

РЕАЛИЗАЦИЯ РАДИАЦИОННО-СТОЙКОГО КОДИРОВАНИЯ В РАМКАХ МЕЖКРИСТАЛЬНОЙ СВЯЗИ СИСТЕМ, СОСТОЯЩИХ ИЗ НЕСКОЛЬКИХ ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ СХЕМ

© 2018 г. Никитин А.А.

Ракетно-космическая корпорация «Энергия» имени С.П. Королёва (РКК «Энергия»)
Ул. Ленина, 4А, г. Королёв, Московская обл., Российская Федерация, 141070, e-mail: post@rsce.ru

В работе представлен новый подход к разработке многокристальных систем на базе программируемых логических интегральных схем (ПЛИС), реализующих объемные алгоритмы управления космическими аппаратами. При этом подходе учитываются требования по стойкости к специальным факторам и специфика реализуемых алгоритмов. Описан принцип разделения единого алгоритма на функциональные узлы и принципы их взаимодействия. Предложена методика защиты межкристальных связей с помощью кодов, восстанавливающих ошибки, с учетом вероятности возникновения ошибок в кодере. Обобщены результаты тестирования математической модели метода радиационно-стойкого кодирования на базе сверточного кода, а также испытания имплементации разработанной схемы в кристаллы ПЛИС. Сделаны выводы о перспективах применения данного метода с учетом расширения номенклатуры доступных к применению ПЛИС российского производства.

Ключевые слова: сверточный код, бортовая система управления, межкристальная связь, программируемая логическая интегральная схема, разработка, испытания.

IMPLEMENTATION OF RADIATION-RESISTANT ENCODING WITHIN THE FRAMEWORK OF CHIP-TO-CHIP INTERCONNECTIONS IN SYSTEMS CONSISTING OF SEVERAL FIELD-PROGRAMMABLE GATE ARRAYS

Nikitin A.A.

S.P. Korolev Rocket and Space Public Corporation Energia (RSC Energia)
4A Lenin str., Korolev, Moscow region, 141070, Russian Federation, e-mail: post@rsce.ru

The paper discusses a new approach to development of multichip systems based on Field-Programmable Gate Arrays (FPGA) implementing volume algorithms for spacecraft control. This approach takes into account requirements for resistance to special factors and specifics of the implemented algorithms. It describes the principle of dividing a single algorithm into functional units and principles of their interfacing. A technique is proposed for protecting chip-to-chip interconnections using error correcting codes taking into account the probability of errors occurring in the encoder. It summarizes the results of testing the math model of a radiation-resistant encoding method based on convolutional code, as well as tests of the implementation of the developed circuit in FPGA chips. It draws conclusions about future prospects for application of this technique taking into account the expanding range of available Russian-made FPGAs.

Key words: convolution code, onboard control system, chip interconnection, field-programmable gate array, development, testing.



НИКИТИН А.А.

НИКИТИН Андрей Александрович — инженер-электроник 2 категории РКК «Энергия», e-mail: andrey.nikitin23@rsce.ru
 NIKITIN Andrey Aleksandrovich — Electronics engineer 2 category at RSC Energia, e-mail: andrey.nikitin23@rsce.ru

Введение

Для реализации алгоритмов управления космическими аппаратами (КА) универсальным решением являются программируемые логические интегральные схемы (ПЛИС) [1, 2].

Круг задач, возлагаемых на системы управления КА, с каждым годом расширяется. Реализуемые алгоритмы на сегодняшний день должны не только выполнять свои задачи, но и осуществлять защиту своей физической реализации от радиационных сбоев, а также иметь средства самоконтроля. Поэтому для реализации таких сложных систем требуются ПЛИС, имеющие большой запас логической емкости и большое количество портов ввода-вывода.

У разработчиков возникают трудности в реализации требуемых алгоритмов из-за ограниченного выбора производителей компонентной базы в условиях импортозамещения, так как ПЛИС российского производства имеют скромный объем логических элементов [3, 4] и крайне ограниченный перечень доступных корпусов.

Для этой цели нужен новый подход в проектировании алгоритмически сложных систем на базе ПЛИС, который должен учитывать требования к аппаратуре, специфику ее применения и ограничения по компонентной базе.

Архитектура многокристальной системы и многопоточной межкристальной связи

Одним из вариантов решения сформулированной проблемы является разделение алгоритма управления на несколько ПЛИС, однако оно возможно только с учетом их функциональных особенностей. Наиболее распространенной задачей для систем управления КА является реализация многоканальных алгоритмов управления. Оптимальное решение задачи разделения функционала системы представлено на рис. 1. Данная архитектура подразумевает наличие базовой и N локальных ПЛИС. Базовая ПЛИС реализует алгоритмы, общие для всех объектов управления, а также интерфейсную часть. Каждая локальная ПЛИС на основе информации от базовой ПЛИС и сигналов обратной связи реализует индивидуальные алгоритмы для каждого объекта управления.

