

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
Институт проблем информатики 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 И. Ушаков (I. Ushakov, 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.т.н. В. Д. Ильин
проф., д.ф.-м.н. Л. А. Калининченко
д.т.н. В. А. Козмидиади
проф., д.т.н. К. К. Колин
проф., д.ф.-м.н. В. Ю. Королев
проф., д.ф.-м.н. А. В. Печинкин
проф., д.г.-м.н. Р. Б. Сейфуль-Муллюков
проф., д.т.н. И. Н. Сеницын
к.т.н. А. В. Филин
к.ф.-м.н. С. А. Христочевский

Редакция

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

© Институт проблем информатики Российской академии наук, 2014

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» ВИНИТИ
и в системе Google Scholar

Журнал «Системы и средства информатики»
включен в «Перечень российских рецензируемых журналов,
в которых должны быть опубликованы основные научные результаты диссертаций
на соискание ученых степеней доктора и кандидата наук», утвержденный ВАК

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

СИСТЕМЫ И СРЕДСТВА ИНФОРМАТИКИ

Том 24 № 2 Год 2014

СОДЕРЖАНИЕ

Методы и средства оптимального планирования параметров процессов в системах послепродажного обслуживания изделий наукоемкой продукции И. Н. Синицын, А. С. Шаламов, И. В. Сергеев, Э. Р. Корепанов, В. В. Белоусов, Т. С. Гумникова, В. С. Шоргин, Е. С. Агафонов	4
Согласование агрегированных и детализированных прогнозов при решении задач непараметрического прогнозирования М. М. Стенина, В. В. Стрижов	23
Оценивание эффективной пропускной способности узла в инфокоммуникационной тандемной сети А. В. Бородина, Е. В. Морозов	37
Инструменты для системной верификации рекуррентного обработчика сигналов В. С. Петрухин, Д. Ю. Степченков, Н. В. Морозов, Ю. А. Степченков	55
Создание высокопроизводительного генератора нагрузки для проверки систем высокочастотной торговли Д. К. Гурьев, М. А. Гай, И. Л. Иткин, А. А. Терентьев	67
Использование инструментов для пассивного тестирования при сертификации клиентов трейдинговых систем А. Н. Алексеенко, А. А. Аверина, Д. С. Шаров, П. А. Проценко, И. Л. Иткин	83
Технология анализа исходного кода программного обеспечения и частичных спецификаций для автоматизированной генерации тестов А. А. Андрианова, В. М. Ицыксон	99

ИНСТРУМЕНТЫ ДЛЯ СИСТЕМНОЙ ВЕРИФИКАЦИИ РЕКУРРЕНТНОГО ОБРАБОТЧИКА СИГНАЛОВ*

В. С. Петрухин¹, Д. Ю. Степченков², Н. В. Морозов³, Ю. А. Степченков⁴

Аннотация: Рассмотрена процедура выбора и разработки комплекса программных и аппаратных инструментов для проектирования и отладки нетрадиционного цифрового сигнального процессора на базе рекуррентно-динамической потоковой гибридной архитектуры — рекуррентного обработчика сигналов (РОС). Экспериментальный характер обрабатываемой архитектуры РОС и необходимость наличия отработанного управляющего процессора предопределили выбор в качестве элементной базы его реализации ПЛИС (программируемые логические интегральные схемы) семейства Cyclone V фирмы Альтера и, соответственно, среды разработки — Quartus II. Мощные инструменты верификации, входящие в состав этой среды, позволяют сократить время получения готового проекта и существенно уменьшить аппаратные затраты. На основе сравнительного анализа и установленных критериев определен состав и предложена оптимальная структура аппаратных инструментов отладки РОС, позволяющая существенно упростить процесс верификации и отладки РОС в реальном аппаратном окружении.

