

СИСТЕМЫ И СРЕДСТВА ИНФОРМАТИКИ

**Научный журнал Российской академии наук
(издается под руководством Отделения нанотехнологий
и информационных технологий РАН)**

Издается с 1989 года
Журнал выходит ежеквартально

Учредители:

**Российская академия наук
Федеральный исследовательский центр
«Информатика и управление» Российской академии наук**

РЕДАКЦИОННЫЙ СОВЕТ

академик РАН И. А. Соколов — председатель Редакционного совета
академик РАН Г. И. Савин
академик РАН А. Л. Стемпковский
член-корреспондент РАН Ю. Б. Зубарев
профессор Ш. Долев (S. Dolev, Beer-Sheva, Israel)
профессор Ю. Кабанов (Yu. Kabanov, Besancon, France)
профессор М. Никулин (M. Nikulin, Bordeaux, France)
профессор В. Ротарь (V. Rotar, San-Diego, USA)
профессор М. Финкельштейн (M. Finkelstein, Rostok, Germany)

РЕДАКЦИОННАЯ КОЛЛЕГИЯ

академик РАН И. А. Соколов — главный редактор
профессор, д.ф.-м.н. С. Я. Шоргин — заместитель главного редактора
д.т.н. В. Н. Захаров проф., д.ф.-м.н. В. Ю. Королев
проф., д.т.н. В. Д. Ильин проф., д.г.-м.н. Р. Б. Сейфуль-Мулюков
проф., д.ф.-м.н. Л. А. Калининченко проф., д.т.н. И. Н. Синицын
д.т.н. В. А. Козмидади к.т.н. А. В. Филин
проф., д.т.н. К. К. Колин к.ф.-м.н. С. А. Христочевский

Редакция

профессор, д.г.-м.н. Р. Б. Сейфуль-Мулюков
к.ф.-м.н. Е. Н. Арутюнов
С. Н. Стригина (ответственный секретарь)

© Федеральный исследовательский центр «Информатика
и управление» Российской академии наук, 2015

Журнал входит в систему Российского индекса научного цитирования (РИНЦ):

http://elibrary.ru/title_about.asp?id=28980

Журнал включен в базу данных CrossRef (систему DOI — Digital Object Identifier),
в базу данных Ulrich's periodicals directory
и в информационную систему «Общероссийский математический портал Math-Net.Ru»

Журнал реферируется в «Реферативном журнале» ВИНТИ
и в системе Google Scholar

Журнал «Системы и средства информатики»
включен в формируемый Минобрнауки России Перечень рецензируемых научных изданий,
в которых должны быть опубликованы основные научные результаты диссертаций
на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук

<http://www.ipiran.ru/journal/collected>

СИСТЕМЫ И СРЕДСТВА ИНФОРМАТИКИ

Том 25 № 4 Год 2015

СОДЕРЖАНИЕ

Аналитическое моделирование нормальных процессов в стохастических системах со сложными дробно-рациональными нелинейностями И. Н. Сеницын, В. И. Сеницын, Э. Р. Корепанов	4
Анализ энергоэффективности вычислительного комплекса, моделируемого с помощью системы обслуживания с пороговым управлением и интенсивностями, зависящими от времени Р. В. Разумчик, А. И. Зейфман, А. В. Коротышева, Я. А. Сатин	19
Приближенная оптимизация стратегии распределения вычислительных ресурсов на примере системы андеррайтинга М. Г. Коновалов, Р. В. Разумчик	31
Метрическая классификация временных рядов с выравниванием относительно центроидов классов А. В. Гончаров, М. С. Попова, В. В. Стрижов	52
Архитектура стенда для экспериментального исследования моделей, алгоритмов и решений по обеспечению информационной безопасности в облачных вычислительных средах А. А. Грушо, М. И. Забейхайло, А. А. Зацаринный, А. В. Николаев, В. О. Писковский	65
Аппаратно-программное моделирование и тестирование рекуррентного операционного устройства Д. В. Хилько, Ю. А. Степченков, Ю. Г. Дьяченко, Ю. И. Шикунев, Н. В. Морозов	78
Исследование особенностей сигналов миограммы Т. В. Захарова, В. Ю. Королев, А. А. Щемирова	91
Метод синхронизации сигналов магнитоэнцефалограмм и миограмм Е. В. Щенявская, Т. В. Захарова	101

