

## **Research of the Finite State Machine with Programmable Logic as a Part of Digital Information and Control System Based on FPGA**

**Barkalov A.<sup>1</sup>, Titarenko L.<sup>1-2</sup>, Zeleneva I.<sup>3</sup>, Hrushko S.<sup>3</sup>**

<sup>1</sup>University of Zielona Gora, Zielona Gora, Poland

<sup>2</sup>Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

<sup>3</sup>Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine

**Abstract.** The purpose of this research is to determine the effective way of implementation of the control algorithm, as an important functional part of information and control computer systems. The search criterion is the minimum of internal hardware FPGA resources required for the control unit implementation. This criterion allows miniaturization of dimensions, improves reliability by reducing the number of interconnections inside the chip, and ensures the possibility of a compact arrangement of various system components, which is especially important when using the "system-on-chip" design technology. The control unit holds a prominent place in the digital information and control systems. A comparative analysis of two control unit models represented as a finite state machine with either hard or programmable logic was proposed. Advantages and disadvantages of both models were determined according to the peculiarities of the information and control system algorithms, and, it was proved that the FSM model with a programmable logic matched these peculiarities in a greater degree. The purpose of this study was achieved due to the application of the proposed method for the implementation of the FSM with the programmable logic using the embedded memory of the FPGA and ProASIC chips. The main result was a substantial decrease in the LUT number used. The experimental results were obtained applying the chips of the world's top manufacturers - Xilinx, Altera/Intel, Microsemi. The studies were carried out based on the onboard computing complex control algorithm.

**Keywords:** control algorithm, digital information and control system, FSM with programmable logic, FPGA, embedded memory.

**DOI:** 10.5281/zenodo.3239172

### **Studiul mașinii cu logică programabilă ca parte a unui sistem digital de management al informațiilor pe FPGA**

**Barkalov A.A.<sup>1</sup>, Titarenko L.A.<sup>1-2</sup>, Zeleneva I.Ya.<sup>3</sup>, Grushko S.C.<sup>3</sup>**

<sup>1</sup> Universitatea Zelenogursky, Green Gora, Polonia

<sup>2</sup> Universitatea Națională de Radio Electronică din Kharkhov, Kharkhov, Ucraina

<sup>3</sup>Universitatea Tehnică Națională din Zaporoje, Zaporoje, Ucraina

**Rezumat.** Scopul acestor studii este de a găsi o modalitate eficientă de implementare a algoritmului de control, ca o parte importantă a sistemelor computerizate de informare și control. Criteriul de căutare este condiția celor mai mici costuri ale resurselor interne ale cipului FPGA în timpul implementării circuitului dispozitivelor de comandă. Minimizarea costurilor hardware este un factor important în proiectarea hardware-ului sistemelor moderne de management al informațiilor, deoarece permite miniaturizarea dimensiunilor, îmbunătățește fiabilitatea prin reducerea numărului de interconexiuni și oferă de asemenea posibilitatea unui aranjament compact al diverselor componente ale sistemului, ceea ce este deosebit de important atunci când se utilizează tehnologia de proiectare system-on-chip ". Ca parte a sistemelor digitale de informare și control, un loc important este ocupat de unitatea de comandă. A fost efectuată o analiză comparativă a metodelor de implementare a unui dispozitiv de comandă sub formă de automate cu logică rigidă și programabilă. Avantajele și dezavantajele acestora sunt determinate în conformitate cu particularitățile algoritmilor sistemelor de control al informațiilor. Se arată că modelul automatului de control cu logică programabilă corespunde mai mult particularităților acestor algoritmi decât modelul automatului cu logică rigidă. Scopul acestor studii a fost realizat prin metoda propusă de implementare a unui circuit logic automat cu o logică programabilă care utilizează cipurile integrate FPGA și ProASIC. Rezultatele studiilor prezentate confirmă eficacitatea acestei metode. Rezultatul principal este o reducere semnificativă a numărului de LUT-uri utilizate în implementarea schemei. Datele experimentale au fost obținute pentru cei mai importanți producători de chipuri din lume - Xilinx, Altera / Intel, Microsemi.

**Cuvinte-cheie:** algoritm de control, informație digitală și sistem de control, mașină automată cu logică programabilă, FPGA FPGA, memorie încorporată.

