

И. А. Каляев, И. И. Левин, Е. А. Семерников, А. И. Дордопуло

## РЕКОНФИГУРИРУЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ НА ОСНОВЕ ПЛИС СЕМЕЙСТВА VIRTEX-6

Реконфигурируемые вычислительные системы на основе ПЛИС обладают высокой реальной производительностью и близким к линейному росту производительности при увеличении аппаратного ресурса системы. В статье рассматриваются конструктивные особенности, технические характеристики и достигаемые значения реальной производительности для вычислительных модулей реконфигурируемых вычислительных систем (РВС) на основе ПЛИС семейства Virtex-6. Описан программный комплекс средств разработки параллельных прикладных программ для РВС. *Реконфигурируемые вычислительные системы; ПЛИС; аппаратный ресурс; вычислительные модули; Virtex-6*

### ВВЕДЕНИЕ

В последние годы наметилась устойчивая тенденция использования новых архитектурных решений для достижения пиковых значений производительности сверхвысокопроизводительных систем. Одним из наиболее распространенных решений является использование программируемых логических интегральных схем (ПЛИС) для выполнения вычислений. На второй строчке списка TOP-500 за ноябрь 2010 года значится суперЭВМ Jaguar - Cray XT5-HE, произведенная фирмой Cray Inc., с пиковой производительностью 2331.00 Тфлопс (в предыдущем списке TOP-500 за июнь 2010 года эта суперЭВМ находилась на первом месте), в составе которой в качестве сопроцессоров используются ПЛИС большой интеграции. В большинстве содержащих ПЛИС вычислительных систем, так же как и в Jaguar-Cray XT5-HE, кристаллы ПЛИС используются как дополнение к микропроцессорам, выполняющее трудно- или неэффективно реализуемые на универсальных микропроцессорах фрагменты вычислений.

Однако, как это показано в [1, 2, 3], ПЛИС обладают значительно большим вычислительным потенциалом, который в полной мере может быть реализован в реконфигурируемых вычислительных системах (РВС), содержащих множество кристаллов ПЛИС, используемых как основной вычислительный элемент. Успешно развивающаяся более 20 лет в НИИ многопроцессорных вычислительных систем Южного федерального университета (г. Таганрог) концепция построения многопроцессорных вычислительных систем с программируемой архитектурой позволила создать целый ряд РВС различных архитектур и конфигураций, предназна-

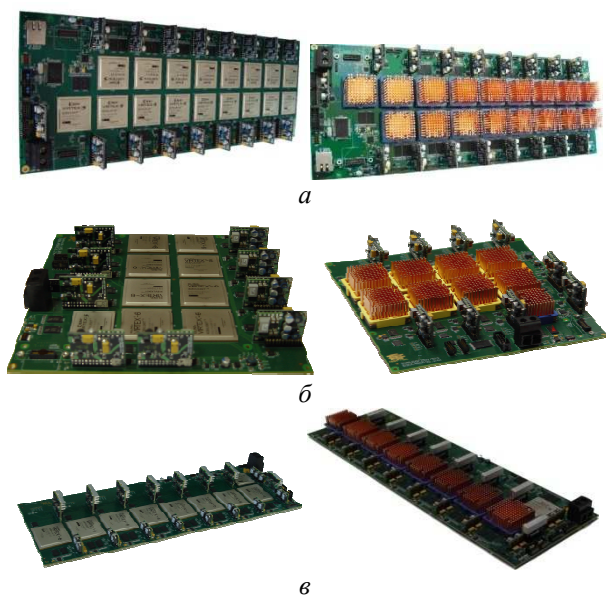
ченных для решения вычислительно трудоемких задач различных предметных областей, успешно эксплуатируемых организациями и ведомствами Российской Федерации. В качестве элементной базы для построения таких РВС используются ПЛИС Xilinx семейства Virtex большой интеграции, соединенные в единый вычислительный ресурс быстрыми каналами передачи данных – LVDS и Rocket GTX.