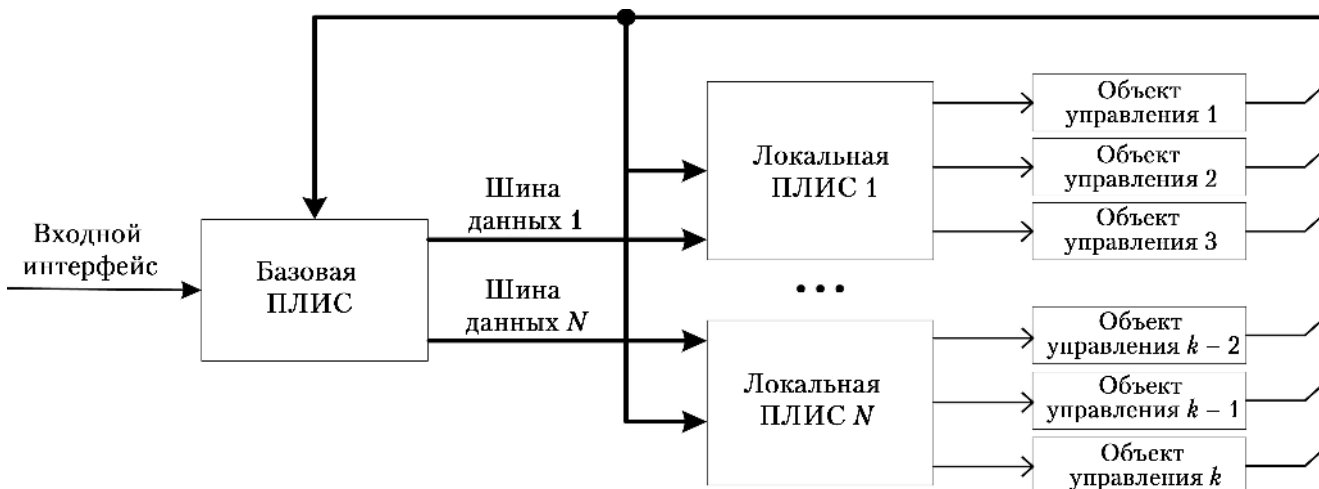


Рис. 1. Структурная схема многопоточной архитектуры межкристальной связи

Для передачи исходного сообщения длиной k_0 символов от базовой ПЛИС к локальным может быть использован или любой из интерфейсов, или параллельная шина. Большая часть реализуемых на ПЛИС интерфейсов обладает временной задержкой, не позволяющей обмениваться информацией на едином функциональном уровне, а параллельная шина позволяет элементам системы функционировать на одной частоте с задержкой, равной времени распространения сигнала между ними. Такой подход также дает возможность разрабатывать все ПЛИС как единое функциональное устройство, в котором базовые и локальные ПЛИС являются законченными IP-блоками, позволяющими реализовывать систему как на базе $N+1$ ПЛИС, так и в одном кристалле. Данный подход позволяет при расширении номенклатуры элементной базы российского производства без изменений интегрировать всю систему в новую ПЛИС.

Одним из условий реализации подобной схемы является защита каждой линии шин данных посредством реализации кодирования, восстанавливающего ошибки [5], так как именно на основе информации, передаваемой по ним, будет формироваться управляющий сигнал для объектов управления.

Выбор метода защиты каналов, а именно, типа кодов, восстанавливающих ошибки, возлагается на разработчика системы и зависит от множества факторов, в частности, от методов кодирования, заложенных в других микросхемах, наработок в данной области команды разработчиков, специфики передаваемых данных и многих других. В данной работе будет рассмотрена защита с помощью сверточного кодирования, так как оно является наиболее подходящим для блочного кода [6].

Базовым элементом сверточных кодера и декодера является структура, хранящая значения одного символа m из k_0 символов входного сообщения в течение n_0 тактов. В отличие от классической реализации кодера на базе сдвигового регистра, в данной работе рассматривается реализация на циклически перезаписываемом регистре, представленном на рис. 2. Основная идея в том, что на каждом следующем такте запись производится в новый бит, а при переполнении регистра происходит перезапись

наиболее старых записей [7]. При этом циклически перезаписываемый регистр $H(m)$ для $n_0 = 3$ во время тактов t_n будет содержать следующие значения:

$$t_1: H(m_i) = (m_i, m_{i-2}, m_{i-1}) \text{ — сбоя нет};$$

$t_2: H(m_{i+1}) = (\bar{m}_i, m_{i+1}, m_{i-1})$ — сбой в первом бите \bar{m}_i ;

$t_3: H(m_{i+2}) = (m_i, m_{i+1}, m_{i+2})$ — перезапись бита не производилась.

Кодовые слова $G(H(m))$ из данных значений регистра будут следующими:

$$t_1: G(H(m_i)) = (\oplus_2(m_i, m_{i-2}), \oplus_2(m_i, m_{i-1}), \oplus_2(m_{i-1}, m_{i-2}));$$

$$t_2: G(H(m_{i+1})) = (\oplus_2(\bar{m}_i, m_{i+1}), \oplus_2(\bar{m}_i, m_{i-1}), \oplus_2(m_{i+1}, m_{i-1}));$$

$$t_3: G(H(m_{i+2})) = (\oplus_2(m_i, m_{i+1}), \oplus_2(m_i, m_{i+2}), \oplus_2(m_{i+1}, m_{i+2})).$$

Как видно, в третьем такте за счет отсутствия перезаписи может быть восстановлен любой из сбоев, вызванный следующими эффектами [8, 9]:

- SET (Single Event Transients) — переходная ионизационная реакция;
- SEU (Single Event Upsets) — однократный восстанавливаемый сбой.

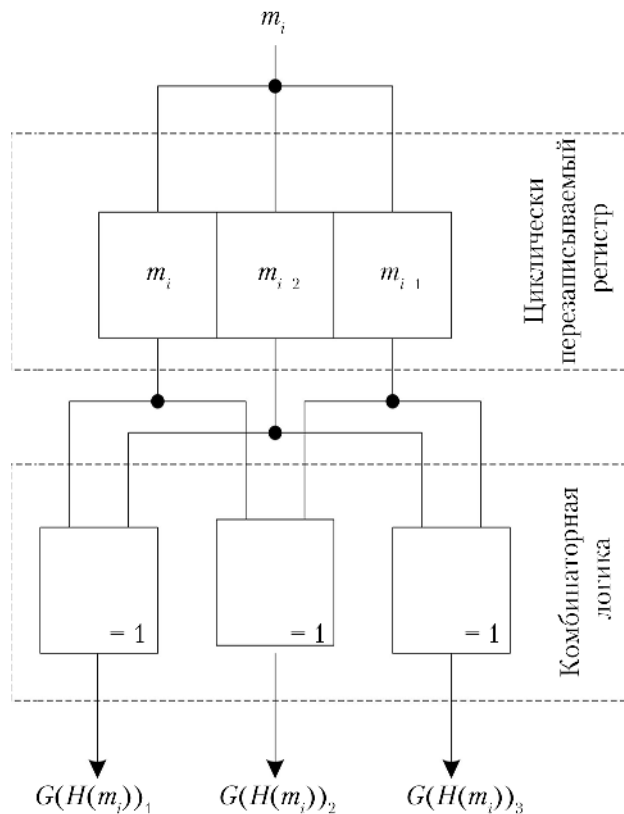


Рис. 2. Структурная схема линии связи с применением сверточного кодирования

Для реализации циклически перезаписываемого регистра необходимо реализовать счетчик, представленный на рис. 3, для выдачи адреса $A = (a_1, \dots, a_T)$, несущего номер модифицируемого разряда регистра $H(m)$.