## Исследование автомата с программируемой логикой в составе цифровой информационно-управляющей системы на FPGA

Баркалов А.А.<sup>1</sup>, Титаренко Л.А.<sup>1,2</sup>, Зеленева И.Я.<sup>3</sup>, Грушко С.С.<sup>3</sup>

<sup>1</sup>Университет Зеленогурский, Зеленая Гура, Польша

<sup>2</sup>Харьковский национальный университет радиоэлектроники, Харьков, Украина

<sup>3</sup>Запорожский национальный технический университет, Запорожье, Украина

**Аннотация.** Целью данных исследований является поиск эффективного способа реализации алгоритма управления, как важной функциональной части информационно-управляющих компьютерных систем. Критерием поиска служит условие наименьших затрат внутренних ресурсов кристалла ПЛИС при имплементации схемы устройства управления. Минимизация аппаратных затрат является важным фактором при проектировании аппаратной части современных информационно-управляющих систем, так как обеспечивает возможность миниатюризации габаритов, способствует повышению надежности за счет уменьшения числа межсоединений, а также обеспечивает возможность компактного расположения различных составляющих системы, что особенно актуально при использовании технологии проектирования «система-на-кристалле». В составе цифровых информационно-управляющих систем важное место занимает устройство управления. В работе выполнен сравнительный анализ способов реализации устройства управления в виде автоматов с жесткой и программируемой логикой. Определены их достоинства и недостатки в соответствии с особенностями алгоритмов информационно-управляющих систем. Показано, что модель управляющего автомата с программируемой логикой более соответствует особенностям указанных алгоритмов, чем модель автомата с жесткой логикой. Цель данных исследований достигнута за счет предложенного способа имплементации схемы автомата с программируемой логикой с использованием встроенной памяти микросхем FPGA и ProASIC. Приведены результаты исследований, которые подтверждают эффективность данного способа. Главным результатом является значительное сокращение числа использованных LUT при имплементации схемы. Экспериментальные данные получены для микросхем ведущих мировых фирм-производителей – Xilinx, Altera/Intel, Microsemi. Исследования выполнены на основе алгоритма управления бортовым вычислительным комплексом.

**Ключевые слова:** алгоритм управления, цифровая информационно-управляющая система, автомат с программируемой логикой, ПЛИС FPGA, встроенная память.

### ВВЕДЕНИЕ

Современные цифровые информационно-управляющие системы (ЦИУС) [1-3, 8] получают все более широкое распространение во многих сферах производства, в организации управления сложными технологическими объектами и процессами, в проектировании «умного» дома и даже города, и т.п. В частности, в данной работе речь пойдет о системах управления в составе бортового вычислительного комплекса. Наличие множества сложных компонентов и подсистем, которые обеспечивают измерение параметров, внесение необходимых поправок, мониторинг производительности, энергопотребления и др., влекут за собой все большее усложнение управляющих систем [4, 5]. Несмотря на это, системы измерения и управления, применяемые, например, в бортовом оборудовании, должны обеспечивать достаточно высокую надежность, соответствие стандартам и безопасность процессов [2, 10].

Программируемые логические интегральные схемы (ПЛИС) являются современной элементной базой,

используемой в мировой промышленности, которая позволяет реализовывать как простые и портативные устройства, так и сложные цифровые системы контроля и управления [5, 21-22]. Различные функции ЦИУС реализуются в базе ПЛИС надежным и удобным способом. Использование ПЛИС имеет ряд преимуществ в решении вопросов повышения надежности проектируемых систем путем резервирования, а также распараллеливания процессов управления [11].

Применение технологии «система-на-кристалле» является одной из эффективных мер по обеспечению достаточного уровня надежности. В таких системах за счет высокой оптимизации достигается снижение энергопотребления, повышение отказоустойчивости и производительности, уменьшение объема средств и времени для аппаратной отладки. Применение «систем-на-кристалле» позволяет реализовать в пределах одного кристалла как условно-распределенный вычислительный комплекс, так и модуль обнаружения неисправности и управления переключением, что, в частности,

актуально для встроенных и бортовых систем [9].

В печати представлено достаточно много публикаций, посвященных новым подходам к эффективному использованию свойств гибкой внутренней архитектуры ПЛИС FPGA, что дает возможности оперативного реконфигурирования, дистанционного перепрограммирования, разработки резервированных и многоверсионных систем [6-7, 11-14, 21]. Все эти меры направлены в конечном итоге на повышение отказоустойчивости и гарантоспособности информационно-управляющих систем, особенно для применения в критичных условиях или бортовом оборудовании.

Минимизация аппаратных затрат при реализации схемы цифрового устройства на ПЛИС [15-19] также является не менее важным фактором, который необходимо учитывать на этапе проектирования с целью повышения надежности системы в целом. Необходимость разработки методов проектирования систем на ПЛИС, ориентированных на снижение аппаратных затрат, обусловлена, помимо микроминиатюризации, повышением надежности системы за счет упрощения функций, описывающих алгоритм управления, представленный микропрограммным автоматом.