В НИИ многопроцессорных вычислительных систем Южного федерального университета серийно выпускались реконфигурируемые вычислительные системы на основе ПЛИС семейства Virtex 5, описанные в [3, 4], разработанные по государственному контракту № 02.524.12.4002 «Создание семейства высокопроизводительных многопроцессорных вычислительных систем с динамически перестраиваемой архитектурой на основе реконфигурируемой элементной базы и их математического обеспечения для решения вычислительно трудоемких задач», выполняемого по заданию Федерального агентства по науке и инновациям в рамках Федеральной целевой программы «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007-2012 годы».

Переход к принципам открытой масштабируемой архитектуры [1] в области разработки РВС положил начало новому семейству вычислительных систем под названием «Орион» и привел к созданию в 2010 году платы модифицированного вычислительного модуля с новой компоновкой и конструктивными решениями на основе ПЛИС семейства Virtex 5, принципы построения и технические характеристики которого описаны в [5].

### 1. ВЫЧИСЛИТЕЛЬНЫЕ МОДУЛИ РВС НА ОСНОВЕ ОТКРЫТОЙ МАСШТАБИРУЕМОЙ АРХИТЕКТУРЫ

В настоящее время коллектив разработчиков НИИ многопроцессорных вычислительных систем Южного федерального университета приступил к выпуску РВС нового поколения на основе разработанных вычислительных модулей с использованием ПЛИС семейства Virtex-6. Разработаны и созданы платы нового поколения на основе ПЛИС семейства Virtex-6, построенные на основе открытой масштабируемой архитектуры [1] для вычислительных модулей двух перспективных конструктивных исполнений – «Саиф» и «Ригель», названных именами звезд из астрономического созвездия «Орион». Фотографии платы модифицированного вычислительного модуля на основе ПЛИС семейства Virtex 5 и плат нового поколения представлены на рис. 1, а–в.



**Рис. 1.** Платы вычислительных модулей: а – плата вычислительного модуля на основе ПЛИС семейства Virtex 5; б – плата вычислительного модуля «Саиф»; в – плата вычислительного модуля «Ригель»

В табл. 1 приведены технические характеристики рассматриваемых плат вычислительных модулей. Вычислительные модули на основе этих плат «Орион-5», «Саиф» и «Ригель» имеют высоту 1U, 6U и 1U соответственно и предназначены для установки в стандартную 19" вычислительную стойку, которая является базовым компонентом для создания сверхвысокопроизводительных комплексов на основе ПЛИС. Фотографии вычислительных модулей

«Орион-5», «Саиф» и «Ригель» представлены на рис. 2.

Таблица 1

Технические характеристики плат вычислительных модулей	Плата вычислительного модуля		
	«Орион-5»	«Саиф»	«Ригель»
Число ПЛИС	16	8	8
Тип и наименование ПЛИС	Virtex 5	Virtex 6	Virtex 6
Количество эквивалентных вентилях в 1 ПЛИС, млн шт.	11	24	24
Интерфейс и скорость межмодульного обмена, Гбит/сек	LVDS, 1,2	Gigabit Ethernet, 1	Gigabit Ethernet, 1
Потребляемая мощность, ВА	250	300	300



**Рис. 2.** Вычислительные модули нового поколения: а – вычислительный модуль «Орион-5»; б – вычислительный модуль «Саиф»; в – вычислительный модуль «Ригель»

Применение ПЛИС семейства Virtex 6 в качестве элементной базы для построения вычислительных модулей «Саиф» и «Ригель» позволяет при сохранении стоимости поставки вычислительного модуля увеличить производительность в 1,5–2 раза по сравнению с аналогичным решением на основе ПЛИС семейства Virtex 5 для вычислительного модуля

«Орион-5». Этот факт позволяет рассматривать созданные вычислительные модули нового поколения как наиболее перспективные варианты для построения РВС различных архитектур и конфигураций и обеспечивает им существенное конкурентное преимущество по большинству технико-экономических параметров: удельной производительности, энергоэффективности и др.