АППАРАТНО-ПРОГРАММНОЕ МОДЕЛИРОВАНИЕ И ТЕСТИРОВАНИЕ РЕКУРРЕНТНОГО ОПЕРАЦИОННОГО УСТРОЙСТВА*

*Д. В. Хилько¹, Ю. А. Степченков², Ю. Г. Дьяченко³,
Ю. И. Шикунов⁴, Н. В. Морозов⁵*

Аннотация: Рассматриваются основные проблемы моделирования и отладки операционного устройства, основанного на принципах нетрадиционной потоковой рекуррентной архитектуры. Приводится краткое описание методологии аппаратно-программного моделирования Hardware in the Loop, также рассматривается возможность ее применения для разработки рекуррентного операционного устройства (РОУ). Предложен подход и методика аппаратно-программного моделирования и тестирования устройства на основе новой архитектуры. Описываются разработанные программная и аппаратная модели устройства, средства моделирования и отладки, а также результат их интеграции в единую среду. Вводится понятие целевой платформы моделирования, названной GAROS IDE. Приводится описание ее основных компонент, функциональных возможностей и перспектив развития. Представлены результаты апробации платформы на одной из подзадач распознавания изолированных слов — алгоритме Rasta-фильтрации.

Ключевые слова: моделирование; тестирование; рекурсивность; рекуррентность; потоковая архитектура; методология программирования

DOI: 10.14357/08696527150406

1 Введение

Коллективом отдела «Архитектуры перспективных компьютерных систем» ИПИ РАН ведется разработка концепций принципиально новой архитектуры,

*Исследование выполнено при частичной финансовой поддержке РФФИ (проект 13-07-12068).

¹Институт проблем информатики Федерального исследовательского центра «Информатика и управление» Российской академии наук, dhillko@yandex.ru

²Институт проблем информатики Федерального исследовательского центра «Информатика и управление» Российской академии наук, ystepchenkov@ipiran.ru

³Институт проблем информатики Федерального исследовательского центра «Информатика и управление» Российской академии наук, diaura@mail.ru

⁴Институт проблем информатики Федерального исследовательского центра «Информатика и управление» Российской академии наук, yshikunov@yandex.ru

⁵Институт проблем информатики Федерального исследовательского центра «Информатика и управление» Российской академии наук, nmozov@ipiran.ru

а также вычислительного устройства на ее основе. Предлагаемая нетрадиционная архитектура предназначена для реализации параллельных вычислений ограниченной размерности в области цифровой обработки сигналов (ЦОС).

В настоящее время разрабатывается опытный образец, названный рекуррентным обработчиком сигналов (РОС), исполняемый в гибридном, двухуровневом варианте с ведущим фон-неймановским процессором на управляющем (верхнем) уровне (УУ) и рядом потоковых процессоров на нижнем уровне — РОУ [1]. Данная реализация архитектуры была названа «Гибридная архитектура рекуррентного обработчика сигналов» (ГАРОС). В качестве демонстрационной задачи для ГАРОС была выбрана задача распознавания изолированных слов ввиду высокой степени готовности ее математического обеспечения и ее алгоритмов, воспроизводящих широкий спектр типовых задач ЦОС.

Предварительным этапом разработки любой технически сложной системы является создание ряда моделей: математических, программных и аппаратных, средствами которых осуществляется отладка основных механизмов функционирования системы. В ходе проектирования указанных моделей был выявлен ряд проблем, результат решения которых был оформлен в виде *методики (подхода) аппаратно-программного моделирования рекуррентного обработчика сигналов*. Настоящая статья посвящена описанию выявленных проблем, разработанного подхода и результатов его апробации.

2 Проблемы моделирования ГАРОС. Методика аппаратно-программного моделирования и отладки

Существует несколько основных подходов к разработке систем ЦОС. В работе [2] автор приводит достаточно подробное описание данных подходов, а также приводит ключевые аспекты метода аппаратно-программного моделирования Hardware in the Loop (HIL). «Методология HIL обеспечивает промежуточный уровень контроля разрабатываемых систем — между программным обеспечением моделирования и тестированием оборудования проекта», — пишет автор. Согласно методологии HIL все основные этапы разработки систем ЦОС интегрируются в единый итерационный цикл проектирования. Другими словами, программные и аппаратные модели системы функционируют и отлаживаются одновременно.