Упрощение функций ведет к уменьшению аппаратной избыточности, а также к уменьшению объема конфигурационной памяти [2, 7-9, 19-22]. Стойкость к одиночным сбоям для ПЛИС на основе ячеек статической памяти зависит от стойкости отдельных ее составляющих: конфигурационной памяти, автоматов управления конфигурационной памятью и пользовательских триггеров. Конфигурационная память определяет выполняемую ПЛИС логическую функцию. При этом сбой в конфигурационной памяти производит более негативный эффект, чем просто одиночный сбой в регистре данных.

С течением времени накопление подобных ошибок в конфигурационной памяти приводит к сбою даже при резервировании. Поскольку степень влияния одиночных сбоев на конфигурационную память ПЛИС зависит от ее объема, уменьшение количества ячеек конфигурационной памяти повысит стойкость ПЛИС к одиночным сбоям и, как следствие, надежность схемы в целом.

Важной частью цифровых информационно-управляющих систем являются устройства управления, которые могут быть реализованы в виде автомата с жесткой или программируемой логикой [9, 16-19]. В большинстве случаев, судя по публикациям, используется модель автомата с жесткой логикой (АЖЛ), а именно – автомат Мура, что обусловлено относительной простотой и устойчивостью, поскольку выходные сигналы зависят только от состояний автомата. Автомат Мура также обеспечивает максимальное быстродействие за счет одновременного анализа всех логических условий, которые служат входными сигналами схемы автомата [19-20].

Однако алгоритмы функционирования систем управления имеют структурную особенность, состоящую в том, что логических вершин в таком алгоритме достаточно много – это соответствует частому опросу всевозможных датчиков и других внешних устройств [1, 4-5]. Кроме того, характерно чередование логических и операторных вершин, что соответствует процессу «опрос датчика – реакция устройства». Как показали практические исследования авторов, реализация таких алгоритмов в виде управляющих автоматов с жесткой логикой может оказаться неэффективной по нескольким причинам: системы функций переходов и выходов получаются громоздкими, термы имеют большую размерность, что неизбежно ведет к избыточному использованию внутренних ресурсов кристалла (блоков LUT, межсоединений) при имплементации схемы. Кроме того, затрудняется автоматизация процесса формирования систем функций, что также приводит к ошибкам и усложняет проектирование и отладку.

Автоматы с программируемой логикой (АПЛ - programmable logic finite state machine, PLFSM) позволяют лучше учесть описанные выше свойства алгоритмов управления объектами за счет явной адресации переходов. В процессе синтеза схемы АПЛ алгоритм управления преобразуется в массив битовых строк, соответствующих управляющим словам, и записывается в управляющую память. Комбинационная схема, обеспечивающая корректное считывание и обработку управляющих слов, требует меньше логических ресурсов в сравнении с АЖЛ, поскольку в модели АПЛ

реализуемые системы функций имеют намного меньший ранг и объем. Недостатком в этом случае является некоторое снижение быстродействия по той причине, что логические условия (ЛУ) анализируются в АПЛ по одному в каждом такте, и при большом числе ЛУ в алгоритме управления снижение быстродействия будет существенным. Решить эту проблему в определенной степени можно путем использования ресурсов быстродействующей встроенной памяти FPGA [20, 23-24].

Целью данных исследований является поиск наилучшего способа реализации устройства управления, как важной части информационно-управляющей компьютерной системы. Критерием поиска служит условие наименьших затрат внутренних ресурсов кристалла ПЛИС при имплементации схемы управляющего автомата. Выполнение этого критерия, в свою очередь, позволит улучшить такие показатели, как надежность, энергопотребление, уменьшить габариты устройства. Цель исследований достигнута за счет предложенного в статье способа имплементации схемы управляющего автомата с программируемой логикой с использованием встроенной памяти микросхем FPGA и ProASIC. Приведены результаты исследований, которые подтверждают эффективность данного способа. Главным результатом является значительное сокращение числа использованных LUT при имплементации схемы. Другие важные характеристики схемы при этом не ухудшаются. Экспериментальные данные получены авторами для микросхем ведущих мировых фирм-производителей – Xilinx, Altera/Intel, Microsemi.

### МЕТОДЫ

В качестве решения поставленной задачи предлагается рассмотреть автомат с программируемой логикой и принудительной адресацией микрокоманд [18]. При этом необходимо выполнить модификацию известной структуры, ориентированную на особенности элементного базиса FPGA и ProASIC [25-27]. Общим структурным свойством этих микросхем является наличие как распределенных матричных ресурсов LUT (look-up-table), так и быстродействующей встроенной памяти EMB (embedded memory blocks).

Базовая структура автомата с программируемой логикой (рис.1) включает комбинационный блок, который в свою очередь состоит из схемы формирования адреса микрокоманды (microcommand address generator, MAG) и схемы формирования микроопераций (micro-operation formation scheme, MOFS), а также блок управляющей памяти (control memory, CM), который хранит микропрограмму управления всем цифровым устройством. Кроме того, в схеме АПЛ обязательно есть регистр адреса микрокоманды (microcommand address register, MAR) [18].