Ключевые слова: отладочные средства; потоковая архитектура; верификация

DOI: 10.14357/08696527140204

1 Введение

В ИПИ РАН ведутся работы по созданию вычислителя нетрадиционной рекуррентной архитектуры, предназначенного для реализации параллельных вычислений в области цифровой обработки сигналов [1]. Для апробации архитектурных решений и исследования функционирования разрабатываемого РОС он реализуется на ПЛИС. Завершаемым этапом процесса разработки РОС с потоковой архитектурой [2] является верификация проекта РОС на физической модели или системная отладка, т. е. отладка с использованием реальных аппаратных и программных инструментов. Функционирование аппаратных и программных инструментов при этом осуществляется в реальном масштабе времени, что существенно повышает трудоемкость отладки.

* Работа выполнена при частичной финансовой поддержке по Программам фундаментальных исследований ОНИТ РАН за 2013 г. (проект 1.5) и Президиума РАН (проект 16).

¹ Институт проблем информатики Российской академии наук, cokrat2@rambler.ru

² Институт проблем информатики Российской академии наук, stepchenkov@mail.ru

³ Институт проблем информатики Российской академии наук, nmorozov@ipiran.ru

⁴ Институт проблем информатики Российской академии наук, ystepchenkov@ipiran.ru

Такой вариант отладки позволяет проверить работоспособность РОС при подаче на него реальных сигналов и в условиях помех, а также устранить нестыковки интерфейсов различных частей РОС и возможные ошибки разводки печатной платы. Довольно часто исследования непосредственно на реальной модели являются единственным средством верификации системы (особенно для столь сложных проектов), поскольку имеются причины, ограничивающие возможности моделирования (симуляции). Во-первых, процесс моделирования может занимать достаточно большое время, так как в модельном времени исследование ведется на несколько порядков медленнее. Во-вторых, достоверность программного моделирования ограничивается соответствием модели входного воздействия реальным условиям.

Можно выделить два основных метода внутрисхемной отладки ПЛИС [3]:

- (1) применение встроенных в ПЛИС отладочных инструментов на основе JTAG (Joint Test Action Group);
- (2) использование внешнего контрольно-измерительного оборудования: осциллографов смешанных сигналов и логических анализаторов. Использование внешнего оборудования связано со значительными материальными затратами.

В качестве элементной базы реализации РОС выбраны ПЛИС (семейство Cyclone V фирмы Альтера [4]), а в качестве интегрированной среды разработки — система автоматизированного проектирования (САПР) Quartus II, содержащая готовые компоненты для проведения проектирования и отладки систем. Использование таких сред позволяет значительно сократить время разработки и уменьшить временные затраты на отладку.

Апробация новых архитектурных решений проводилась в условиях, связанных со значительными временными и финансовыми ограничениями, поэтому основными критериями при выборе состава отладочных средств было максимальное использование стандартных средств отладки, реализованных в Quartus II, и минимальное привлечение дополнительной аппаратуры. В соответствии с выбранными критериями проведен анализ встроенных отладочных средств Quartus II на предмет их использования для отладки устройства РОС.

2 Средства отладки Quartus II

Среда разработки Quartus II имеет следующие встроенные средства отладки [5, 6] (здесь и далее в разд. 2 используется переработанный материал из соответствующих пунктов данных источников в преломлении к поставленным целям и задачам):

- редактор отладочных выводов (SignalProbe Pins);
- редактор интерфейса для внешнего логического анализатора (Logic Analyzer Interface Editor);

- редактор содержимого внутрисхемной памяти (In-System Memory Content Editor);
- встраиваемый логический анализатор SignalTap II (SignalTap II Logic Analyzer);
- редактор контрольных сигналов внутри ПЛИС (In-System Sources and Probes Editor);
- виртуальный JTAG (Virtual JTAG);
- отладочные средства процессора Nios II.

2.1 Отладочные выводы SignalProbe Pins

Отладочные выводы SignalProbe Pins — простейшее средство отладки. Используя этот механизм, можно сделать доступными для наблюдения внутренние сигналы проекта. Для этого они назначаются на не занятые в проекте выводы ПЛИС. Редактор позволяет выбрать соответствие между внутренними сигналами проекта и внешними выводами, назначив их сигналы на не занятые в проекте выводы и соответствующим образом их переименовав.