В рамках HIL используются такие программные средства, как MATLAB/Simulink, Altera DSP Builder и др. С точки зрения разработчиков РОС приведенная выше методология HIL оказалась привлекательной для внедрения. Однако был обнаружен целый ряд проблем, препятствующих приложению HIL в полном объеме. Среди этих проблем можно выделить следующие:

- (1) разработчикам не удалось реализовать некоторые положения, лежащие в основе ГАРОС, в частности рекуррентность средствами среды MATLAB/Simulink;

- (2) построение VHDL (VHSIC (Very High Speed Integrated Circuits) Hardware Description Language) модели на основе математической трудно или невозможно завершить ввиду проблемы 1;
- (3) уникальность представления данных в ГАРОС в виде «элемента самоопределяющихся данных» сильно затрудняет или делает невозможным трансляцию в язык C++, используемый в системе MATLAB в качестве основного;
- (4) результат трансляции математической модели в VHDL и его последующий перенос в программируемую логическую интегральную схему (ПЛИС) не удовлетворяют физическим ограничениям микросхемы;
- (5) отсутствие средств разработки и отладки программного обеспечения для РОС привело к необходимости ручного проектирования и тестирования, что сильно затрудняло автоматизацию процесса моделирования.

Были выявлены и другие проблемы моделирования, но указанные являются основными. В процессе поиска решения данных проблем был выработан подход к организации процесса моделирования и отладки ГАРОС, определены компоненты аппаратно-программных средств, которые необходимо создать, осуществлена их разработка. Предлагаемый подход заключается в *разработке целевой платформы* для проверки работоспособности, оценки производительности оборудования системы ГАРОС в реальных условиях, отладки специализированного программного обеспечения (ПО) для ГАРОС.

Указанная целевая платформа должна включать в себя: средства математического моделирования ГАРОС (имитационную модель, написанную на языке высокого уровня); средства аппаратного моделирования (VHDL-модель, исполняемую в среде ModelSim); опытный образец на ПЛИС; средства разработки ПО, предназначенного для функционирования в среде ГАРОС.

С учетом выбранного подхода по аналогии с методологией НИЛ была разработана *методика аппаратно-программного моделирования и отладки ГАРОС*:

- (1) разработка с нуля либо усовершенствование математической (имитационной) модели, в качестве платформы разработки был выбран язык программирования C#;
- (2) разработка с нуля либо усовершенствование VHDL-модели средствами программных продуктов Altera;
- (3) разработка тестов проверки соответствия имитационной и VHDL-модели и их последующее моделирование;
- (4) реализация проекта на базе ПЛИС компании Altera;
- (5) отладка полученной системы;
- (6) интеграция этапов 1–5 в единый итерационный цикл проектирования;
- (7) создание средств разработки ПО, а также единой среды моделирования, интегрирующей в себе вышеуказанные модели и тесты;

- (8) итерационная разработка и отладка ГАРОС средствами единой среды моделирования.

Таким образом, была заложена основа интегрированной среды разработки для ГАРОС, названная авторами в работе [3] GAROS IDE.

3 Описание среды GAROS IDE и ее ключевых компонентов

В процессе поиска решения проблем программируемости ГАРОС была разработана методология программирования, названная авторами в работе [4] *рекуррентно-поточковой методологией программирования*. В рамках данной методологии удалось формализовать процесс разработки программного обеспечения, определить его ключевые этапы, а также подходы к организации отладки разрабатываемых программ. Формализованное представление процесса программирования ГАРОС послужило фундаментом для разработки прототипа интегрированной среды программирования GAROS IDE, функциональные возможности компонентов которого охватывают все введенные в рамках методологии этапы.

В настоящее время ведутся работы по развитию и усовершенствованию интегрированной среды разработки GAROS IDE, назначение которой эволюционировало от простой среды программирования до полноценного комплекса моделирования, программирования и отладки. В работе [5] приводится общая структура прототипа этой среды, ее архитектура и описание компонентов. На рис. 1 представлена структура GAROS IDE.

