

УДК 681.324.192

**ОСОБЕННОСТИ ГИБРИДНОГО ВАРИАНТА
РЕАЛИЗАЦИИ НА ПЛИС РЕКУРРЕНТНОГО
ОБРАБОТЧИКА СИГНАЛОВ***Ю. А. Степченков, В. С. Петрухин*

В статье анализируется современное состояние архитектур, предназначенных для параллельных вычислений. Предложена структура компромиссного (гибридного) варианта реализации архитектуры рекуррентного процессора для сигнальной обработки, разрабатываемой в ИПИ РАН. Статья рассматривается как вводная. Конкретные результаты реализации отдельных положений предлагаемой структуры представлены в двух других статьях настоящего сборника.

1. Введение

Результат анализа развития архитектур, предназначенных для параллельных вычислений, в расширенном варианте был представлен в [1], а в уточненном и сжатом — в [2]. Там же была показана целесообразность гибридного варианта реализации рекуррентного обработчика сигналов (РОС) на базе семейства микросхем Excalibur фирмы Altera. Научные исследования в этом направлении в отделе «Архитектур перспективных компьютерных систем» до 2007 г. проводились в ограниченном объеме, а основные усилия были направлены на развитие самосинхронной схемотехники как основы построения аппаратных средств РОС. Полученные результаты изложены в работах [3; 4].

За прошедшее время операционный (нижний) уровень РОС продолжал совершенствоваться, и его спецификация приобрела завершенность. Определилась задача экспериментальной реали-

зации двухуровневого варианта РОС с ведущим фон-неймановским процессором на верхнем (управляющем) уровне и рядом потоковых процессоров на нижнем уровне.

Прошедшие четыре года после публикации [2] — достаточно большой срок в такой динамично развивающейся области, как разработка микропроцессоров и выбор аппаратной среды как базы для реализации РОС. В связи с этим возникла необходимость уточнить состояние архитектур, предназначенных для параллельных вычислений (в виде краткого обзора появившихся публикаций на эту тему) и достаточно обоснованно выбрать необходимую аппаратную среду реализации РОС. Конкретные результаты реализации отдельных положений предлагаемой структуры представлены в [5; 6].

В области коммерчески выпускаемых микропроцессоров (МП) за последние 35 лет произошли колоссальные изменения: разрядность микропроцессора увеличилась в 16 раз — с 4 разрядов (Intel 4004, 1971 г.) до 64; число транзисторов с 2300 приблизилась к 1 млрд; тактовая частота с 108 кГц перешла за 4 ГГц; появилась быстродействующая кэш-память. Однако, достигнув указанных рубежей, производители микропроцессоров отступили — *возможности дальнейшего повышения производительности микропроцессоров путем уменьшения технологических норм и наращивания числа транзисторов оказались исчерпанными* [7]. Дальнейшее наращивание числа запускаемых на выполнение команд и объемов кэш-памяти не дает прироста производительности вычислительных систем пропорционально затрачиваемым аппаратным ресурсам и энергии [8], и простое наращивание тактовой частоты не оправданно экономически [7].

Леониду Черняку [8] проблема видится в том, что до сих пор осталась неизменной фон-неймановская архитектура со всеми присущими ей достоинствами и пороками. Позволим себе пространную цитату из указанной работы: «Ее сила — во внутренней гармоничности, но пребывать в согласии способны лишь компоненты относительно простой системы. Обычно монолитные схемы имеют предел количественного роста, за которым начинаются проблемы внутренней структурной сложности, и собранные из транзисторов процессоры — не исключение».

Для преодоления ограничений фон-неймановской архитектуры используется ряд подходов, в том числе применяются специализированные ускорители, однородные и неоднородные много-

ядерные структуры [9] и программируемые массивы [8]. Однако эффективность использования этих подходов носит ограниченный характер в силу ориентации опять же на основополагающие принципы фон-неймановской архитектуры.