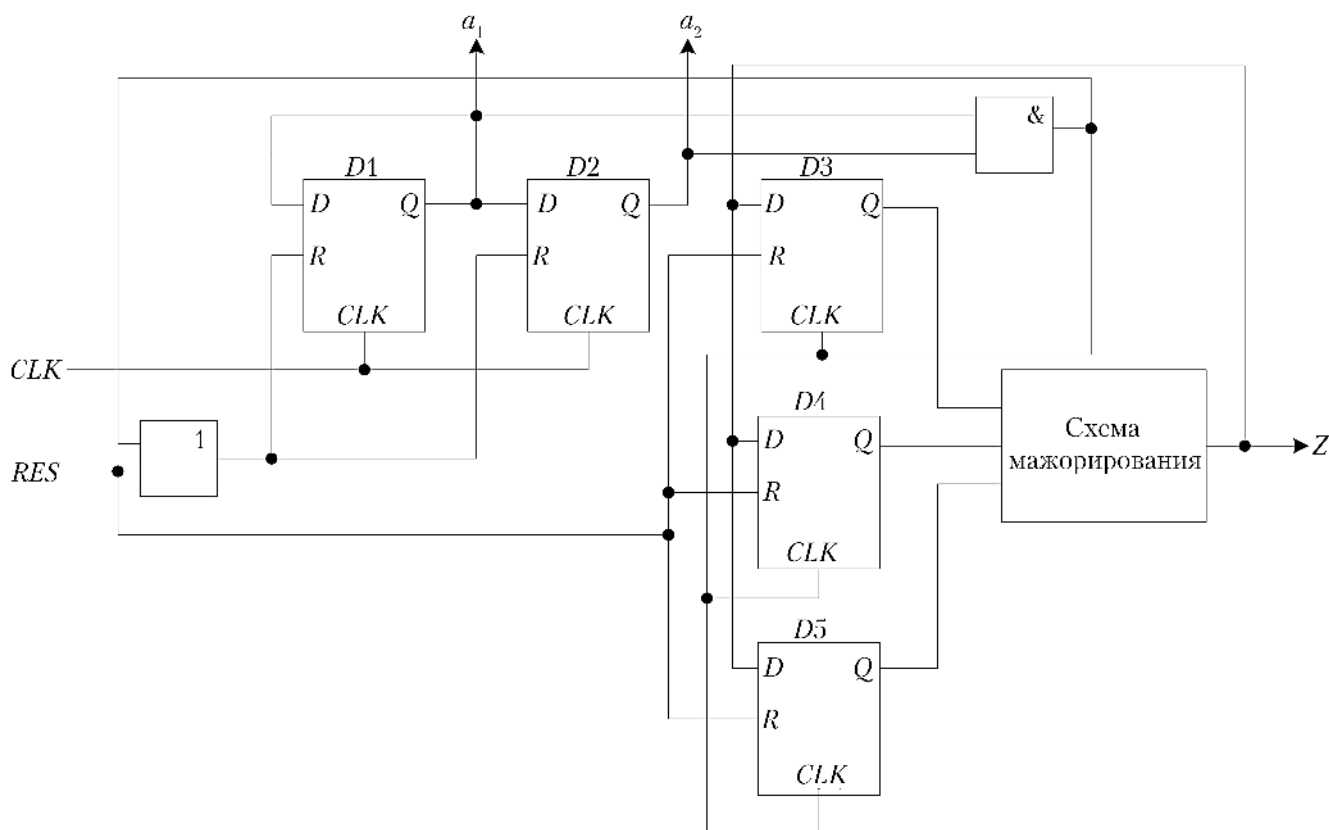


Рис. 3. Схема реализации счетчика адреса

Счетчик будет реализовываться на регистре длиной T разрядов при условии, что $T = \log_2 n_0$, и значение T округлено до целого значения в большую сторону. В частности, для $n_0 = 3$ необходимо реализовать счетчик, представленный на рис. 3, на двух триггерах $D1$ и $D2$, а для ограничения счета адресами 0, 1 и 2 — дополнительный сигнал сброса при $a_1 = 1$ и $a_2 = 1$ с помощью элемента конъюнкции, и проинформировать сброс счетчика или по данному сигналу, или по сигналу внешнего синхронного сброса.

На основе данной схемы был составлен перечень всех возможных критических сбоев, представленный в табл. 1.

Исходя из анализа данных табл. 1, было предложено не реализовывать схему защиты счетчика, а ввести дополнительный избыточный символ, формируемый

по детерминированному алгоритму, и кодировать его аналогично символам исходного сообщения. Данный подход позволит на этапе декодирования восстанавливать значение адреса с учетом сбоя и на его основе проводить декодирование символов исходного сообщения.

Одним из вариантов формирования избыточного символа является схема, представленная на рис. 3, состоящая из триггеров $D3–D5$ и схемы мажорирования. По своей сути она является одноразрядным счетчиком, защищенным от сбоев троированием, для которого тактовым сигналом является сигнал сброса счетчика адреса. Схема защиты является обязательным условием, так как восстановление адреса декодером возможно только при условии достоверности избыточного символа.