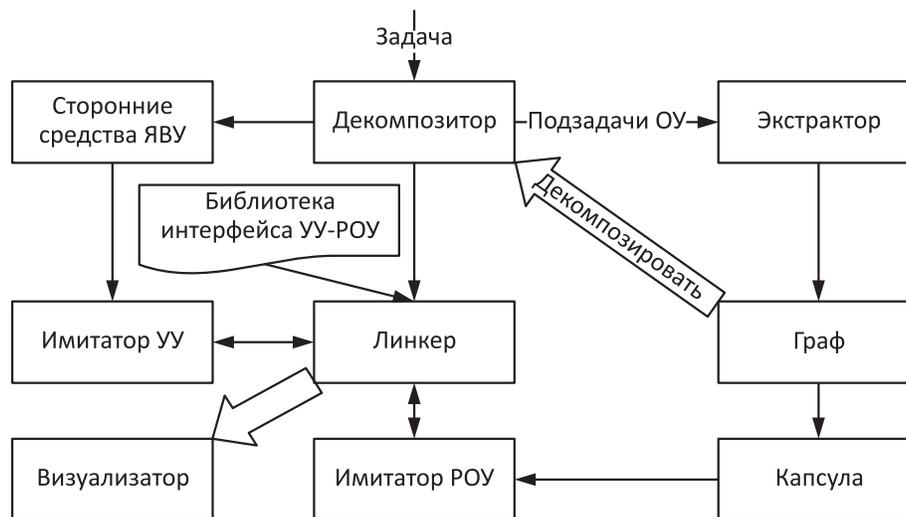


Рис. 1 Обобщенная структура GAROS IDE

Компонент «Декомпозитор» предназначен для декомпозиции решаемой задачи.

Сторонние средства ЯВУ (языков высокого уровня) — набор программных средств для разработки традиционного ПО: для реализации программы УУ.

Компонент «Экстрактор» предназначен для извлечения графа вычислительного процесса (ВП) из описания подзадачи на языке высокого уровня.

Компонент «Граф» — многофункциональный компонент, предоставляющий широкий спектр возможностей для работы с визуальными графами. В случае необходимости (в соответствии с рекуррентно-поточковой методологией) подзадача может быть передана в «Декомпозитор» для повторной декомпозиции.

Компонент «Капсула» предназначен для работы со специализированными программами, функционирующими в РОУ, — капсулами.

Компонент «Имитатор УУ» предназначен для интерпретирования программы УУ. Может работать в двух режимах: полной интерпретации программы и интерпретации при помощи программы ПРАПОР [6].

Компонент «Имитатор РОУ» предназначен для интерпретирования программы операционного уровня. Может работать в четырех различных режимах: интерпретация в режиме ПРАПОР, имитационное моделирование при помощи программы СИМПРА (программа реализует имитационную модель РОУ) [7], моделирование при помощи VHDL-модели, интерпретация на опытном образце.

Компонент «Линкер» предназначен для организации взаимодействия между имитаторами УУ и РОУ с учетом информации, хранящейся в картах памяти и данных. Осуществляет сборку и интерпретацию всей задачи в целом.

Компонент «Визуализатор» предназначен для отображения результатов моделирования и отладки ПО.

С точки зрения описываемого в статье подхода и методики аппаратно-программного моделирования и отладки ГАРОС наиболее важным компонентом является «Имитатор РОУ», включающий в себя как имитационную модель РОУ, так и аппаратную VHDL-модель РОУ. В работе [1] приводится подробное описание VHDL-модели РОУ, а в работе [8] — описание имитационной модели. Далее приводится их краткое описание.

4 Имитационная модель рекуррентного операционного устройства

Имитационная модель включает в себя следующие компоненты:

- (1) буферная память (БП) — компонент, логически разделенный на два блока (физически — единое целое); адресуется двумя индексными регистрами IR0 и IR1;
- (2) интерфейсные регистры IR0, IR1, F0, F1, D0, D1, предназначенные для моделирования взаимодействия между УУ, БП и РОУ, а также для поддержки библиотеки интерфейса;

- (3) вспомогательное устройство управления, организующее многократное исполнение капсулы и ее фрагментов;
- (4) распределитель — основной компонент, отвечающий за обработку потока операндов и их рассылку по секциям назначения;