Дальнейшее усложнение архитектуры не приносит ожидаемого выигрыша в производительности. Совершенствование архитектуры единственного ядра не дает ожидаемой отдачи: сроки разработки «продвинутых» моделей резко возросли; заявленные производителями пиковые характеристики процессоров практически недостижимы без низкоуровневого программирования (большинство вычислительных задач выполняется на массовых микропроцессорах с эффективностью 10–20%) [10].

Все эти и другие факторы стимулируют поиски новых путей развития индустрии МП: «...они не укладываются в рамки традиционных представлений — для дальнейшего движения требуются новые архитектурные подходы» [7].

Перед разработчиками с новой силой встала задача — максимально сблизить пиковую и реальную производительности МП без традиционных компромиссов, а путем обращения к новым архитектурам [7].

Кроме проблем фундаментального плана следует отметить и ряд технико-экономических факторов прикладного значения, вносящих свой вклад в смену критериальных ориентиров.

1. Современные процессоры создаются в предположении, что аппаратура статична, а программное обеспечение (ПО) создается специально под нее. Однако практика показывает, что граница между аппаратурой и ПО весьма условна, пришло время реконфигурируемых (программируемых) процессорных ядер. Сегодня в ряде случаев проще менять аппаратное обеспечение, чем перерабатывать сложные программные системы.

2. Еще недавно процессоры стоили дорого, потребляемая ими энергия не являлась критическим фактором, и проблема рассеиваемой в них мощности решались экономически эффективными способами. Сейчас, напротив, транзисторы дешевы, а решение проблемы энергетической эффективности МП и их теплоотвода становится все более актуальным.

3. Прежде исходили из того, что арифметические операции выполняются медленно, а доступ к оперативной памяти осуществляется быстро. Теперь же доступ к памяти становится одним из барьеров для роста производительности.

4. Если раньше увеличение параллелизма обеспечивалось путем совершенствования компиляторов, то сейчас эту задачу передают аппаратному обеспечению.

Архитектура рекуррентного компьютера (РК), разрабатываемая в ИПИ РАН [11] имеет два существенных резерва повышения производительности по сравнению с традиционными подходами. Если в логическом плане число шагов для выполнения отдельной инструкции в фон-неймановских и потоковых архитектурах равно пяти (например, выборка инструкции, ее дешифрация, чтение исходных данных, выполнение инструкции и запись результата), то в РК их три: сравнение тегов/дешифрация инструкции, выполнение инструкции и запись результата.

Второй резерв связан с ориентацией РК на самосинхронную (СС) схемотехнику, цикл работы которой определяется реальным быстродействием аппаратных средств (реальными задержками элементов, обработкой реальных данных в рамках реального окружения — напряжения питания, температуры и емкостных нагрузок). Время реального цикла функционально идентичной СС-аппаратуры меньше (в 1,5–2 раза), чем цикл синхронной аппаратуры, вынужденный ориентироваться на худшие условия и, следовательно, худшее быстродействие [12; 13].

В ИПИ РАН разработана *методология проектирования СС-элементной базы*. СС-исполнение аппаратуры обеспечивает реализацию в ней целого ряда замечательных свойств. В рамках рассматриваемой проблемы имеется возможность реализовать *концептуальное единство* рекуррентной парадигмы, изначально ориентирующейся на асинхронный механизм реализации на архитектурном уровне, и СС-исполнения аппаратуры, которое позволит эксплуатировать алгоритмический параллелизм наиболее естественным и эффективным способом.

Результаты работ по различным вариантам исполнения РК свидетельствуют, что он не требует сложных аппаратных механизмов для поддержки присущей ему капсульной модели программирования [14].

В рамках работ по становлению архитектуры РК поиск решения проблемы программируемости ведется одновременно по двум направлениям.

Первое направление предполагает разработку уникальной вычислительной среды, в максимальной степени учитывающей специфику РК. Отсюда и масштабность проблем, и долговременный

характер их решения. Работы в рамках этого направления отвечают критерию *научной целесообразности*.

Работы в рамках второго направления предполагают поиск возможного компромиссного решения, обеспечивающего совместимость с существующими вычислительными системами, которые удовлетворяют в первую очередь требованию поддержки потокового характера вычислительного процесса (ВП), реализуемого в РК на базе использования функционального стиля программирования. Работы в этом направлении отвечают критерию *практической целесообразности*.