В табл. 2 представлены пиковые производительности рассматриваемых вычислительных модулей и вычислительных стоек на их основе. Производительность соответствует обработке данных с одинарной ( $Pi_{32}$ ) и двойной ( $Pi_{64}$ ) точностью в соответствии со стандартом IEEE-754 для вычислительных модулей и стоек описанных изделий. Технические характеристики вычислительных модулей представлены в табл. 2.

Таблица 2

Наименование вычислительного модуля	Технические характеристики вычислительных модулей		
	Производительность вычислительного модуля $Pi_{32}/Pi_{64}$ (Гфлопс)	Число вычислительных модулей в 19" стойке	Производительность стойки $Pi_{32}/Pi_{64}$ (Тфлопс)
«Орион-5»	1000/340	24	24/8,1
«Саиф»	1600/500	6	9/3
«Ригель»	1600/500	24–36	34,5–51,8

В табл. 3 приведены производительности вычислительных модулей на задачах символьной обработки данных, использующих битовые преобразования, и задачах математической физики на основе арифметики с плавающей запятой одинарной точности.

Таблица 3

Вычислительный модуль	Производительность вычислительных модулей	
	Символьная обработка данных (Топ/с)	Математическая физика, арифметика с плавающей запятой (Тфлопс)
«Орион-5»	116	1/0,34
«Саиф»	199,6	1,6/0,5
«Ригель»	199,6	1,6/0,5

В табл. 4 приведены суммарные скорости передачи данных между кристаллами ПЛИС и блоками распределенной памяти, между ПЛИС в пределах одного вычислительного модуля и других вычислительных модулей.

Таблица 4

Скорость передачи данных	Вычислительные модули		
	«Орион-5»	«Саиф»	«Ригель»
С блоками распределенной памяти (Гбит/с)	12,8	12,8	12,8
Между ПЛИС вычислительного поля (Тбит/с)	1,2	1,0	1,0
С другими вычислительными модулями (Тбит/с)	1,2	1,0	1,0

Таким образом, вычислительные модули нового поколения «Саиф» и «Ригель» на основе ПЛИС семейства Virtex 6 открывают перспективы для построения вычислительных систем более высокой производительности при сохранении стоимости системы по сравнению с РВС на основе вычислительного модуля «Орион-5». В то же время вычислительные модули обладают достаточной автономностью и могут легко комплексоваться с персональным компьютером типа IBM PC в качестве ускорителей и использоваться при решении различных задач.

## 2. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РВС НА ОСНОВЕ ОТКРЫТОЙ МАСШТАБИРУЕМОЙ АРХИТЕКТУРЫ

Для вычислительных модулей нового поколения «Саиф» и «Ригель» сохраняется преемственность принципов программирования РВС. Программирование всех рассмотренных вычислительных модулей и систем на их основе осуществляется с помощью единого комплекса системного программного обеспечения, поддерживающего структурно-процедурные методы организации вычислений. Программирование РВС отличается от программирования суперЭВМ традиционной архитектуры, поскольку включает организацию не только параллельных процессов и потоков данных, но и программирование структуры вычислительной системы в поле логических ячеек ПЛИС. Комплекс программного обеспечения вычислительных модулей предоставляет прикладному программисту следующие возможности:

- программирование как структурной, так и процедурной составляющих на языке высокого уровня без участия высококвалифицированного схемотехника;
- реконфигурация прикладных программ при перераспределении вычислительного ресурса РВС;

- обеспечение совместимости и переносимости проектов между PBC разных архитектур;
- масштабирование прикладной задачи при увеличении ресурса;
- удаленное использование вычислительных ресурсов PBC.

Созданный комплекс программного обеспечения [3] по функциональному назначению разделяется на комплекс средств разработки прикладных программ и комплекс средств управления и администрирования ресурсов PBC.