Простота использования SignalProbe Pins и то, что при использовании отладочных выводов не тратится логический ресурс ПЛИС, являются его основными достоинствами. Нет необходимости в дополнительном оборудовании, благодаря чему сохраняются характеристики пользовательского проекта и отсутствуют крупные дополнительные аппаратные и временные затраты. Это делает SignalProbe Pins приемлемым для отладки отдельных блоков схемы. Однако для комплексной отладки РОС он не подходит, потому что не позволяет считывать содержимое памяти. Кроме того, может не хватить неиспользованных выводов.

2.2 Интерфейс для внешнего логического анализатора

При помощи набора управляемых мультиплексоров на вход внешнего логического анализатора отбираются нужные сигналы анализируемого проекта. При этом выходы мультиплексоров назначаются на не занятые в проекте выводы ПЛИС. С помощью JTAG-интерфейса тестируемое устройство подключается к компьютеру. Пользователю предоставляется возможность управления мультиплексорами за счет редактора LAI (Logic Analyzer Interface).

После создания интерфейса для внешнего логического анализатора требуются повторные компиляции проекта, учитывающие изменения в коммутации, и загрузка на ПЛИС. Смена групп отслеживаемых сигналов осуществляется «на лету», что позволяет обойтись в дальнейшем без перезагрузок и перекомпиляций.

Интерфейс LAI может размещаться в микросхемах отладочной платы; внутренние блоки оперативной памяти ПЛИС для его использования не нужны. Интерфейс LAI позволяет решить проблему недостаточного количества внешних выводов, присущую средству SignalProbe Pins, за счет аппаратных затрат

на реализацию интерфейса и привлечения внешнего оборудования. Привлечение дорогостоящего внешнего логического анализатора не соответствует сформулированным критериям выбора средств для отладки РОС. Помимо этого, использование внешнего анализатора может вызвать изменение характеристик пользовательского проекта. Соответственно, использование при отладке РОС интерфейса для внешнего логического анализатора нежелательно.

2.3 Редактор содержимого внутрисхемной памяти

Редактор содержимого внутрисхемной памяти (In-System Memory Content Editor) позволяет анализировать данные, записанные в блоках памяти ПЛИС. Это упрощает отладку РОС, поскольку содержимое блоков памяти РОС модифицируется в процессе работы. Симулятор пакета Quartus II не позволяет решить эту задачу на имитационной модели, поэтому особенно важно решить ее на физической модели. Так как доступ к данным в памяти в процессе работы осуществляется через дополнительный порт, создаваемый редактором содержимого памяти, то возможность анализа доступна только для модулей однопортовой памяти. Связь с ПЛИС осуществляется с помощью JTAG. Соответствующее конфигурирование позволяет редактору содержимого памяти выбирать нужную область памяти с заданными именами и характеристиками, выполнять однократное и циклическое чтение с соответствующим отображением для пользователя и даже редактирование.

Редактор содержимого внутрисхемной памяти — полезный механизм, повышающий скорость и удобство отладки. Его отличительной особенностью является возможность редактирования памяти без дополнительной переконфигурации ПЛИС, отсутствие требования к использованию дополнительной аппаратуры и к ресурсам ПЛИС. Использование такого редактора при отладке РОС перспективно.

2.4 Встраиваемый логический анализатор SignalTap II

Встраиваемый логический анализатор SignalTap II позволяет:

- управлять записью в память на логическом ядре ПЛИС;
- использовать компьютер в качестве средства отображения и анализа ПЛИС;
- иметь память на блоках встроенного оперативного запоминающего устройства (ОЗУ) для записи отсчетов в реальном масштабе времени.

Подключение к внутренним сигналам и выводам микросхемы ПЛИС осуществляется стандартными средствами САПР и вносит минимальные искажения в наблюдаемые сигналы. К SignalTap II, представляющему собой параметризируемую мегафункцию, возможен доступ как с помощью редактора параметризируемых модулей MegaWizard, так и через специализированный пользовательский интерфейс. Соответственно, возможности SignalTap II чрезвычайно широки. Он позволяет выбирать сигналы проекта для наблюдения, осуществлять запись

логических состояний этих сигналов, выполнять совместную работу с САПР Quartus II, подключаясь к ней через JTAG-интерфейс, захватывать наблюдаемые сигналы в реальном времени на частотах свыше 300 МГц.