Таблица 1

Вероятные сбои в счетчике адреса

Начальное значение	01	10	00
Без сбоя	$a_{i0} = 01 \rightarrow a_{i0+1} = 10$	$a_{i0} = 10 \rightarrow a_{i0+1} = 00$	$a_{i0} = 00 \rightarrow a_{i0+1} = 01$
Сбой в a_1	$a_{i0} = 01 \rightarrow a_{i0+1} = 01$	$a_{i0} = 10 \rightarrow a_{i0+1} = 01$	$a_{i0} = 00 \rightarrow a_{i0+1} = 10$
Сбой в a_2	$a_{i0} = 01 \rightarrow a_{i0+1} = 01$	$a_{i0} = 10 \rightarrow a_{i0+1} = 01$	$a_{i0} = 00 \rightarrow a_{i0+1} = 00$

В табл. 2 представлен полный перечень возможных кодовых слов на основе избыточного символа $Z = 1$ под действием внешних сбоев. Если расписать также все состояния для исходного избыточного символа $Z = 0$, то станет видно, что сбои в счетчике адреса и соответствующие им выходные кодовые слова полностью совпадают с расписанными в табл. 2, однако состояние циклически перезаписываемого регистра будет инвертированным.

При декодировании последовательности кодовых слов, сформированных на основе избыточного символа Z в условиях радиационного защелкивания (эффект *SEL – Single Event Latchup*) бита циклически перезаписываемого регистра, они могут совпасть со словами, сформированными при сбое схемы счетчика адреса, и решение будет неоднозначным, поэтому была реализована схема кодирования, представленная на рис. 4.

Данная схема на каждом такте записывает с помощью дешифратора значение избыточного символа Z параллельно в оба циклически перезаписываемых регистра по адресу a . Далее дешифратор формирует на входе комбинаторной логики значение одного из регистров, в зависимости от сигнала C меняющего свое состояние с частотой, равной половине тактовой частоты. Таким образом, исключается возможность формирования двух последовательных недостоверных кодовых слов G в результате радиационного защелкивания.

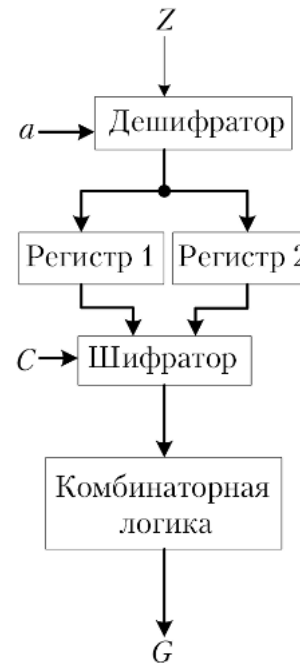


Рис. 4. Схема кодирования избыточного символа Z

Суммируя все доработки, радиационно-стойкий кодер для межкристальной связи ПЛИС выполняется в соответствии со схемой базовой ПЛИС, представленной на рис. 5.

Как видно из схемы, представленной на рис. 5, сообщение, сформированное универсальным алгоритмом управления, разбивается на k_0 символов. К исходному сообщению добавляется избыточный символ Z , формируемый на основе генерируемого на каждом такте адреса A . Все k_{0+1} символов записываются в соответствующие им циклически перезаписываемые регистры, входящие в состав

Таблица 2

Вероятные выходные кодовые слова на основе избыточного символа при ошибках в адресе

Начальное значение	00				01				10			
	A	Z	H	G	A	Z	H	G	A	Z	H	G
Без сбоя	00	1	001	011	01	1	011	110	10	1	111	000
	01	1	011	110	10	1	111	000	00	0	110	011
	10	1	111	000	00	0	110	011	01	0	100	110
Сбой в a_1	<u>0</u> 0	1	001	011	<u>0</u> 1	1	011	011	<u>1</u> 0	1	111	000
	10	1	101	101	01	1	011	011	01	0	101	101
	00	0	100	110	10	1	111	000	10	0	001	011
Сбой в a_2	<u>0</u> 0	1	001	011	<u>0</u> 1	1	011	110	<u>1</u> 0	1	111	000
	00	0	000	000	01	0	001	011	01	1	111	000
	01	0	000	000	10	0	001	011	10	1	111	000

сверточного кодера, по установленному для данного такта адресу. Из каждого регистра комбинаторной логикой формируется выходное кодовое слово, которое поступает на передатчик кодового слова и передается в декодер. Кодовое слово $G(H(Z))$, сформированное из избыточного символа, поступает параллельно на все локальные ПЛИС, а кодовые слова из информационных символов $G(H(m_i))$ разделяются на группы в зависимости от того, к какому объекту управления они относятся.

Декодирование начинается с анализа кодовых слов G'_Z , сформированных на основе избыточного символа, из которых восстанавливаются достоверные адреса A , и только после этого на их основе параллельно декодируют все k_0 кодовых слов из символов исходного сообщения. Алгоритм декодирования слов G'_Z строится по принципу конечного автомата, граф которого представлен на рис. 6. Переход к новому состоянию осуществляется на каждом такте по алгоритму, индивидуальному для каждого состояния.