Средства разработки прикладных программ содержат: транслятор языка ассемблера; транслятор языка программирования PBC высокого уровня COLAMO; интегрированную среду разработки прикладных задач IDE, поддерживающую языки ассемблера и COLAMO; синтезатор масштабируемых параллельно-конвейерных решений, оперирующий библиотекой IP-ядер и интерфейсов.

Язык программирования высокого уровня COLAMO [3, 4, 5] обеспечивает поддержку создания как структурной, так и процедурной составляющих прикладной программы, реконфигурацию прикладных задач без участия высококвалифицированного схемотехника за счет неявного описания параллелизма и переносимость прикладных задач между PBC разных архитектур за счет использования файла описания архитектуры PBC и элементов библиотеки масштабируемых IP-ядер.

Транслятор COLAMO v.2.0 осуществляет трансляцию процедурной составляющей программы, организующей потоки данных, в язык ассемблера Argus v.3.0 и создание структурной составляющей в объектном представлении, которая автоматически передается в среду разработки масштабируемых параллельно-конвейерных процедур Fire!Constructor для синтеза конфигурационных файлов ПЛИС на языке VHDL.

Фундаментальным типом вычислительной структуры в языке COLAMO является конструкция «кадр». Кадром является программно-неделимый компонент, представляющий собой совокупность арифметико-логических команд, выполняемых на различных элементарных процессорах, обладающих распределенной памятью и соединенных между собой в соответствии с информационной структурой алгоритма таким образом, что вычисления производятся с максимально возможными параллелизмом и асинхронностью.

Кадр фактически определяет вычислительную структуру и потоки данных в PBC в данный момент времени. При этом все операции

в теле кадра выполняются асинхронно с максимальным параллелизмом, а последовательность смены кадров однозначно определяется программистом.

В языке отсутствуют явные формы описания параллелизма. Распараллеливание достигается с помощью объявления типов доступа к переменным и индексации элементов массивов. Для исключения конфликтов одновременного чтения и записи ячеек памяти в пределах текущего кадра используется широко распространенное в языках потока данных правило единственной подстановки: переменная, хранящаяся в памяти, может получить значение в кадре только один раз.

Для обращения к данным используются два основных метода доступа: параллельный доступ (задаваемый типом Vector) и последовательный доступ (задаваемый типом Stream). На рис. 3 представлены программы, являющиеся граничными примерами извлечения параллелизма, и графы синтезируемых вычислительных структур.

Тип доступа Stream указывает на последовательную обработку элементов одномерного массива, а тип Vector позволяет обрабатывать элементы одномерного массива одновременно.

Многомерные массивы состоят из множества измерений, каждое из которых может иметь последовательный или параллельный тип доступа, задаваемый ключевым словом Stream или Vector соответственно.

Применение неявного описания параллелизма за счет задания типа доступа позволяет достаточно просто управлять степенью распараллеливания программы на уровне описания структур данных и дает возможность программисту максимально просто описывать различные виды параллелизма в достаточно сжатом виде.

Трансляция программы на языке высокого уровня COLAMO состоит в создании схемотехнической конфигурации вычислительной системы (структурной составляющей) и параллельной программы, управляющей потоками данных (поточковой и процедурной составляющих).

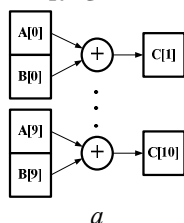
Операторы и функции языка (сумматоры, умножители, функции сравнения, тригонометрические функции и др.), используемые в тексте параллельной программы, имеют готовые схемотехнические решения. Данные решения разрабатываются специалистами-схемотехниками в интегрированной среде разработки цифровых устройств ISE фирмы XILINX или с ней совместимых и включаются в библиотеку транслятора

языка COLAMO и библиотеку стандартных примитивов среды Fire!Constructor.

```

VAR A,B,C: Integer [10 : Vec-
tor] Mem;
VAR I : Number;
CADR SummaVector;
For I := 0 to 9 do
C[I] :=A[I]+B[I];
ENDCADR;