SignalTap II имеет два режима формирования сигнала захвата: последовательный и режим формирования условий на основе машины состояний. Первый режим (sequential) позволяет использовать и комбинировать стандартные условия захвата. Второй режим (state-based) дает возможность создавать особые условия захвата сигнала для наблюдения, например: комбинацию нескольких событий, появление некоторого события несколько раз и т. п.

Различные настройки позволяют просмотреть данные, записанные до, во время или после захвата сигнала, что существенно упрощает процесс отладки. К сожалению, наблюдение некоторых внутренних сигналов проекта невозможно.

К основным достоинствам SignalTap II можно отнести возможность событийной отладки, создание сложной системы условий, минимально вносимые искажения в наблюдаемые сигналы, возможность анализа изменений сигналов до, во время и после захвата. Это делает его особенно удобным для верификации проектов на основе ПЛИС в реальном аппаратном окружении; он будет основным средством отладки РОС.

2.5 Редактор внутрисхемных сигналов

Редактор внутрисхемных сигналов (In-System Sources and Probes Editor) состоит из мегафункции `altsource_probe` и интерфейсной графической оболочки, которая позволяет контролировать все элементы мегафункции `altsource_probe` внутри отлаживаемого проекта в реальном времени. Каждый элемент мегафункции `altsource_probe` дает для отображения исходные выходные порты (источники) и отводы входных портов (пробники). Интерфейсная графическая оболочка показывает все доступные в реальном времени контролируемые элементы мегафункции `altsource_probe` в проекте и обеспечивает создание виртуального пульта (виртуальных кнопок и виртуальной лицевой панели) отладки разрабатываемого проекта, имитацию внешнего счетчика данных, отображение и редактирование констант во время прогона, механизм для подвода исследуемых сигналов и средства сохранения данных.

Редактор внутрисхемных сигналов состоит из трех панелей:

- (1) конфигулятора цепи JTAG, позволяющего разработчику определять устройство программирования, кристалл, файл настроек редактора;
- (2) менеджера элементов, отображающего информацию об элементах, позволяющих контролировать данные динамически;
- (3) собственно редактора внутрисхемных сигналов, сохраняющего журнал данных для исследуемых элементов схемы, позволяющего модифицировать эти данные и записывать их в кристалл.

Использование механизмов скрипирования Tcl (Tool command language) позволяет придать отображению сигналов удобный вид, что делает его одним из основных средств отладки.

Для работы редактора внутрисхемных сигналов в рассматриваемом случае требуется программное средство САПР Quartus II с набором его инструментальных средств, загрузочный кабель USB-blaster и отладочная плата.

С помощью редактора внутрисхемных сигналов можно создать виртуальный пульт для управления отладкой РОУ. Для этого пользователь формирует набор виртуальных кнопок и индикаторов (ВКИ), используя приложение In Source In Probe. В процессе отладки он может подавать различные сигналы на входы схем РОУ и следить за состоянием выбранных выходов схем. С помощью виртуального пульта можно проводить ручное тестирование отдельных схем РОУ. Редактор внутрисхемных сигналов сам по себе не требует внешних устройств. Особенно эффективно это средство отладки в сочетании с SignalTap II и In-System Memory Content Editor, давая наибольшую свободу в контроле над сигналами и создании виртуальных входов. Все это позволит существенно сократить время верификации проекта.

2.6 Виртуальный JTAG

Интерфейс VJTAG (Virtual JTAG — виртуальный JTAG) создан для обеспечения обмена данными между платой и компьютером через кабель USB-blaster. Отладочная плата Cyclone V содержит встроенный контроллер JTAG, позволяющий программировать ПЛИС непосредственно в схеме. Контроллер JTAG представляет собой конечный автомат: для получения доступа к его сигналам и обеспечения информационного обмена с внутренними устройствами ПЛИС служит мегафункция VJTAG. Управление интерфейсом VJTAG осуществляется программой quartus_stp, входящей в состав САПР Quartus II, при помощи сценариев на языке Tcl. Конечные устройства (контроллеры памяти, устройства управления светодиодами и пр.) обмениваются данными через декодер с VJTAG-мегафункцией [7].