Для синтеза автомата с программируемой логикой исходная граф-схема алгоритма (ГСА) должна быть представлена в виде микропрограммы (МП), состоящей из микрокоманд заданного формата. Микрокоманда (МК) содержит операционную часть (ОЧ - operating part, OP) и адресную часть (АЧ - address part, AP).

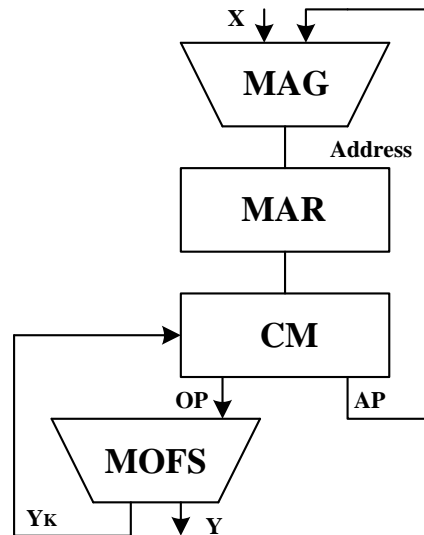


Рис.1. Структура автомата с программируемой логикой<sup>1</sup>.

В процессе функционирования АПЛ очередная МК считывается из УП по адресу, находящемуся в регистре адреса микрокоманд. Операционная часть МК поступает на схему формирования микроопераций и преобразовывается в микрооперации Y, управляющие операционным автоматом. Адресная часть МК используется схемой формирования адреса, наряду с логическими условиями X для формирования в регистре адреса следующей микрокоманды. Функционирование завершается после

<sup>1</sup>Appendix 1

формирования признака окончания микропрограммы  $Y_k$ .

Автоматы с программируемой логикой отличаются регулярностью структуры и достаточно простыми алгоритмами синтеза. При этом всем АПЛ присущ существенный недостаток: если переход является L-направленным, то есть зависит от L логических условий, то он выполняется за L тактов. Это приводит к увеличению числа микрокоманд в микропрограмме, за счёт ввода L-1 дополнительных микрокоманд безусловного перехода и, как следствие, к увеличению времени выполнения микропрограммы. Частично этот недостаток можно сгладить за счёт принудительной адресации микрокоманд в формате управляющей памяти.

Принудительная адресация определяет такой формат микрокоманды, который включает поля FY (операционная часть), FX (проверяемое логическое условие), FA0 (адрес перехода при равенстве условия нулю или при безусловном переходе), FA1 (адрес перехода при равенстве логического условия единице). Этот способ адресации позволяет комбинировать пары «операторная + условная» в одну микрокоманду, что приводит к уменьшению числа микрокоманд в микропрограмме и сокращению времени выполнения. Но при этом возрастает объём управляющей памяти, что при автоматической имплементации АПЛ в микросхему FPGA приводит к дополнительному использованию ресурсов LUT (look-up-table).

Предлагается модифицированная структура АПЛ, в которой вся комбинационная часть (это схемы формирования адресов и микроопераций) выполнена на LUT, а таблица управляющей памяти реализована на блоках встроенной памяти ЕМВ (embedded memory blocks). Регистр адреса микрокоманды реализуется на распределенных триггерах кристалла [25-27]. Такая структура не реализуется автоматически средствами САПР. Для ее реализации нужно выполнить специальную подготовку программного описания схемы автомата на языке описания аппаратуры. В частности, проекты АПЛ с базовой структурой в данных исследованиях были выполнены на языке VHDL и затем использованы в пакетах Quartus II, Vivado, Libero для исследования микросхем фирм

Altera/Intel, Xilinx, Microsemi соответственно. Проект АПЛ со встроенной памятью разработан в пакете Quartus II [28].

## РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

Исследования были проведены на основе реального алгоритма управления конфигурацией бортовой информационно-управляющей системы. Изначально задача состояла в том, чтобы синтезировать на микросхемах FPGA и, как более дорогостоящий вариант, ProASIC, схему устройства управления, которая обеспечит малые габариты, надежность и экономное энергопотребление. Исходные данные были представлены в виде функционального дерева с отмеченными состояниями автомата Мура. В ходе решения поставленной задачи были развернуты исследования с целью поиска наилучших вариантов.

Продукция мировых лидеров производства микросхем – Altera/Intel, Xilinx, Microsemi – выбрана в качестве элементного базиса по той причине, что она соответствует одним из самых высоких квалификационных стандартов: стандарту MIL-Std 883 класса B, задающему методы контроля микроэлектронных устройств и процедуры проверки, применяемые в аэрокосмических системах, а также стандарту MIL-PRF-38535 класса Q и N, устанавливающему нормы производительности (performance), качества (quality) и надежности (reliability) [2, 25-27].