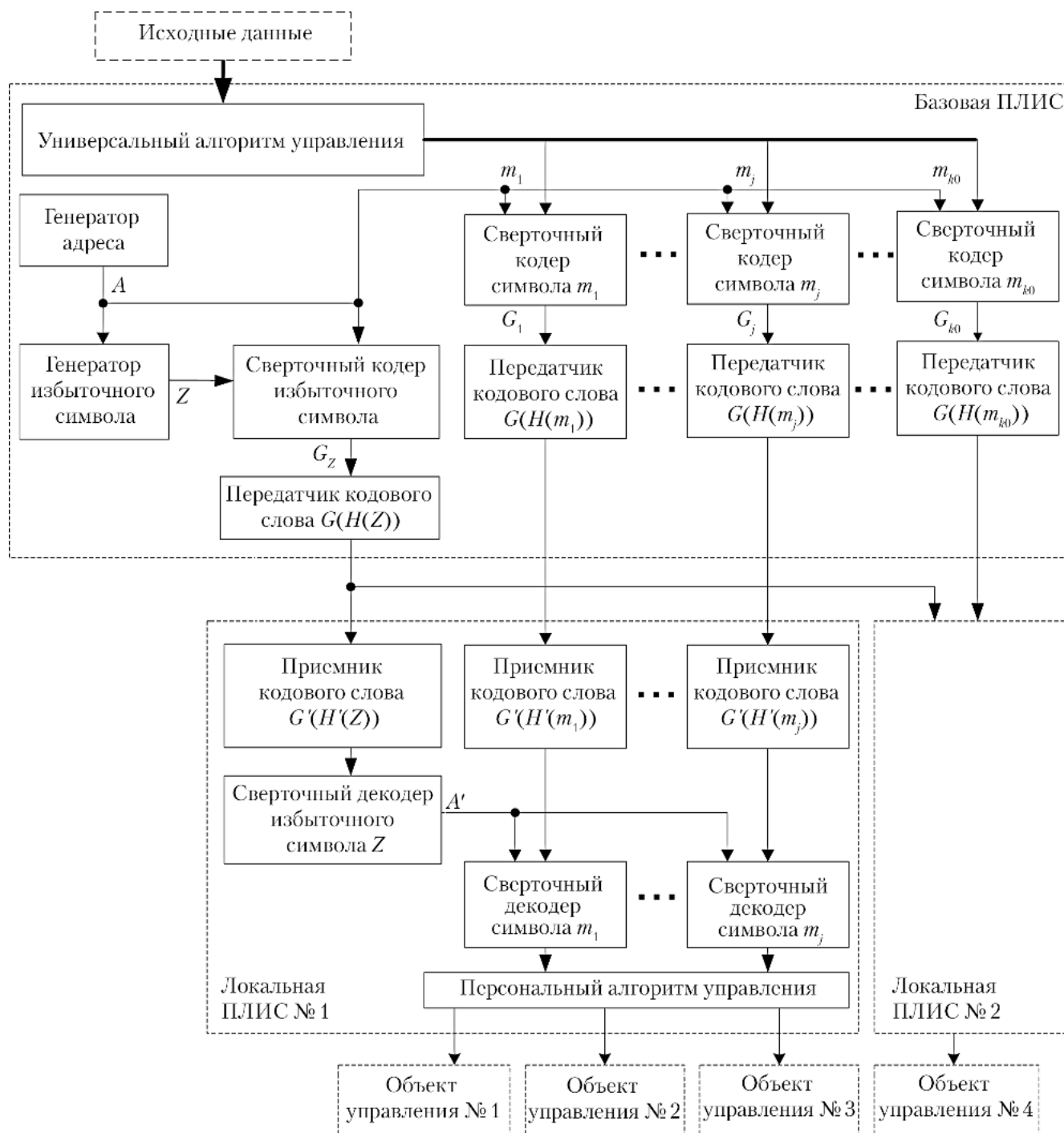


Рис. 5. Функциональная схема сверточных кодера и декодера

В соответствии с рис. 6 восстановление достоверного значения адреса, т.е. переход к состоянию «0», происходит с максимальной задержкой в два такта. Поэтому в декодере реализуется сдвиговый регистр из трех переменных B_1 , B_2 и B_3 , хранящих состояния адреса во время тактов t_0 , t_{0+1} и t_{0+2} . На каждом такте, независимо от состояния конечного автомата, происходит запись B_1 в B_2 и B_2 в B_3 параллельно с выводом нового значения A_1 в адресную шину для сверточных декодеров каждого символа исходного сообщения. Количество записанных новых значений в переменные B_1 , B_2 и B_3 может варьироваться в зависимости от состояния конечного автомата в настоящем такте.

Восстановление достоверного адреса производится на основе сохраненных на предыдущем такте значений адреса и циклически перезаписываемого регистра H'_z . Основным состоянием декодера является состояние «0», логика перехода из которого описывается алгоритмом, представленным на рис. 7. В основе перехода из состояния «0» лежит модифицированный алгоритм Витерби [10], суть которого для данной задачи сводится к перебору всех возможных модификаций H'_z , при условии принятия достоверным значения адреса, и формированию кодовых слов G'_{z1} и G'_{z0} длиной n_0 символов путем n_0 линейных комбинаций разрядов модифицированных регистров

H'_{z1} и H'_{z0} . Как видно из блок-схемы, представленной на рис. 7, необходимо произвести расчет расстояний Хемминга $d(G'(H(Z)), G'_{z1})$ и $d(G'(H(Z)), G'_{z0})$ и проверить равенство одного из расстояний нулю и равенство кодовых слов $G'(H(Z))$, принятых на последних двух тактах. На основе двух решений выбирается следующее состояние, причем переход к состоянию «0» говорит о восстановлении достоверного адреса и выводе его в сдвиговый регистр из переменных B_1 , B_2 и B_3 .