При работе с отладочной схемой, использующей VJTAG, можно выделить три типа фрагментов этой схемы: встроенные в ПЛИС (порождающие некоторые ограничения по использованию мегафункций), обязательные компоненты VJTAG (не допускающие вариаций) и требующие ручного управления и создания пользователем [8].

При создании интерфейса передачи данных VJTAG требуется:

- создать мегафункцию VJTAG (можно использовать sld_virtual_jtag);
- создать логику, ответственную за коммутацию сигналов и декодирование команд;
- написать сценарий на языке Tcl/Tk (Tool kit);
- создать (подключить) конечные логические устройства.

Все это, с одной стороны, обеспечивает гибкость и уникальность отладочной схемы, поскольку позволяет создавать схемы, зависящие от конкретных требований. С другой стороны, это усложняет использование мегафункции, заставляет детально изучать принципы построения JTAG-контроллеров и создания программного обеспечения (ПО) для них. Ввиду необходимости создания дополнительного оборудования и повышенной трудоемкости Virtual JTAG не отвечает критериям выбора отладочных средств и его использование при отладке РОС нецелесообразно.

Дополнительные отладочные средства в Quartus II появляются в случае использования процессора Nios II, отладочные средства которого построены на основе модуля JTAG.

Программные инструменты отладки связываются с отладочным модулем JTAG и предлагают следующие средства: загрузка программы в память; пуск/останов исполнения программы; установка программных и аппаратных точек останова и точек просмотра; анализ регистров и памяти процессора; накопление следов (данных) исполнения программ в реальном времени.

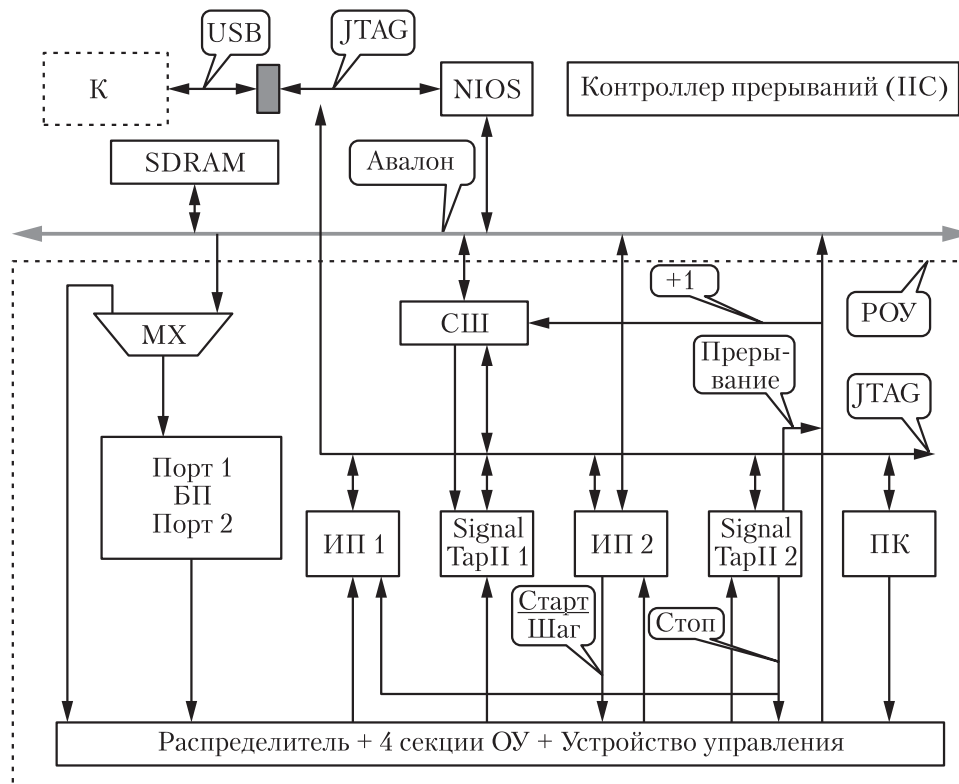
Серьезно расширяет возможности отладки с помощью встроенного логического анализатора SignalTap II плагин Nios II. Он дополняет отладочные средства SignalTap II, разрешая захватывать коды операций процессора Nios II.