2. Структура гибридного варианта реализации рекуррентного процессора

Теоретическую основу архитектуры РК, которая разрабатывается в ИПИ РАН, составляет новая рекуррентная парадигма. Организация ВП в РК базируется на следующем теоретическом базисе:

- парадигме управления ВП от потока данных, в котором два потока — поток данных и поток команд — интегрированы в один самодостаточный поток данных, т. е. традиционная двухпоточность сводится к однопоточности;
- рекуррентной форме представления обрабатываемой информации;
- парадигме вычислений, в основе которой лежит графодинамическое представление алгоритмов, их рекуррентной свертки (до момента инициации исполнения) и саморазвертки в процессе исполнения.

Интеграция и необходимая модернизация этих отдельных положений в единое целое характеризуется теоретической новизной и является базой для возможного построения архитектур РК, характеризующихся повышенной эффективностью организации ВП.

Архитектура рекуррентного обработчика сигналов (РОС) изначально разрабатывается как специализированная, предназначенная для реализации параллельных ВП обработки сигналов в реальном времени. Она базируется на рекуррентной парадигме, которую можно рассматривать как развитие парадигмы потока данных (Data flow) [15], но построенной на другом принципе,

а именно на базе самодостаточных рекуррентно сжатых данных [16].

В работе [17] высказывается мнение (к которому присоединяются и авторы настоящей статьи) о том, что кроме общеизвестных недостатков фон-неймановской организации вычислений (доминирующей в проектах общего назначения), «одной из ее главных сил ... является универсальный характер ее программной организации. Она универсальна не только в смысле машины Тьюринга, но способна эффективно поддерживать спектр стилей программирования и эффективно моделировать многообразие структур алгоритмов». Поэтому стремление объединить положительные особенности фон-неймановских и потоковых архитектур вызвало интерес к комбинированным (гибридным) потоково-фон-неймановским архитектурам. Различные подходы в реализации данной архитектуры изложены в работе [18].

Для развития потоково-фон-неймановской архитектуры предлагается использовать двухуровневую архитектуру с управляющим фон-неймановским процессором на верхнем уровне и многими потоковыми процессорами на нижнем уровне, а связь между уровнями осуществлять посредством устройства сопряжения. На управляющий процессор могут быть возложены следующие функции:

- связи многопроцессорной потоковой системы с внешним миром;
- интерфейса между стандартным и специализированным программным обеспечением РК;
- вычислителя для обработки возможных последовательных частей алгоритма, реализуемого комбинированным РОС;
- устройства управления для обработки исключительных ситуаций;
- устройства-компоновщика самодостаточных капсул для их исполнения на уровне РОУ архитектуры РОС.

Однако необходимо приложить серьезные усилия, чтобы ведущий управляющий процессор не стал узким местом всей гибридной потоково-фон-неймановской архитектуры.

Для проведения экспериментальных исследований и отработки структурных решений РОС в качестве элементной базы предполагается использовать программируемые логические интегральные схемы (ПЛИС). Гибкость и оперативность изменяемой структуры делает их бесспорными фаворитами при использовании в исследовательских проектах.

Подробный анализ выбора элементной базы для реализации РОС был выполнен авторами [18]. При выборе конкретного семейства ПЛИС авторы руководствовались возможностью адекватного и эффективного решения предлагаемой структуры РОС. Следует отметить, что по своей структуре современные ПЛИС отвечают требованиям построения РОС с гибридной архитектурой. В отдельных случаях они содержат управляющий процессор, который посредством интерфейса соединен с матрицей PLD [19]. На базе матрицы PLD можно реализовать специализированное устройство, в нашем случае — несколько процессоров.

В результате анализа предпочтение было отдано семейству микросхем ПЛИС фирмы Altera — Excalibur, которое может содержать следующие встроенные процессорные ядра:

- ARM — 32-разрядный ARM922 Thumb® RISC-процессор (встроенный, несинтезируемый);
- MIPS — 32-разрядный MIPS32 4K RISC-процессор (синтезируемый);
- NIOS — 32-разрядный NIOS II RISC-процессор (синтезируемый).