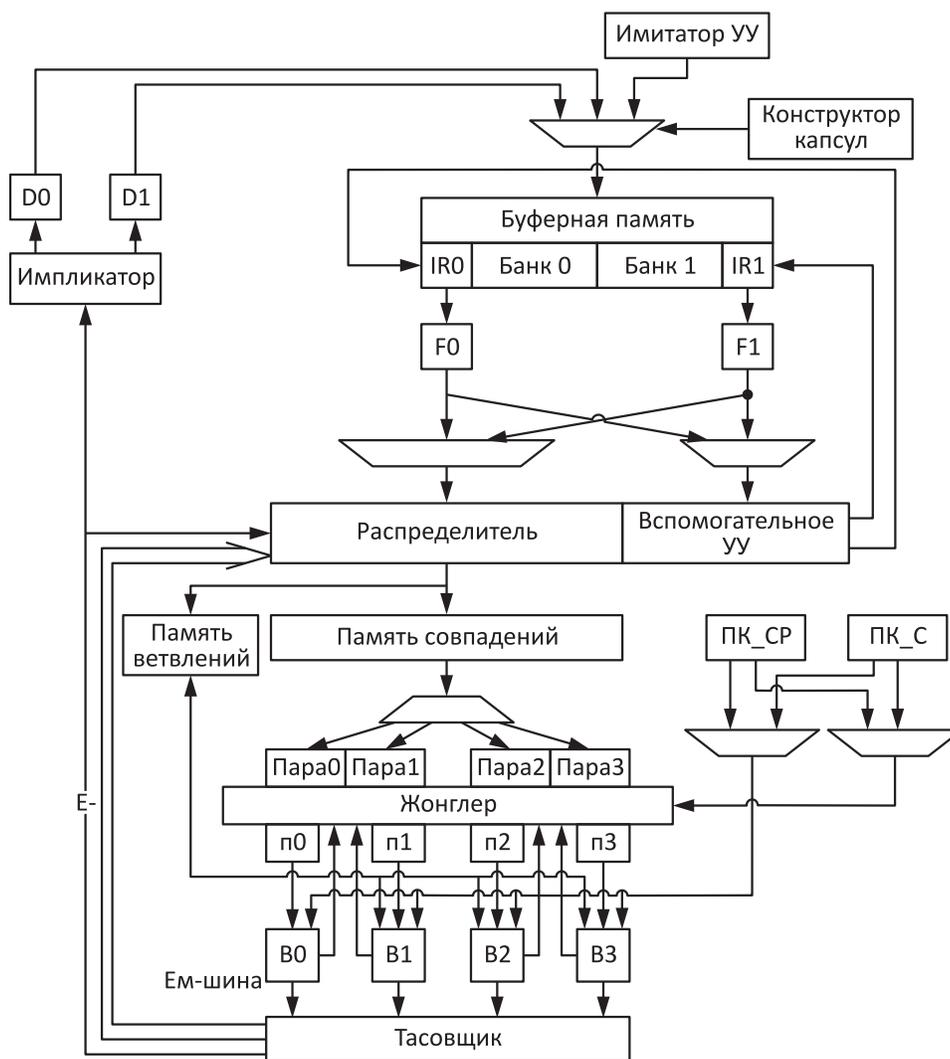


Рис. 2 Архитектура имитационной модели

- (5) импликатор — компонент обработки выходных результатов, отвечающий за своевременный отбор выходных данных потока промежуточных операндов и запись их в БП;
- (6) память констант (секционная и секционная-регистровая) — компонент, предназначенный для хранения постоянных данных;
- (7) память совпадений — упрощенный аналог ассоциативной;
- (8) память переходов, хранящая условия переходов и функциональные поля операнда — результата перехода, структурно аналогичная памяти совпадений;
- (9) жонглер, предназначенный для анализа пар операндов и принятия решения о характере протекания вычислений в вычислителе;
- (10) вычислитель (4 шт.) — совокупность вычислительных блоков (умножитель, АЛУ16 и АУ40), поддерживающая разные режимы суперскалярной работы;
- (11) шины E- и Em-магистралы пересылки промежуточных данных;
- (12) конструктор капсул.

Архитектура имитационной модели представлена на рис. 2.

VHDL-модель рекуррентного операционного устройства

На рис. 3 приведена структура VHDL-модели ГАРОС. Ключевые компоненты соответствуют аналогичным в имитационной модели.