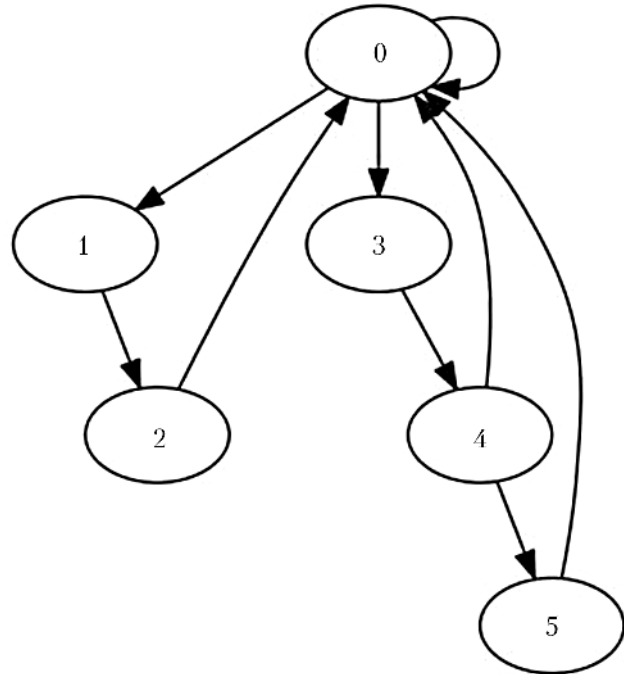


Рис. 6. Граф конечного автомата, описывающего декодирование слов G'_z

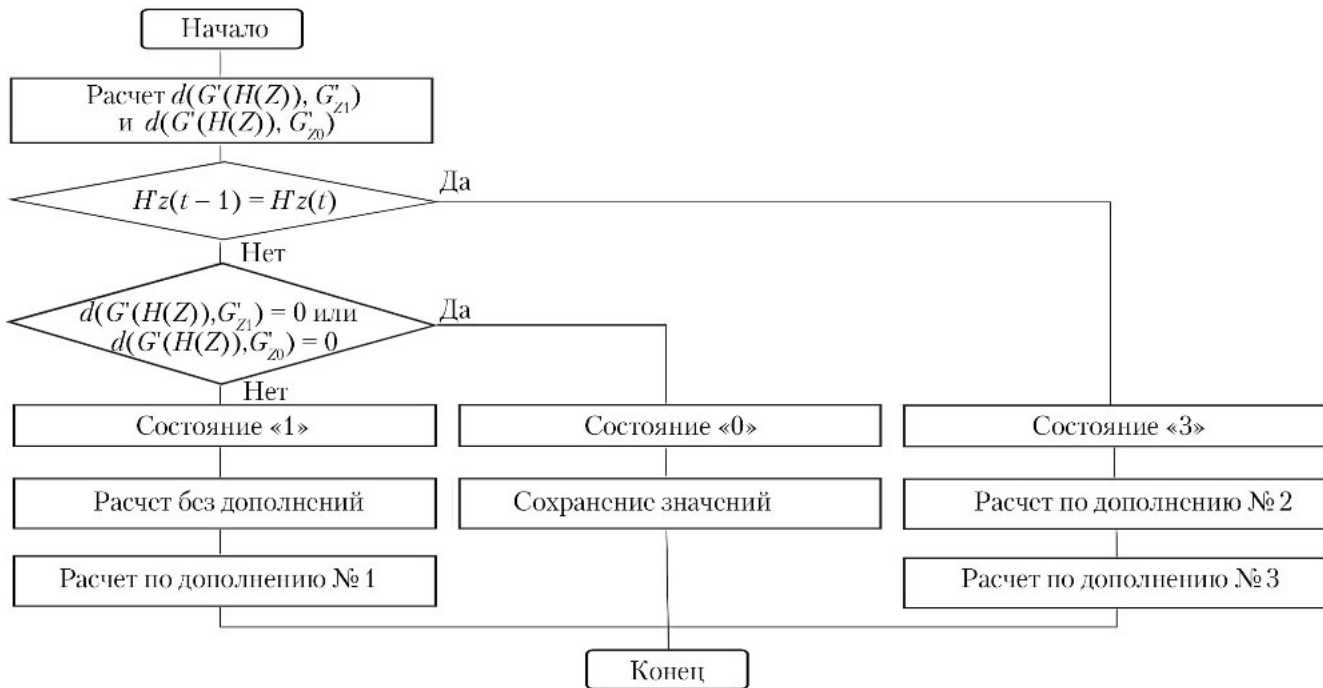


Рис. 7. Блок-схема алгоритма перехода из состояния «0»

Переход к состояниям «1» и «3» говорит о наличии сбоя при формировании адреса. При решении каждой из ветвей: «1» – «2» или «3» – «4» – «5», решается параллельно по два алгоритма на каждом этапе, а окончательное решение выбирается при переходе к состоянию «0».

При переходе из состояния «0» в «1» решаются следующие алгоритмы:

- без дополнений – для сбоя кодового слова на этапе передачи. На данном этапе сохраняются оба модифицированных регистра H'_{z1} и H'_{z0} , а адрес признается верным;

- дополнение № 1 – для сбоя при формировании адреса. Перебираются все возможные варианты адресов и значений, записываемых в регистр, на их основе рассчитываются расстояния Хемминга между ними и принятым словом. Выбирается слово с нулевым расстоянием, и все его параметры сохраняются для дальнейших расчетов.

При переходе из состояния «0» в «3» решаются следующие алгоритмы:

- дополнение № 2 – для сбоя в циклически перезаписываемом регистре. Адрес принимается достоверным, а модифицируемый символ инвертируется, таким образом формируя значение регистра H'_z для дальнейших расчетов.

- дополнение № 3 – для сбоя при формировании адреса. Для дальнейших расчетов перезаписываются значения, сохраненные на предыдущем такте, без изменений.

На следующем такте для всех трех дополнений обоих переходов («1» – «2» и «3» – «4») производятся модификация регистров H'_{z1} и H'_{z0} , исходя из параметров, сохраненных на предыдущем такте; расчет расстояний $d(G'(H(Z)), G'_{z1})$ и $d(G'(H(Z)), G'_{z0})$ и выбор нулевого расстояния; сохранение параметров для следующего перехода и переход к состоянию соответствующего дополнения. Расчет для алгоритма без дополнений отличается необходимостью формировать регистры H'_{z1} и H'_{z0} для каждого из сохраненных на предыдущем такте значений и соответственно выбирать нулевое расстояние из четырех значений вариантов модифицированных кодовых слов.