В первую очередь предполагалось использование процессора, выполненного на основе индустриального стандарта ARM, который получил к тому времени признание и поддержку большого количества фирм-разработчиков специализированных схем для различных применений. Программные продукты для ARM образовали солидную инфраструктуру ПО и средств разработки. Кроме того, в составе семейства процессоров ARM имеется несколько удачных экспериментальных асинхронных моделей, в том числе первый коммерчески доступный асинхронный (self-timed, clockless) процессор ARM996HS [20]. Интересно отметить, что в состав процессора ARM996HS входит также и асинхронный сопроцессор, выполняющий операции деления и извлечения корня. Квазисамосинхронная реализация подобного сопроцессора выполнена в ИПИ РАН [4]. Как отмечалось выше, РК также ориентирован на использование самосинхронной схемотехники. Поэтому можно было бы предположить, что в составе семейства Excalibur появится и встроенное асинхронное ядро ARM.

Однако этого не произошло. Более того, обнаружилось три фактора, ставящие под сомнение целесообразность использования семейства Excalibur в качестве аппаратной среды реализации РОС.

1. Отсутствие аппаратно реализованных умножителей приводит к существенному снижению их быстродействия при программном синтезе. В результате быстродействие устройств умножения с накоплением (МАС), определяющих быстродействие всего РОУ, оказалось недопустимо низким.

2. 23 марта 2007 г. фирма Altera объявила о прекращении производства семейства Excalibur с рекомендуемой заменой последних семействами ПЛИС Cyclone, и в частности Cyclone III. Количество встроенных умножителей (18 × 18 битов) в семействе Cyclone III в зависимости от конкретной серии изменяется от 24 до 256, что полностью покрывает потребности четырехсекционного варианта РОУ.

3. Уточненный анализ аппаратных затрат РОУ показал, что максимальное число логических элементов (38 400 в микросхеме EPXA10 семейства Excalibur) оказывается недостаточным для его реализации вместе с управляющим процессором в одной ПЛИС.

В то же время максимальная логическая емкость семейства Cyclone III [21] увеличилась до 120 тыс. логических элементов; до 287 МГц выросло быстродействие встроенных умножителей, реализована поддержка синтезируемого процессорного ядра Nios II (за счет использования незначительной части программируемых ресурсов ПЛИС). Например, при синтезе 32-разрядного процессора Nios II в кристалле Cyclone требуется меньше 6000 логических элементов [22].

Серьезным фактором при выборе ПЛИС для реализации РОС явились результаты моделирования наиболее сложных функциональных устройств: умножитель с накоплением (Multiplier with Accumulation, МАС) и память адресной проверки (ПАП). Результаты моделирования показали, что для достижения требуемого быстродействия и балансировки отдельных устройств необходимо наличие встроенных (несинтезируемых) 16-разрядных умножителей [5].

В структуру РОС на основе ПЛИС (рис. 1) входят: управляющий процессор — управляющий уровень гибридного РОС на базе фон-неймановской архитектуры, распределитель, четыре однотипных рекуррентных процессорных устройства (ПУ) и интерфейс межпроцессорного обмена — исполнительный уровень гибридного РОС. Двухпортовая память является информационным интерфейсом двух уровней РОС.