В целом процессор Nios II является мощным инструментом отладки, но требующим больших затрат на подготовку и программирование и, к сожалению, не дающий полностью доступа ко всем входам и выходам сигнального процессора, в связи с чем использование процессора Nios II можно рассматривать только как дополнительное.

3 Отладка рекуррентного обработчика сигналов

Фирма Альтера не раскрывает протокол взаимодействия персонального компьютера с отладочной платой. Поэтому пользователь вынужден использовать встроенные средства в Quartus или проектировать свои (уникальные) отладочные средства, например на основе виртуального JTAG. Использование компромиссного варианта в виде сочетания встроенных и дополнительно спроектированных средств отладки позволяет оптимизировать временные и аппаратные затраты на их реализацию. Структура аппаратных средств РОС с учетом его реализации на основе платы Cyclone V GX FPGA Development Board изображена на рисунке.

Разработанный РОС реализован в виде гибридного двухуровневого варианта с ведущим фон-неймановским процессором на верхнем, управляющем уровне (УУ) и рядом потоковых процессоров на нижнем уровне — рекуррентном операционном устройстве (РОУ). Управляющий уровень, реализованный в виде программы на основе процессора Nios II РОУ, представляет собой VHDL (VHSIC (Very High Speed Integrated Circuits) Hardware Description Language) описания для синтеза на ПЛИС Cyclone V.



Структура аппаратных средств РОУ

Обмен данными между этими уровнями осуществляется через буферную память (БП). Для упрощения процесса изложения материала БП отнесена к РОУ. Управляющий уровень состоит из следующих основных модулей: вычислительного ядра Nios II с внутренним контроллером прерываний (ИС — internal interruption controller); системной шины Авалон; памяти процессора Nios II.

В состав РОУ входят следующие функциональные модули: двухпортовая БП; мультиплексор порта 1 записи в БП (МХ); память констант (ПК); распределитель; четыре секции обрабатывающего устройства (ОУ); устройство управления; средства отладки.

Все аппаратные средства РОУ взаимодействуют с компьютером (К) через интерфейс USB 2.0. Для обеспечения доступа ко всем функциональным элементам используется стандартный интерфейс JTAG, встроенный логический анализатор SignalTap II и редактор содержимого внутрисхемной памяти.

В свою очередь, дополнительно разработанные средства отладки РОУ состоят из следующих функциональных модулей: счетчика шагов (СШ); модуля

ИП 1 (источников и пробников) регистров РОУ, модуля ИП 2 (источников и пробников) виртуальных клавиш и индикаторов, двух анализаторов SignalTap II.

Следует отметить, что аппаратные средства отладки РОУ также привязаны к стандартному интерфейсу диагностики и отладки средств JTAG.

Счетчик шагов исполнения РОУ предназначен для контроля процесса отладки и анализа исполняемой капсулы. Пользователь непосредственно и программа УУ могут изменять содержимое СШ в процессе отладки РОУ.

Модуль ИП 1 обеспечивает возможность изменения и просмотра содержимого регистров ОУ в процессе отладки. Модуль ИП 2 содержит набор ВКИ, предназначенных для управления процессом отладки. Он содержит две группы управляющих виртуальных кнопок:

- (1) формирования отладочных команд;
- (2) управления работой РОУ.

Пользователь имеет возможность формировать необходимый набор отладочных команд. Для этого набор ВКИ соединен с шиной Авалон, что обеспечивает возможность программного доступа со стороны Nios. Пользователь набирает команду и инициирует ее исполнение на процессоре Nios.

Встроенные средства отладки САПР Quartus II не обеспечивают возможности чтения и редактирования содержимого оперативной двухпортовой памяти. В данном случае БП является двухпортовой. Поэтому функция чтения и записи данных в память осуществляется программным способом с помощью Nios II и модуля ИП 2.