На третьем такте для состояния «2» осуществляется безусловный переход к состоянию «0». Решение выбирается, исходя из условия равенства кодовых

слов $G'(H(Z))$, принятых на последних двух тактах. В случае истинности условия правильным признается дополнение № 1, для которого производится расчет, аналогичный предыдущему такту, и формируются решения всех трех переменных A_1 , A_2 и A_3 . При ложности условия производится расчет, аналогичный дополнению № 1, но на основе данных, полученных для решения без дополнений при состоянии «1», решения также выводятся в переменные A_1 , A_2 и A_3 .

Для состояния «4» сначала проверяется наличие решений по дополнению № 3, аналогично состоянию «3», в случае наличия расстояния Хемминга, равного нулю, формируются решения всех трех переменных A_1 , A_2 и A_3 , и осуществляется переход к состоянию «0». Если решений нет, то производится аналогичный расчет для исходных данных дополнения № 2, а также осуществляется переход к состоянию «5» с параллельным сохранением обоих модифицированных регистров H'_{z1} и H'_{z0} и выводом решений A_1 и A_2 , что позволяет не увеличивать задержку до трех тактов. Если решение находится, то производятся операции, аналогичные дополнению № 3 на данном такте.

Переход из состояния «5» в «0» осуществляется на следующем такте параллельно с расчетом оставшегося адреса A_3 , аналогично решению без дополнений при состоянии «1».

После восстановления адреса восстанавливаются сбой в символах исходного сообщения с помощью алгоритма, идентичного декодированию избыточного символа без дополнений, т. е. с задержкой в один такт.

Реализация

Математическая модель разработанного метода радиационно-стойкого кодирования для проведения покрывающего тестирования была реализована в среде *Matlab*. В основе проверки лежало тестирование потоков управления по принципу «белого ящика» с покрытия пятого уровня [11]. В соответствии с выбранным уровнем проверялись следующие покрытия:

- покрытие ответвлений – каждая точка разветвления кода алгоритма проверяется хотя бы один раз;

- покрытие условий — условия, имеющие два и более вариантов решения, проверяются при выполнении каждого случая хотя бы по разу;

- покрытие ответвлений и условий — суммарные проверки каждого условия и ответвления.

Тестирование показало, что алгоритм метода кодирования позволяет решать со 100%-ной вероятностью следующие задачи:

- исправление ошибок в каналах передачи кодовых слов из символов исходного сообщения и избыточного символа;

- исправление ошибки, связанной с инверсией значений отдельных битов, хранимых в циклически перезаписываемых регистрах;

- коррекция адреса, используемого для декодирования при ошибке в схеме формирования адреса кодера.

Проверенный алгоритм был реализован на языке описания *VHDL* для интегрирования в кристалл ПЛИС. Для моделирования сбоя был разработан триггер, имеющий дополнительный вход, имитирующий радиационный сбой, и на его основе строилась схема базовой ПЛИС. Схема, представленная на рис. 5, была реализована в макете, в котором в качестве базовой ПЛИС выступала микросхема 5576XC7T, а в качестве двух локальных ПЛИС и объектов управления — *Actel AZPE1500-PQ208* [12].

Кодер избыточного символа, включая схемы генератора адреса и генератора избыточного символа, занял 56 логических элементов, что составляет 3% от всего объема ПЛИС 5576XC7T.

Декодер избыточного символа — 320 логических элементов или 18% объема ПЛИС.

Кодер одного символа исходного сообщения занял 34 логических элемента или 2% объема ПЛИС.

Декодер одного символа избыточного сообщения занял 120 логических элементов или 7% объема ПЛИС.

Заключение

В работе предложен новый подход к реализации межкристальной связи ПЛИС, позволяющий за счет радиационно-стойкого кодирования на основе сверточного кода значительно повысить достоверность передаваемой информации и исключить влияние на нее внешних факторов космического пространства.

Данный подход разделяет борьбу с радиационными сбоями на два процесса: исправление ошибок в передаваемых данных и подмена понятия сбоя в генераторе адреса на асинхронное изменение значения адреса для декодера.

Исходя из объемов, занятых реализованным алгоритмом кодирования, можно сделать вывод, что если целью является разгрузка базовой ПЛИС, в частности, для реализации дополнительных алгоритмов защиты, то за счет использования ($3%+k_0 \cdot 2%$) от объема ПЛИС 5576XC7T можно перенести часть функций, занимающих значительно больший объем, в локальные ПЛИС. Стоит отметить, что с расширением номенклатуры ПЛИС российского производства и соответствующим повышением количества логических элементов на кристалл применение данного подхода позволит реализовывать более сложные алгоритмы за счет использования защищенных обратных связей из локальных ПЛИС в базовую.

Также важным моментом является то, что суммарная задержка для полного цикла составила не более трех тактов или 1,5 мкс (для частоты 2 МГц), что позволяет использовать данный подход без значительного изменения протоколов внешних взаимодействий прибора.

Так как разработанный метод позволяет абстрагироваться от физической модели радиационного сбоя и исходить только из электрических параметров типа отказа (*SEU*, *SET* или *SEL*), то результаты работы можно использовать при разработке бортовых приборов для перспективных КА как на околоземной орбите, так и для межпланетных миссий.

Список литературы

1. Калинов С.Н., Петров С.А., Радченко А.В. Применение систем-на-кристалле СНК на основе встраиваемых в ПЛИС IP-ядер при разработке бортовых приборов и систем управления // Тезисы докладов XXI Научно-технической конференции молодых ученых и специалистов. Королёв: РКК «Энергия», 2017. С. 148–149.
2. Stepanova E.A., Shafran S.V., Kudryavtsev I.A. GALILEO E5 receiver for reliability Improvement of GNSS-based position. // Труды международного симпозиума «Надежность и качество». Пенза: Пензенский государственный университет, 2016. С. 306–308.

3. Каталог изделий АО «Воронежский завод полупроводниковых приборов – Сборка». Воронеж: АО «ВЗПП-С», 2017. 78 с.

4. Каталог продукции АО «ПКК Миландр». Зеленоград: АО «ПКК Миландр», 2017. 113 с.