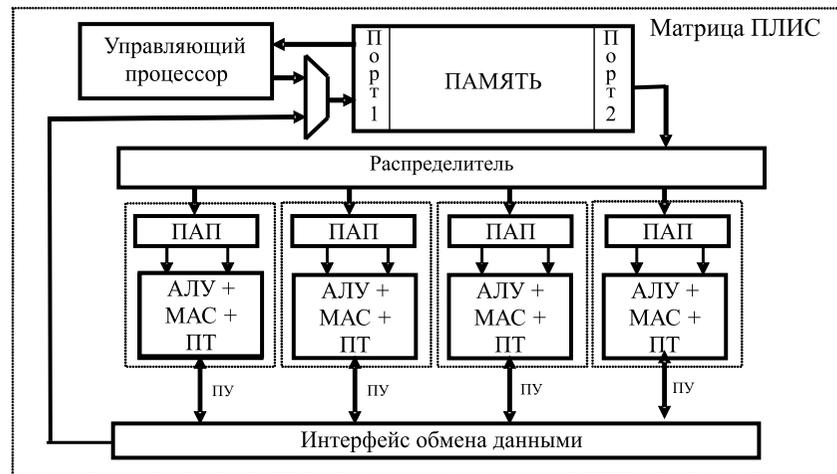


Рис. 1. Структура РОС на основе ПЛИС

В свою очередь каждое ПУ состоит из памяти адресной проверки (ПАП), вычислителя (на базе 16-разрядного АЛУ и умножителя с накоплением МАС) и преобразователя тегов (ПТ) [23].

В качестве устройства сопряжения управляющего процессора и исполнительных устройств (поток данных между ними носит интенсивный характер) предлагается использовать специальный тип двухпортового запоминающего устройства (ЗУ), реализованного в ПЛИС.

Одновременное чтение капсулы из одного порта и запись результатов операции в другой порт предположительно позволит сбалансировать поток данных между управляющим процессором и четырьмя операционными процессорными устройствами.

На управляющий уровень РОС возлагается выполнение процедур предварительной подготовки капсул, реализация последовательных частей исполняемой программы и запись результатов в один порт ЗУ. Операционный уровень РОС обеспечивает считывание капсул, готовых для исполнения, параллельные вычисления в процессорных устройствах и запись результатов вычислений в другой порт. Введение в состав РОС двухпортового ЗУ позволит также исключить потери времени на перемещение данных между двумя уровнями архитектуры комбинированного РОС. Первоначально специфицированный вариант РОС [11]

предполагал обязательное использование входного и выходного буферных FIFO, которые в данном случае исключаются.

3. Выводы

1. Специалисты в области архитектур ВС все чаще приходят к мнению, что господствующая фон-неймановская архитектура превысила тот предел количественного роста ее структуры, при котором гармония компонентов относительно простой системы сменяется неэффективностью более сложной структуры. Эффективность традиционных средств решения этой проблемы постоянно снижается, что приводит к пересмотру путей развития индустрии МП, которые не укладываются в рамки традиционных представлений, для дальнейшего движения требуют новых архитектурных подходов. Поиску новых подходов и посвящены исследования по разработке архитектуры на базе новой вычислительной рекуррентно-поточковой парадигмы.

2. Исходя из архитектурных новаций, доказавших свою эффективность в ходе испытаний опытных экземпляров МП или по результатам компьютерного моделирования, можно предположить, что применительно к разработке компьютеров специализированного назначения предпочтителен мультаядерный вариант исполнения операционного уровня на базе рекуррентного подхода и (в перспективе) самосинхронной схмотехники и мультитредовый вариант исполнения управляющего уровня в виде одного синхронного процессора средней производительности с фон-неймановской архитектурой.

3. В силу перечисленных причин и других аргументов, изложенных в [18; 19], для применения в качестве элементной базы РОС предлагается использовать изделия фирмы Altera, конкретно — семейство Cyclone III.

Список литературы

1. Степченков Ю. А., Петрухин В. С. Исследование новой вычислительной парадигмы и разработка на её основе логического проекта динамического многопоточного процессора обработки сигналов: Отчет о НИР, шифр «Сигнал», № г. р. 01.2.00 104927. М.: ИПИ РАН, 2001. С. 2.1–2.97.
2. Степченков Ю. А., Петрухин В. С. Перспективы развития потоковых сигнальных процессоров и возможная реализация рекур-