Приложение SignalTap II имеет расширенные возможности и позволяет разработчику создавать и встраивать в РОУ определенное число логических анализаторов, оперативно изменять условия фиксации данных в их памяти и отображать эти данные на экране компьютера. Особенность этого подхода состоит в исследовании поведения внутренних сигналов без использования дополнительных контактов ввода-вывода и какого-либо внешнего оборудования. Пользователь может подсоединять его входы к различным точкам схемы РОУ, задавать условия фиксации сигналов и далее просматривать временные диаграммы. Для этих целей в состав отладочных средств введен логический анализатор SignalTap II.

Управление процессом отладки РОУ пользователь может осуществлять через виртуальный пульт. Для этого, используя редактор внутрисхемных сигналов, пользователь формирует набор ВКИ. В процессе отладки он может подавать различные сигналы на входы схем РОУ и следить за состоянием выбранных выходов схем. С помощью виртуального пульта можно проводить ручное тестирование отдельных схем РОУ.

Архитектура РОУ существенно отличается от традиционной, поэтому процесс отладки носит событийный характер. Для обеспечения возможности настройки и управления процессом отладки РОУ используется формирователь отладочных событий, построенный на основе SignalTap II. Пользователь в приложении

SignalTap II формирует условия отладочных событий, при возникновении которых работа РОУ останавливается и формируется сигнал Прерывание, который инициирует исполнение программы обработки отладочного события в Nios.

4 Заключение

1. Предоставляемые САПР Quartus II инструменты системной отладки удобны в использовании и позволяют существенно упростить процесс верификации проектов на основе ПЛИС в реальном аппаратном окружении.
2. Реализованные в РОС отладочные средства достаточны для поддержки эффективного процесса совместной отладки аппаратных и программных средств.

Литература

1. Волчек В. Н., Степченков Ю. А., Петрухин В. С., Прокофьев А. А., Зеленов Р. А. Цифровой сигнальный процессор с нетрадиционной рекуррентной потоковой архитектурой // Проблемы разработки перспективных микро- и наноэлектронных систем-2010: Сб. трудов / Под общ. ред. акад. А. Л. Стемпковского. — М.: ИППМ РАН, 2010. С. 412–417.
2. Шнейдер А. Ю., Петрухин В. С., Степченков Ю. А. Принципы построения средств отладки рекуррентного вычислителя // Проблемы разработки перспективных микро- и наноэлектронных систем-2012: Сб. трудов / Под общ. ред. акад. А. Л. Стемпковского. — М.: ИППМ РАН, 2012. С. 133–136.
3. Упрощение отладки ПЛИС Xilinx и Altera // Каталог оборудования 2012–2013. Решения в области контрольно-измерительной аппаратуры. — Tektronix, 2012. С. 386–395. http://www.tehcom.com/Companies/Tektronix/Tektronix_Catalog_2012_Rus.pdf.
4. Cyclone V GX FPGA Development Board: Reference Manual. — Altera Corporation, 2013. http://www.altera.com/literature/manual/rm_cvfx_fpga_dev_board.pdf.
5. Антонов А., Филиппов А., Золотуха Р. Средства системной отладки САПР Quartus II // Компоненты и технологии, 2008. № 12(89). С. 43–50.
6. Quartus II Handbook Version 12.0. Vol. 3: Verification. — Altera Corporation, 2013. http://www.altera.com/literature/hb/qts/quartusii_handbook.pdf.
7. Гребенников А. Интерфейс VJTAG для отладочной платы DK-START-3C25N // Современная электроника, 2010. № 9. С. 60–63.
8. Михайлов М., Грушвицкий Р. Проектирование в условиях временных ограничений: отладка проектов (часть 3) // Компоненты и технологии, 2007. № 9(74). С. 133–138.

Поступила в редакцию 31.03.14

SYSTEM VERIFICATION TOOLS FOR RECURRENT SIGNAL PROCESSOR

V. S. Petrukhin, D. Y. Stepchenkov, N. V. Morozov, and Y. A. Stepchenkov

Institute of Informatics Problems, Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation

Abstract: A procedure of selecting and developing a software and hardware suite is studied which is intended for designing and debugging a nontraditional digital signal processor based on the recurrently-dynamic dataflow architecture — the recurrent signal processor (RSP). The experimental character of the developed RSP's architecture as well as a necessity for a finished master processor have predetermined Cyclone V family FPGA (Field Programmable Gate Array) (Altera) as the base of RSP implementation and Quartus II design software for its development. The powerful verification tools contained by Quartus II allow both reducing the time of obtaining the finished design and reducing the hardware expenses essentially. On the basis of the comparative analysis and selected criteria, the tools composition for debugging RSP is determined, and an optimal structure of hardware for debugging RSP is offered that allow for essential simplification of the verification process and for debugging RSP in a real hardware environment.