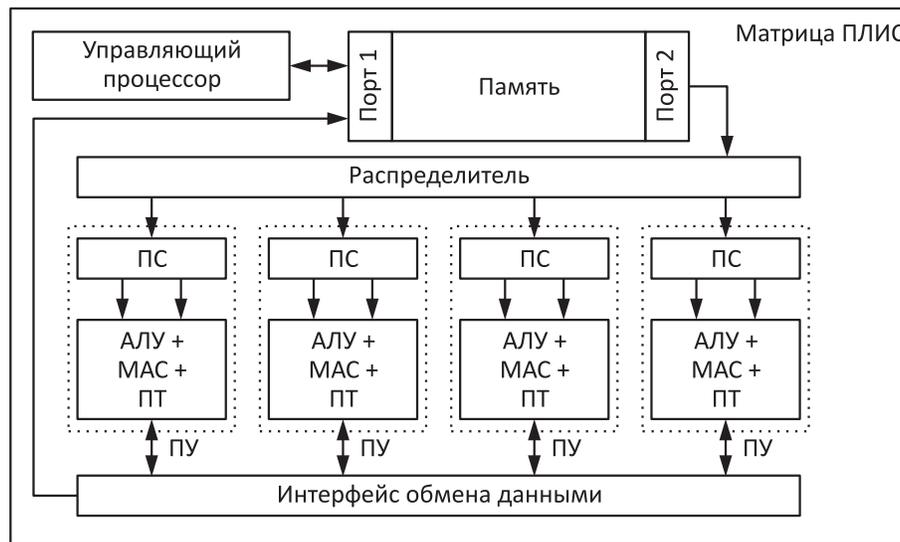


Рис. 3 Структура VHDL-модели ГАРОС

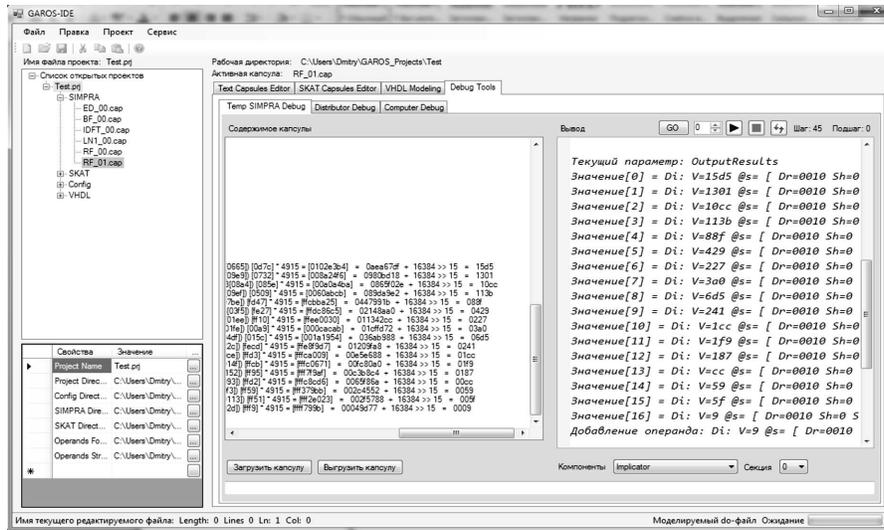


Рис. 5 Результаты имитационного моделирования

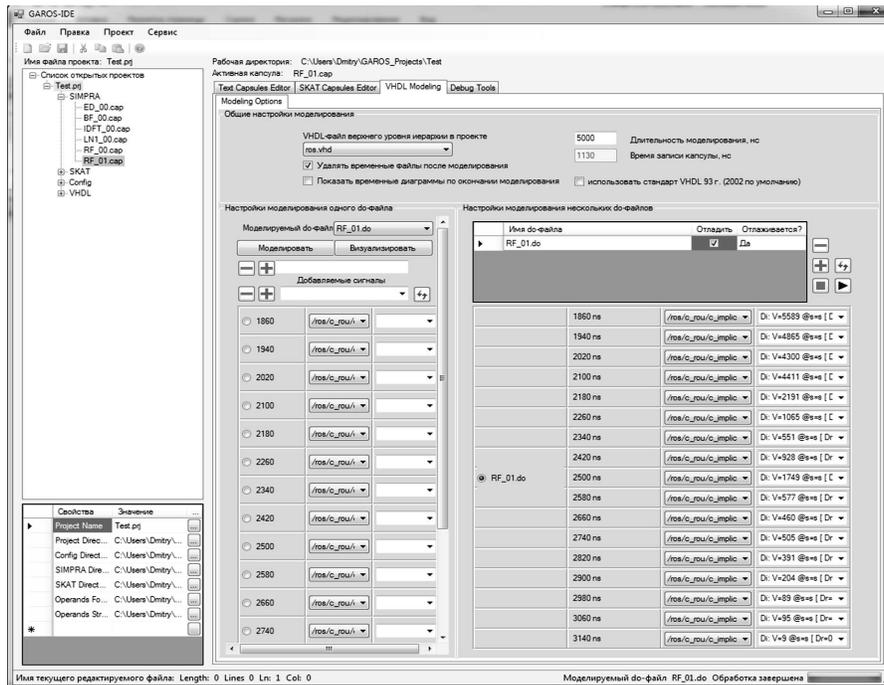


Рис. 6 Результаты VHDL-моделирования

Далее полученная капсула имплементируется в Си-программу распознавателя изолированных слов с целью извлечения всех промежуточных и выходных данных, необходимых для проверки корректности функционирования как капсулы, так и имитационной модели. Извлеченные данные могут быть вставлены в капсулу в виде комментариев для удобства. Следующим шагом является моделирование и отладка капсулы средствами имитационной модели РОУ. На рис. 5 представлен снимок экрана результатов моделирования, демонстрирующий совпадение всех выходных результатов с ожидаемыми.

