. – 637 с.
3. Золотарев, В. В. *Помехоустойчивое кодирование. Методы и алгоритмы: справочник [Текст]* / В. В. Золотарев, Г. В. Овечкин // под ред. чл.-кор. РАН Ю. Б. Зубарева. – М. : Горячая линия-Телеком, 2004. – 126 с.

4. Перепелицын, А. Е. *Анализ применения ПЛИС технологий в медицинском оборудовании [Текст]* / А. Е. Перепелицын, В. С. Харченко // *Радиоэлектронні і комп'ютерні системи*. – 2012. – № 7 (59). – С. 125–130.

5. Elharoussi, M. *VHDL Design and FPGA Implementation of a Parallel Reed-Solomon (15, K, D) Encoder/Decoder [Text]* / M. Elharoussi, A. Hamyani, M. Belkasmı // *IJACSA*. – 2013. – Vol. 4, No.1. – P. 33-37.

6. Тайлеб-Мазуз, Н. *Параметризованный проект высокоскоростного многоканального декодера кода Рида-Соломона [Текст]* / Н. Тайлеб-Мазуз // *Вестник МЭИ*. – М. : Издательский дом МЭИ, 2011. № 5. – С. 112-118.

7. Касперски, К. *Мощность кодов Рида-Соломона или информация, воскресшая из пепла [Текст]* / К. Касперски // *Системный администратор*. – 2003. – № 8(9). – С. 88-94.

8. Лепеха, В. Л. *Реализация декодера Рида-Соломона в ПЛИС [Текст]* / В. Л. Лепеха, Т. М. Лесик, Р. А. Франко // *Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка : Зб. наук. пр.* – К. : Век+, 2009. – № 50.- С. 78-82.

Поступила в редакцию 17.03.2014, рассмотрена на редколлегии 24.03.2014

**Рецензент:** д-р техн. наук, вед. науч. сотр. Г. А. Кучук, Харьковский университет Воздушных Сил им. И. Кожедуба, Харьков, Украина.

## ПРОЕКТУВАННЯ ГНУЧКИХ АРХІТЕКТУР ОБЧИСЛЮВАЛЬНИХ СИСТЕМ НА ПЛІС НА ПРИКЛАДІ РЕАЛІЗАЦІЇ КОДІВ РІДА-СОЛОМОНА

*В. О. Куланов, А. Є. Перепелицын*

Розглядаються особливості реалізації гнучких архітектур систем на ПЛІС за рахунок механізму параметризації. Пропонується класифікація параметризації проектів на ПЛІС (FPGA) за критерієм забезпечення гнучкості під час функціонування системи. Надаються рекомендації щодо використання кожного із розглянутих способів забезпечення гнучкості обчислювальної системи в залежності від зовнішніх вимог. Наводяться результати аналізу можливостей параметризації кодеру і декодеру кодів Ріда-Соломона. Пропонується порівняння кількості ресурсів ПЛІС для різних значень параметрів розробленого кодеку із зазначенням значимості окремих параметрів.

**Ключові слова:** ПЛІС, FPGA, параметризація, гнучка архітектура, обчислювальні системи, коди Ріда-Соломона.

## FLEXIBLE FPGA COMPUTER SYSTEMS DESIGN OF REED-SOLOMON CODES

*V. O. Kulanov, A. E. Perepelitsyn*

The implementation of flexible FPGA-based architectures through the mechanism of parameterization is observed. Classification of FPGA-based projects parameterization by providing them flexibility during the operating mode is proposed. Recommendations of proposed methods that provide flexibility of a computer system according to some requirements specification are given. The results of the analysis of Reed-Solomon encoder and decoder parameterization are described. A comparison between different numbers of occupied FPGA resources for various projects with different set of RS-code parameters is considered.

**Key words:** PLD, FPGA, parameterization, flexible architecture, computing systems, Reed-Solomon code.

**Куланов Віталій Александрович** – канд. техн. наук, доцент кафедри комп'ютерних систем і мереж Національного аерокосмічного університету ім. Н. Е. Жуковського «ХАІ», Харьков, Украина, e-mail: v.kulanov@csn.khai.edu.

**Перепелицын Артём Евгеньевич** – аспирант, ассистент кафедры комп'ютерних систем і мереж Національного аерокосмічного університету ім. Н. Е. Жуковського «ХАІ», Харьков, Украина, e-mail: a.perepelitsyn@csn.khai.edu.