Задача проектирования устройства управления для бортовой системы была решена несколькими способами:

- структурным описанием автомата Мура по состояниям, указанным в исходных данных;
- потоковым описанием с помощью графа переходов, что упрощает процесс решения большой задачи;
- в виде автомата с программируемой логикой и автоматической произвольной имплементацией в микросхему;
- в виде автомата с программируемой логикой и специальной имплементацией управляющей памяти на встроенных ЕМВ блоках.

Подробные результаты исследований для микросхем фирмы Altera/Intel, выполненные в пакете Quartus II, приведены в таблице 1. Очевидно, что потоковое описание неэффективно для решения поставленной

задачи, о чем свидетельствует большое число задействованных регистров.

Таблица 1<sup>2</sup>

Результаты исследования разных способов реализации алгоритма управления на FPGA фирмы Altera/Intel

Chipsets	Specification	Algorithm			
		Structural description	Streaming description	PLFSM	PLFSM using embedded memory
Cyclone III (EP3C5E144A7)	Number of LUTs	138	104	106	16
	Embedded memory, bit	0	0	0	2496
	Number of Rg	6	52	6	6
	Total Thermal Power Dissipation, mW	58,55	58,55	59,11	60,04
	Core Static Thermal Power Dissipation, mW	46,12	46,11	46,11	46,11
	I/O Thermal Power Dissipation, mW	12,44	12,43	13	13,93
	Junction Temperature, C	25,7	25,7	25,8	25,8
Cyclone IV GX (EP4CGX15BF14C6)	Number of LUTs	139	107	108	15
	Embedded memory, bit	0	0	0	2496
	Number of Rg	6	52	6	6
	Total Thermal Power Dissipation, mW	166,64	166,64	67,07	69,04
	Core Static Thermal Power Dissipation, mW	153,34	153,34	58,84	58,84
	I/O Thermal Power Dissipation, mW	13,3	13,3	8,24	10,2
	Junction Temperature, C	27,4	27,4	26	26
Stratix III (EP3SE50F484C2)	Number of LUTs	107	82	51	11
	Embedded memory, bit	0	0	0	2496
	Number of Rg	6	52	6	6
	Total Thermal Power Dissipation, mW	425,5	427,23	425,38	425,95
	Core Static Thermal Power Dissipation, mW	397,32	399,06	397,02	397,29
	I/O Thermal Power Dissipation, mW	28,18	28,18	28,36	28,66
	Junction Temperature, C	26,7	26,7	26,7	26,7
Stratix IV (EP4SGX180DF29C2X)	Number of LUTs	105	82	49	11
	Embedded memory, bit	0	0	0	2304
	Number of Rg	6	52	6	6
	Total Thermal Power Dissipation, mW	842,3	844,05	804,06	804,22
	Core Static Thermal Power Dissipation, mW	799,34	801,11	761,95	761,97
	I/O Thermal Power Dissipation, mW	42,96	42,94	42,11	42,25
	Junction Temperature, C	27,4	27,4	27,3	27,3

<sup>2</sup>Appendix 1

Chipsets	Specification	Algorithm			
		Structural description	Streaming description	PLFSM	PLFSM using embedded memory
ArriaGX (EP1AGX20CF484C6)	Number of LUTs	104	83	49	11
	Embedded memory, bit	0	0	0	2496
	Number of Rg	6	52	6	6
	Total Thermal Power Dissipation, mW	406,44	406,54	406,59	407,14
	Core Static Thermal Power Dissipation, mW	383,95	383,95	383,95	383,96
	I/O Thermal Power Dissipation, mW	22,49	22,59	22,73	23,17
	Junction Temperature, C	26,6	26,6	26,6	26,6

Как и предполагалось авторами, автомат с программируемой логикой, в сравнении с автоматом Мура, дает решение, более эффективное по использованию ресурсов кристалла, при сохранении других характеристик (рассеиваемая и потребляемая мощность) примерно на одном уровне.

Меньше всего ресурсов LUT требуется при реализации АПЛ с использованием встроенной памяти, причем ЕМВ блоки

задействованы достаточно компактно, что можно увидеть с помощью просмотрщика.

Результаты исследований для всех трех фирм (Altera/Intel, Xilinx, Microsemi) приведены на рисунках 2–4. Сравнительная диаграмма результатов имплементации АПЛ с применением блоков встроенной памяти и без них – на рис.5.