Одновременно с имитационным испытанием можно запустить VHDL-модель средствами ModelSim, так как этот процесс в GAROS IDE выполняется в фоновом потоке. Результаты моделирования могут наблюдаться тремя способами: в виде временных диаграмм моделирования (наиболее детализированное представление); в виде SVG (Scalable Vector Graphics) диаграмм визуализации результатов (наиболее удобный для отладки); в виде набора наблюдаемых сигналов (для отлаженных капсул). На рис. 6 приводится снимок экрана результатов VHDL-моделирования капсулы RASTA-фильтрации. Представленные данные (Di) имеют десятичный формат. При переводе их в шестнадцатеричный формат нетрудно убедиться, что данные совпадают.

6 Заключение

В ходе работ по созданию вычислительного устройства, основанного на рекуррентно-поточковой модели вычислений, был разработан эффективный подход, а также методика аппаратно-программного моделирования и тестирования ГАРОС. Для реализации предложенного подхода были спроектированы аппаратно-программные средства, сформировавшие мощную платформу GAROS IDE для выполнения широкого спектра задач: от разработки ПО и моделирования до выработки способа расширения функциональных возможностей ГАРОС.

Таким образом, можно говорить, что платформа GAROS IDE, реализующая аппаратно-программный подход к моделированию и отладке ГАРОС, является эффективным средством поддержки разработки и поэтому требует дальнейшего развития и совершенствования функциональных возможностей.

Литература

1. *Степченко Ю. А., Петрухин В. С.* Особенности гибридного варианта реализации на ПЛИС рекуррентного обработчика сигналов // Системы и средства информатики, 2008. Доп. вып. С. 118–129.
2. *Вычужанин В.* Методика аппаратно-программного моделирования и тестирования проектируемых систем // Современная электроника, 2015. № 2. С. 62–67.
3. *Хилько Д. В., Степченко Ю. А., Шикун Ю. И.* Инструментальная среда проектирования ПО для гибридной архитектуры рекуррентного обработчика сигналов

- (GAROS IDE). Свидетельство о государственной регистрации программы для ЭВМ № 2015614004.
4. Хилько Д. В., Степченков Ю. А. Теоретические аспекты разработки методологии программирования рекуррентной архитектуры // Системы и средства информатики, 2013. Т. 23. № 2. С. 133–156.
 5. Хилько Д. В., Шихунов Ю. И. Разработка инструментальной среды проектирования программного обеспечения для рекуррентно-поточковой модели вычислений // 4-я школа молодых ученых ИПИ РАН: Сб. докл. — М.: ИПИ РАН, 2013. С. 65–77.
 6. Хилько Д. В., Степченков Ю. А., Шнейдер А. Ю. Программа обработки результатов моделирования потоковой рекуррентной архитектуры (ПРАПОР). Свидетельство о государственной регистрации программы для ЭВМ № 2013610199.
 7. Хилько Д. В., Степченков Ю. А. Средства имитационного моделирования потоковой рекуррентной архитектуры (СИМПРА). Версия 2. Свидетельство о государственной регистрации программы для ЭВМ № 2014610123.
 8. Хилько Д. В., Степченков Ю. А., Шихунов Ю. И. Средства имитационного моделирования многоядерной потоковой рекуррентной архитектуры // Многоядерные процессоры, параллельное программирование, ПЛИС, системы обработки сигналов: Сб. научных статей II Всеросс. науч.-практич. конф. — Барнаул: АлтГУ, 2014. С. 58–69.

Поступила в редакцию 30.09.15

HARDWARE AND SOFTWARE MODELING AND TESTING OF THE RECURRENT OPERATIONAL DEVICE

*D. V. Khilko, Yu. A. Stepchenkov, Yu. G. Diachenko, Yu. I. Shikunov,
and N. V. Morozov*

Institute of Informatics Problems, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation

Abstract: The paper considers the main problems of modeling and debugging of the operational device based on the principles of the nonconventional data-flow recurrent architecture. The short description of the methodology of hardware and software modeling of Hardware in the Loop is provided; also, the possibility of its application for developing a recurrent operational device is considered. An approach and a technique of hardware and software modeling and testing of devices on the basis of the new architecture are suggested. The paper describes the developed program and models of the device’s hardware, simulating and debugging tools, and the result of their integration into the uniform environment. The concept of the target platform of modeling called GAROS IDE is introduced. The descriptions of its main components, functionality, and prospects of development are presented. The results of approbation of this platform on one of the subtasks of isolated word recognition (the algorithm of Rasta-filtration) are reported.