- рентного обработчика сигналов // Методы и средства разработки информационно-вычислительных систем и сетей (спец. выпуск). М.: Наука, 2004. С. 89–133.
3. Соколов И.А., Степченков Ю.А., Петрухин В.С. и др. Самосинхронная схмотехника — перспективный путь реализации аппаратуры // Системы высокой доступности. 2007. Т. 3, № 1–2. С. 61–72.
 4. Степченков Ю.А., Дьяченко Ю.Г., Рождественский Ю.В. и др. Квазисамосинхронная реализация устройства деления и извлечения квадратного корня // Системы и средства информатики. Вып. 18. М.: Наука, 2008. С. 234–260.
 5. Петрухин В.С., Волчек В.Н., Прокофьев А.А., Зеленов Р.А. Особенности реализации рекуррентного обработчика сигналов с гибридной архитектурой // Наст. сб. С. 130–148.
 6. Степченков Ю.А., Петрухин В.С., Хилько Д.В. Выбор языковых средств представления параллельных алгоритмов для рекуррентного обработчика сигналов // Наст. сб. С. 149–158.
 7. Волков Д. Реальность и фантазии // Открытые системы. 2006. № 5.
 8. Черняк Л. Микропроцессоры: все только начинается // Открытые системы. 2006. № 5.
 9. Николаев А. Поддержка многоядерных процессоров во встраиваемых системах // Открытые системы. 2006. № 7.
 10. Корнеев В. Современные подходы к повышению производительности // Открытые системы. 2006. № 5.
 11. Степченков Ю.А., Петрухин В.С., Филин А.В. Рекуррентное операционное устройство для процессоров обработки сигналов // Системы и средства информатики. Вып. 11. М.: Наука, 2001. С. 283–315.
 12. Степченков Ю.А., Дьяченко Ю.Г., Петрухин В.С., Филин А.В. Цена реализации уникальных свойств самосинхронных схем // Системы и средства информатики. Вып. 9. М.: Наука, 1999. С. 261–292.
 13. Beerel P., Cortadella J., Kondratyev A. Bridging the gap between asynchronous design and designers (Tutorial) // VLSI Design Conference. Mumbai, 2004.
 14. Филин А.В. Язык описания задач для визуального конструктора рекуррентных капсул: постановка задачи // Методы и средства разработки информационно-вычислительных систем и сетей (спец. выпуск). М.: Наука, 2004. С. 38–55.
 15. Arvind R., Nikhil R.S. Executing a program on the MIT tagged-token dataflow architecture // IEEE Trans. Computer. 1990. Vol. 39, № 3. P. 300–318.

16. Филин А.В. Динамический подход к выбору архитектуры вычислительных устройств обработки сигналов // Системы и средства информатики. Вып. 11. М.: Наука, 2001. С. 247–261.
17. Agarwal A., Lim B.-H., Kranz D., Kubiawicz J. APRIL: A processor architecture for multiprocessing // Proc. 17th Annual Intl. Symp. on Computer Architecture, Seattle, Washington, USA., May 28–31. 1990. P. 104–114.
18. Степченков Ю.А., Петрухин В.С. Исследование программируемости архитектурно-алгоритмических и схмотехнических проблем проектирования рекуррентных компьютеров: заключительный отчет. Раздел 2.2. «Элементная база рекуррентного сигнального процессора». Шифр «ПАРСЕК». № г.р. 01.20.0412412. М.: ИПИ РАН, 2006. С. 2.27–2.40.
19. About Excalibur Embedded Processor solutions. <http://www.altera.com/products/devices/excalibur/exc-index.html/>.
20. Clarke P. ARM clockless core cuts power to about a third. <http://www.eetimes.eu/semi/179103395/>.
21. Cyclone III Device Handbook. Vol. 1. Altera Corporation, 2006. P. 136.
22. Cyclone Devices & Nios II Embedded Processors. http://www.altera.com/products/devices/cyclone/features/cyc-cyclone_nios.html/.
23. Степченков Ю.А., Петрухин В.С. Исследование новой вычислительной парадигмы и разработка на её основе логического проекта динамического многопоточного процессора обработки сигналов: Отчет о НИР, кн. 2 (заключительный), шифр «Сигнал», № г.р. 01.2.00 104927. М.: ИПИ РАН, 2003. С. 12–65.