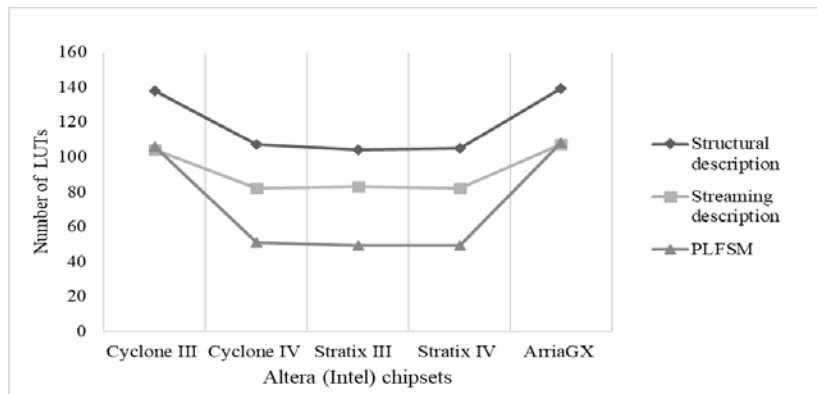


Рис. 2. Зависимость числа использованных LUT при имплементации управляющего автомата в микросхемы Altera/Intel<sup>3</sup>.

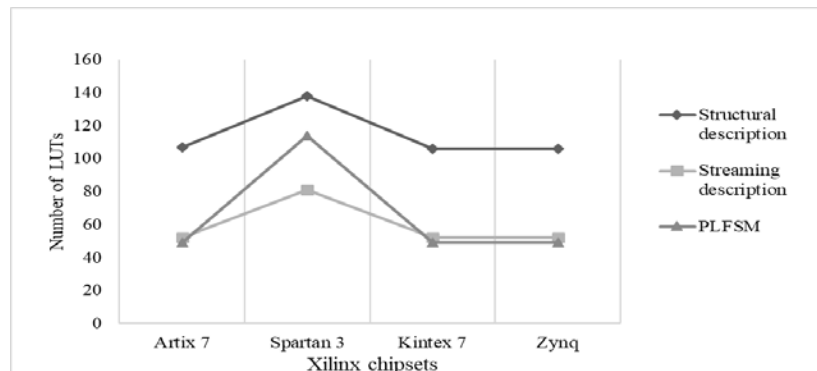


Рис. 3. Зависимость числа использованных LUT при имплементации управляющего автомата в микросхемы Xilinx<sup>4</sup>.

<sup>3,4</sup>Appendix 1

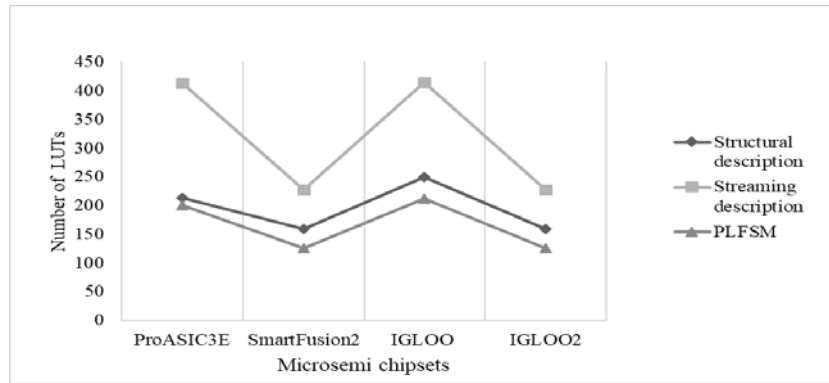


Рис. 4. Зависимость числа использованных LUT при имплементации управляющего автомата в микросхемы Microsemi<sup>5</sup>.