5. Кларк Дж., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи. М.: Радио и связь, 1987. 392 с.

6. Синицин Д.В. Повышение помехоустойчивости радиотехнических систем передачи информации с использованием сверточных алгоритмов обработки сигналов: дис. ... канд. тех. наук: 05.12.04: защищена 22.01.14: утв. 15.07.14 / Владимир: ОАО «Владимирское КБ радиосвязи», 2014. 127 с.

7. Никитин А.А., Жигулевцев Ю.Н. Применение сверточного кодирования для защиты программируемых логических интегральных схем от единичных отказов, вызванных радиационными отказами космического пространства // Вестник Самарского университета. Аэрокосмическая техника, технологии и машиностроение. 2017. Т. 16. № 4. С. 130–136.

8. Чумаков А.И., Никифоров А.Ю., Телец В.А., Бойченко Д.В., Першенков В.С., Скоробогатов П.К., Яненко А.В., Калашников О.А., Потапенко А.И., Сокоян А.В., Ульянов Р.В., Уланова А.В., Анашин В.С., Протопопов Г.А., Соболев С.А., Улимов В.Н., Артамонов А.С. Радиационная стойкость изделий ЭКБ. Науч. изд. / Под ред. А.И. Чумакова. М.: Нац. исслед. ядерный ун-т «МИФИ», 2015. 512 с.

9. Улимов В.Н., Членов А.М., Танеро К.И. Радиационные эффекты в кремниевых интегральных схемах космического применения. М.: Лаборатория знаний, 2014. 297 с.

10. Блейхут Р. Теория и практика кодов, контролирующих ошибки. М.: Мир, 1986. 576 с.

11. Copeland L. *A practitioner's guide to software test design*. Artech House, 2004. 355 p.

12. ProASIC3/e. Режим доступа: <https://actel.ru/item/proasic-e> (дата обращения 21.03.2018 г.).

Статья поступила в редакцию 04.07.2018 г.

Reference

1. Kalinov S.N., Petrov S.A., Radchenko A.V. *Primenenie sistem-na-kristalle SNK na osnove vstraivaemykh v PLIS IP-yader pri razrabotke bortovykh priborov i sistem upravleniya* [Application of SNK crystal on the basis of the IP kernels which are built in FPGA by development of onboard instruments and management systems]. *Tezisy dokladov XXI Nauchno-tehnicheskoy konferentsii molodykh uchenykh i spetsialistov*. Korolev, RKK Energiya publ., 2017. P. 148–149.

2. Stepanova E.A., Shafran S.V., Kudryavtsev I.A. *GALILEO E5 receiver for reliability improvement of GNSS-based position*. *Works of the International Symposium Reliability and Quality*. Penza, Penza State University publ., 2016. P. 306–308.

3. *Katalog izdeliy* АО «Воронежский завод полупроводниковых приборов – Сборка» [Directory of products of JSC The Voronezh Plant of Semiconductor Items – Assembly]. Voronezh, АО «VZPP-S» publ., 2017. 78 p.

4. *Katalog produktsii* АО «ПКК Миландр» [The directory of production of JSC ПКК Миландр]. Zelenograd, АО «ПКК Миландр» publ., 2017. 113 p.

5. Clark J., Kaine J. *Kodirovaniye s ispravleniem oshibok v sistemakh tsifrovoy svyazi* [Coding with correction of errors in the systems of digital communication]. Moscow, Radio i svyaz' publ., 1987. 392 p.

6. Sinitsin D.V. *Povyshenie pomekhoustoychivosti radiotekhnicheskikh sistem peredachi informatsii s ispol'zovaniem svertochnykh algoritmov obrabotki signalov* [Increase in noise immunity of radio engineering information transmission systems with use of convolution algorithms of signal processing]. *Dis. ... kand. tekh. nauk: 05.12.04: zashchishchena 22.01.14: utv. 15.07.14*. Vladimir, ОАО «Владимирское КБ радиосвязи» publ., 2014. 127 p.

7. Nikitin A.A., Zhigulevtsev Yu.N. *Primenenie svertochnogo kodirovaniya dlya zashchity programmiruemykh logicheskikh integral'nykh skhem ot edinichnykh otkazov, vyzvannykh radiatsionnymi otkazami kosmicheskogo prostranstva* [Application of convolutional coding for protection of programmable logic integrated circuits against the single failures caused

by radiation failures of space]. *Vestnik Samarskogo universiteta. Aerokosmicheskaya tekhnika, tekhnologii i mashinostroenie*, 2017, vol. 16, no. 4, pp. 130–136.

8. Chumakov A.I., Nikiforov A.Yu., Telets V.A., Boychenko D.V., Pershenkov V.S., Skorobogatov P.K., Yanenko A.V., Kalashnikov O.A., Potapenko A.I., Sogoyan A.V., Ul'yanenkov R.V., Ulanova A.V., Anashin V.S., Protopopov G.A., Sobolev S.A., Ulimov V.N., Artamonov A.S. *Radiatsionnaya stoykost' izdeliy EKB* [Radiation firmness of products EKB]. Ed. by A.I. Chumakov. Moscow, Nats. issled. yadernyy un-t «MIFI» publ., 2015. 512 p.

9. Ulimov V.N., Chlenov A.M., Tapero K.I. *Radiatsionnye efekty v kremnievykh integral'nykh skhemakh kosmicheskogo primeneniya* [Radiation effects in silicon integrated circuits of space application]. Moscow, Laboratoriya znaniy publ., 2014. 297 p.

10. Bleykhut R. *Teoriya i praktika kodov, kontroliruyushchikh oshibki* [The theory and practice of the codes controlling errors]. Moscow, Mir publ., 1986. 576 p.

11. Copeland L. *A practitioner's guide to software test design*. Artech House, 2004. 355 p.

12. ProASIC3/e. Available at: <https://actel.ru/item/proasic-e> (accessed 21.03.2018).