Keywords: modeling; testing; recursiveness; recurrence property; data-flow architecture; programming methodology

DOI: 10.14357/08696527150406

Acknowledgments

The work was partially supported by the Russian Foundation for Basic Research (project No. 13-07-12068).

References

1. Stepchenkov, Yu. A., and V. S. Petrukhin. 2008. Osobennosti gibridnogo varianta realizatsii na PLIS rekurrentnogo obrabotchika signalov [Features of the hybrid embodiment of the recurrent signal processor on FPGA]. *Sistemy i Sredstva Informatiki — Systems and Means of Informatics*. Add. iss.:118–129.
2. Vychuzhanin, V. 2015. Metodika apparatno-programmnogo modelirovaniya i testirovaniya proektiruemykh sistem [Methods of hardware and software simulation and testing the designed system]. *Sovremennaya Elektronika* [Modern Electronics] 2:62–67.
3. Khilko, D. V., Yu. A. Stepchenkov, and Yu. I. Shikunov. Instrumental'naya sreda proektirovaniya PO dlya gibridnoy arkhitektury rekurrentnogo obrabotchika signalov (GAROS IDE) [Instrumental development environment software for the hybrid architecture of recurrent signal processor (GAROS IDE)]. Certificate of state registration of computer program No. 2015614004.
4. Khilko, D. V., and Yu. A. Stepchenkov. 2013. Teoreticheskie aspekty razrabotki metodologii programirovaniya rekurrentnoy arkhitektury [Theoretical aspects of programming methodology development of recurrent architecture]. *Sistemy i Sredstva Informatiki — Systems and Means of Informatics* 23(2):133–156.
5. Khilko, D. V., and Yu. I. Shikunov. 2013. Razrabotka instrumental'noy sredy proektirovaniya programmnogo obespecheniya dlya rekurrentno-potokovoy modeli vychisleniy [Development IDE software recursively-flow model calculations]. *4-ya shkola molodykh uchennykh IPI RAN* [4th School of Young Scientists of IPI RAN]. 65–77.
6. Khilko, D. V., Yu. A. Stepchenkov, and A. Yu. Shneyder. Programma obrabotki rezul'tatov modelirovaniya potokovoy rekurrentnoy arkhitektury (PRAPOR) [Processing simulation results program of data-flow recurrent architecture (PRAPOR)]. Certificate of state registration of computer program No. 2013610199.
7. Khilko, D. V., and Yu. A. Stepchenkov. Sredstva imitatsionnogo modelirovaniya potokovoy rekurrentnoy arkhitektury (SIMPRA). Versiya 2 [Simulation tools of data-flow recurrent architecture (SIMPRA). Version 2]. Certificate of state registration of computer program No. 2014610123.
8. Khilko, D. V., Yu. A. Stepchenkov, and Yu. I. Shikunov. 2014. Sredstva imitatsionnogo modelirovaniya mnogoyadernoy potokovoy rekurrentnoy arkhitektury [Simulation tools for multicore streaming recurrent architecture]. *Mnogoyadernye protsessy, parallel'noe programirovanie, PLIS, sistemy obrabotki signalov: Sb. nauchnykh statey Vserossiyskoy nauchno-praktich. konf.* [Multicore processes, parallel programming, FPGA, signal processing system: A collection of scientific papers of All-Russian Scientific-Practical Conference]. Barnaul. 58–69.

Received September 30, 2015

Contributors

Khilko Dmitri V. (b. 1987) — scientist, Institute of Informatics Problems, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; dhilko@yandex.ru

Stepchenkov Yuri A. (b. 1951) — Candidate of Science (PhD) in technology, Head of Department, Institute of Informatics Problems, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; ystepchenkov@ipiran.ru

Diachenko Yuri G. (b. 1958) — Candidate of Science (PhD) in technology, senior scientist, Institute of Informatics Problems, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, Moscow 119333, Russian Federation; diaura@mail.ru

Shikunov Yuri I. (b. 1995) — apprentice researcher, Institute of Informatics Problems, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; yishikunov@yandex.ru

Morozov Nikolay V. (b. 1956) — senior scientist, Institute of Informatics Problems, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, Moscow 119333, Russian Federation; nmorozov@ipiran.ru