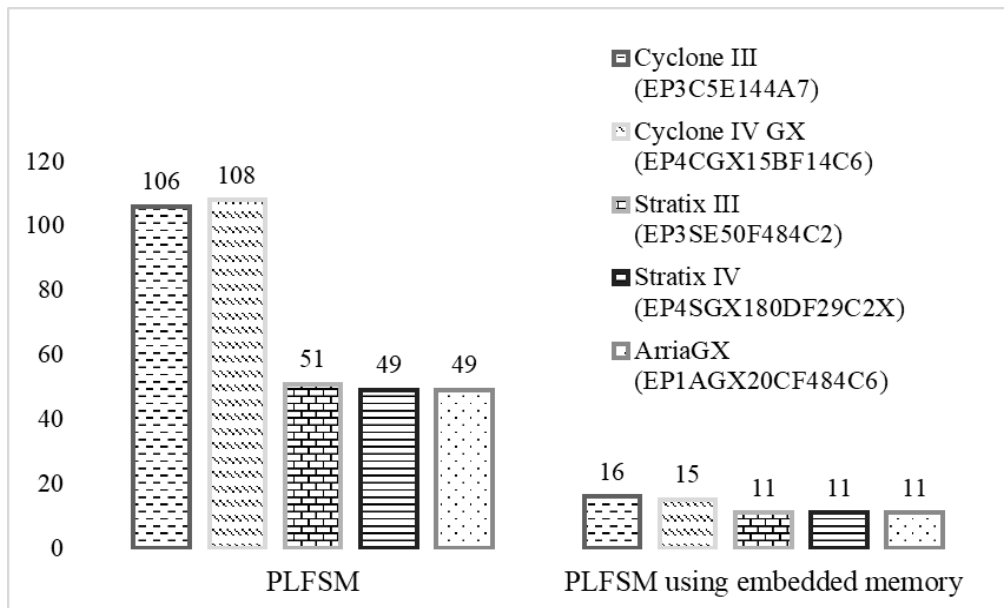


Рис. 5. Количество задействованных LUT при двух разных вариантах имплементации автомата с программируемой логикой<sup>6</sup>.

На графиках видно, что аппаратные затраты практически всегда имеют наименьшее значение в случае имплементации автомата с программируемой логикой.

Результаты исследований показали, что автомат с программируемой логикой, в сравнении с автоматом Мура, обеспечивает существенное уменьшение используемых внутренних ресурсов ПЛИС при реализации такого алгоритма управления сложной системой, который построен по принципу «опрос датчиков – реакция устройства управления». Следует также отметить, что имплементация АПЛ с использованием встроенной памяти ПЛИС позволяет настолько экономно использовать ресурсы LUT, что становится возможным реализовать

другие устройства на освободившихся LUT. Это, в свою очередь, дает возможность выполнить, например, тройное модульное резервирование комбинационных узлов, сбои в функционировании которых могут оказаться критичными для работы устройства в целом.

### ЗАКЛЮЧЕНИЕ

При использовании предлагаемого подхода к имплементации автомата с программируемой логикой, можно улучшить показатели надежности как за счет уменьшения количества внутренних межсоединений кристалла, так и за счет возможности резервирования – локально, на том же кристалле, что не повлияет на стоимость микросхемы и габариты



устройства. Встроенной памятью обладают все рассмотренные в статье, а также другие, новейшие, семейства микросхем ПЛИС, поэтому можно рекомендовать данный

подход для широкого использования при проектировании цифровых информационно-управляющих систем.

#### ПРИЛОЖЕНИЕ 1 (APPENDIX 1)

<sup>1</sup>**Fig.1** The structure of a finite state machine with programmable logic.

<sup>3</sup>**Fig. 2.** Dependence of the number of used LUTs when implementing the control FSM in Altera (Intel) chips.

<sup>4</sup>**Fig. 3.** Dependence of the number of used LUTs when implementing the control FSM in Xilinx chips.

<sup>5</sup>**Fig. 4.** Dependence of the number of used LUTs when implementing the control FSM in Microsemi chips.

<sup>6</sup>**Fig. 5.** The number of used LUTs for different approaches to implementation of the FSM with programmable logic

<sup>2</sup>**Table 1.** The results of the study of different ways to implement the control algorithm on the FPGA company Altera/Intel

#### Литература (References)

[1] Adamski M.A., Karatkevich A., Wegrzyn M. Design embedded control system, 2005. – 267 p.  
 [2] Fedukhin A.V., Mukha A.A., Mukha A.A. PLIS-sistemy kak sredstvo povysheniya otkazoustoychivosti [FPGA systems as a means of increasing failure resistance]. *Matematichni mashini i sistemi - Mathematical Machines and Systems*, 2010, no. 1, pp. 198-204. (In Russian).  
 [3] Mavis D., Eaton P. SEU and SET Mitigation Techniques for FPGA Circuit and Configuration Bit Storage Design. *Proceedings MAPLD Conference*, 2000.  
 [4] Khamis A., Zydek D., Borowik, G., and Subbaram Naidu, D. Control System Design Based on Modern Embedded Systems, 2013, pp. 491–498.  
 [5] Navabi Z. Embedded Core Design with FPGAs, 2007, 433 p.  
 [6] Palagin A.V., Opanasenko V.N. Reconfigurable computing technology. *Cybernetics and Systems Analysis*, 2007, vol. 43, no 5, pp. 675-686.  
 [7] Rafla N. I., Gauba I. A. reconfigurable pattern matching hardware implementation using on-chip RAM-based FSM. *In 2010 53rd IEEE International Midwest Symposium on Circuits and Systems*, 2010, pp. 49–52.  
 [8] Bakhmach, V. Kharchenko, A. Siora, V. Sklyar, A. Andrashov. Experience of I&C Systems Modernization Using FPGA Technology. *International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies (NPIC & HMIT 2010): proceeding of the 7th International Topical Meeting*, 2010, pp. 1345-1352.  
 [9] Klimovicz A.S., Solov'ev V.V. Structural models of finite-state machines for their implementation on

programmable logic devices and systems on chip. *Journal of Computer and Systems Sciences International*, 2015, vol. 54, no. 2, pp. 230-242.