Keywords: debugging means; dataflow architecture; verification

DOI: 10.14357/08696527140204

Acknowledgments

The research was performed with the partial financial support for Programs of Basic Research of the Department of Nanotechnologies and Information Technologies of the Russian Academy of Sciences for 2013 (project 1.5) and the Presidium of the Russian Academy of Sciences (project 16).

References

1. Volchek, V. N., Yu. A. Stepchenkov, V. S. Petrukhin, A. A. Prokof'ev, and R. A. Zelenov. 2010. Cifrovoy signal'nyy protsessor s netraditsionnoy rekurrentnoy potokovoy arkhitekturoy [Digital signal processor with nontraditional recurrent dataflow architecture]. *Problemy Razrabotki Perspektivnykh Mikro- i Naoelektronnykh Sistem* [Problems of the Perspective Micro- and Nanoelectronic Systems Development-2010]. Ed. akad. A. L. Stempkovskij. Moscow: IPPM RAN. 412–417.
2. Shnejder, A. Ju., V. S. Petrukhin, and Yu. A. Stepchenkov. 2012. Printsipy postroeniya sredstv otladki rekurrentnogo vychislitelya [Development principles of debugging tools

- for the recurrent computing device]. *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem* [Problems of the perspective micro- and nanoelectronic systems development-2012]. Ed. acad. A. L. Stempkovskij. Moscow: IPPM RAN. 133–136.
3. Tektronix. 2012. Uproshchenie otladki PLIS Xilinx i Altera [Simplification of debugging FPGA Xilinx and Altera]. Katalog oborudovaniya 2012–2013. Resheniya v oblasti kontrol'no-izmeritel'noy apparatury [Equipment 2012–2013 directory. Decisions in the field of control instrumentation]. 386–395. http://www.tehcom.com/Companies/Tektronix/Tektronix_Catalog_2012_Rus.pdf (accessed March 31, 2014).
 4. Altera Corporation. 2013. Cyclone V GX FPGA Development Board Reference Manual. URL: http://www.altera.com/literature/manual/rm_cvngx_fpga_dev_board.pdf (accessed March 31, 2014).
 5. Antonov, A., A. Filippov, and R. Zolotuh. 2008. Sredstva sistemnoy otladki SAPR Quartus II [System debugging tools of Quartus II CAD]. *Komponenty i Tekhnologii* [Components and Technologies] 12:43–50.
 6. Altera Corporation. 2013. Quartus II Handbook Version 12.0. Vol. 3: Verification. URL: http://www.altera.com/literature/hb/qts/quartusii_handbook.pdf (accessed March 31, 2014).
 7. Grebennikov, A. 2010. Interfeys VJTAG dlya otladochnoy platy DK-START-3C25N. [VJTAG interface for development board DK-START-3C25N]. *Sovremennaya Elektronika* [The Modern Electronics] 9:60–63.
 8. Mihajlov, M., and R. Grushvickij. 2007. Proektirovanie v usloviyakh vremennykh ogranicheniy: Otladka proektov (Chast' 3). [Design in the time restriction conditions: Debugging of projects (part 3)]. *Komponenty i Tekhnologii* [Components and Technologies] 9:133–138.

Received March 31, 2014

Contributors

Petrukhin Vladimir S. (b. 1949) — senior scientist, Institute of Informatics Problems, Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; cokrat2@rambler.ru

Stepchenkov Dmitri Yu. (b. 1973) — senior scientist, Institute of Informatics Problems, Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; stepchenkov@mail.ru

Morozov Nikolai V. (b. 1956) — senior scientist, Institute of Informatics Problems, Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; nmorozov@ipiran.ru

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