[10] Katz R., Wang J., Reed R., Kleyner I., D'Ordine M., McCollum J., Cronquist B., Howard J. The Effects of Architecture and Process on the Hardness of Programmable Technologies. *IEEE Transactions on Nuclear Science*, 1999, pp. 1736-1743.

[11] Grout I. Digital systems design with FPGAs and CPLDs, 2011. 784 p.

[12] Sunggu L. Advanced Digital Logic Design Using VHDL, State Machines, and Synthesis for FPGA's, 2005. 488 p.

[13] Farooq U. Tree-based Heterogeneous FPGA Architectures: Application Specific Exploration and Optimization, 2012. 188 p.

[14] Daigneault M.A., David J. P. Fast description and synthesis of control-dominant circuits. *Computers and Electrical Engineering*, 2014, vol. 40, no. 4, pp. 1199–1214.

[15] Rawski M. Efficient variable partitioning method for functional decomposition. *Electronics and Telecommunications Quarterly*, 2007, vol. 53, no. 1, pp. 63–81.

[16] Bomar B. Implementation of microprogrammed control in FPGAs. *IEEE Transactions on Industrial Electronics*, 2002, vol. 49, no. 2, pp. 415–422.

[17] Barkalov A., Titarenko L., Kolopienczyk M., Mielcarek K. Bazydlo G. Logic Synthesis for FPGA-based Finite State Machines, 2016. 280 p.

[18] Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Units, 2008. 272 p.

[19] Sklyarov V., Sklyarova I., Barkalov A., Titarenko L. Synthesis and Optimization of FPGA-Based Systems, 2014, 432 p. doi: [10.1007/978-3-319-04708-9](https://doi.org/10.1007/978-3-319-04708-9)

[20] Barkalov A., Titarenko L., Kolopienczyk M. Design of EMB-based Moore FSMs. *Journal of Circuits, Systems and Computers*, 2017, vol. 26, no. 7, pp. 1-23.

[21] Maxfield C. The Design Warrior's Guide to FPGAs, 2004. 542 p.

[22] Chu P. RTL Hardware Design Using VHDL: Coding for Efficiency, Portability and Scalability, 2006. 669 p.

[23] Senhadji-Navarro R., García-Vargas I., Guisado J. L. Performance evaluation of RAM-based implementation of Finite State Machines in FPGAs. *19th IEEE International Conference on Electronics, Circuits, and Systems*, 2012. pp. 225–228.

[24] Borowik G., Łuba T., Falkowski B. Logic synthesis method for pattern matching circuits implementation in FPGA with embedded memories.

*12th Design and Diagnostics of Electronic Circuits and Systems*, 2009. pp. 230–233.

[25] Intel FPGA and programmable devices. Available at: <https://www.intel.com/content/www/us/en/products/programmable.html> (accessed 26.03.2019)

[26] FPGA Leadership across Multiple Process Nodes. Available at: <https://www.xilinx.com/products/silicon-devices/fpga.html> (accessed 26.03.2019)

[27] FPGA & SoC. Available at: <https://www.microsemi.com/product-directory/1636-fpga-soc> (accessed 26.03.2019)

[28] Quartus II Handbook. Version 9.1. Altera, 2009. – 1820 p.

**Сведения об авторах.**

**Баркалов Александр Александрович,**



д.т.н., профессор Института информатики и электроники, Университет Зеленогурский (Польша).  
Область научных интересов – синтез цифровых систем; встроенные системы.  
E-mail: [A.Barkalov@iie.uz.zgora.pl](mailto:A.Barkalov@iie.uz.zgora.pl)

**Зеленева Ирина Яковлевна,**



к.т.н., доцент кафедры компьютерных систем и сетей Запорожского национального технического университета (ЗНТУ). Область научных интересов - проектирование устройств управления на ПЛИС; надежность компьютерных систем.  
E-mail: [irina.zeleneva@gmail.com](mailto:irina.zeleneva@gmail.com)

**Тигаренко Лариса Александровна,**



д.т.н., профессор Института информатики и электроники, Университет Зеленогурский (Польша). Профессор кафедры телекоммуникаций, Харьковский национальный университет радиотехники (Украина).  
Область научных интересов – синтез цифровых систем; антенны, системы телекоммуникации.

**Грушко Светлана Сергеевна,**



к.т.н., старший преподаватель кафедры компьютерных систем и сетей ЗНТУ. Область научных интересов – проектирование устройств управления на ПЛИС; интерфейсы компьютерных систем.  
E-mail: [grushko\\_ss@i.ua](mailto:grushko_ss@i.ua)