```

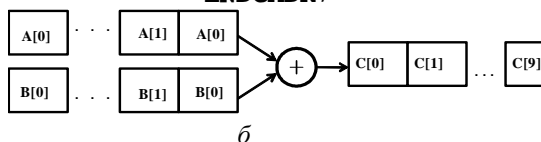


a

```

VAR A,B,C : Integer [10 :
Stream] Mem;
VAR I : Number;
CADR SummaStream;
For I := 0 to 9 do
C[I] :=A[I]+B[I];
ENDCADR;

```



b

**Рис. 3.** Параллельное и последовательное сложение массивов: *a* – тип доступа Vector; *b* – тип доступа Stream

В процессе работы транслятора языка COLAMO формируется информационный граф прикладной задачи из текста параллельной программы, где операторы и функции языка по определенным правилам заменяются соответствующими блоками или группами блоков из библиотеки стандартных примитивов.

Синтезированный вычислительный граф задачи передается в среду разработки вычислительных структур Fire!Constructor для укладки на множество ПЛИС PBC и обеспечения синхронизации между ПЛИС [5]. Одной из задач среды является формирование разбиения информационного графа прикладной задачи на непересекающиеся подграфы, каждый из которых будет структурно реализован в кристаллах ПЛИС выбранной PBC.

Процесс синтеза результата разбиения информационных графов прикладных задач состоит из следующих этапов:

- решение задачи разбиения (компоновки) узлов информационного графа прикладной задачи на непересекающиеся подграфы, каждый

из которых будет размещён в соответствующем БМ;

- решение задачи размещения и трассировки для узлов информационного графа в каждом БМ в отдельности и задачи трассировки связей между БМ;

- синтез файлов VHDL-описаний и файлов временных и топологических ограничений для каждой ПЛИС, каждого БМ выбранной PBC.

Среда Fire!Constructor упрощает создание масштабируемых структурных решений и сокращает время разработки за счет автоматизированного выполнения следующих трудоемких процедур:

- согласования входов и выходов совместно работающих ПЛИС (ucf-файлов);

- автоматической синхронизации информационных потоков при размещении функциональных устройств в едином вычислительном контуре, расположенном в различных кристаллах ПЛИС;

- автоматического обеспечения сбалансированного размещения функциональных устройств по различным ПЛИС.

Технология создания прикладных программ для PBC и общая взаимосвязь транслятора языка COLAMO, среды Fire!Constructor и синтезатора конфигурации ПЛИС в рамках комплекса системного программного обеспечения при создании многокристального схмотехнического решения для PBC представлена на рис. 4.

Такой подход к программированию реконфигурируемых вычислительных систем позволяет освободить программиста от построения графа задачи в виде функциональных библиотек в среде Fire!Constructor и синхронизации потоков данных в PBC, сократив время создания параллельных программ для PBC в 3–10 раз, и исключить участие специалиста-схмотехника при разработке параллельных прикладных программ.

Язык структурно-процедурного программирования Argus представляет собой низкоуровневый язык (ассемблер), предназначенный для описания процедурной составляющей прикладной параллельной программы PBC [3, 4]. Программа на языке Argus организует потоки данных на уровне команд контроллеров распределенной памяти, обеспечивая их синхронизацию.

Интегрированная среда разработки Argus IDE предназначена для интерактивной разработки параллельных программ на языках высокого уровня COLAMO и языке ассемблера Argus в едином языковом пространстве.





Рис. 4. Технология создания прикладных программ для PBC

Среда Argus IDE, объединяя в своем составе трансляторы языков COLAMO и Argus, обеспечивает эффективную разработку масштабируемых параллельных программ для PBC.

Созданное параллельное решение прикладной задачи в виде загрузочного модуля PBC с помощью драйвера загружается в вычислительный модуль PBC. Драйвер вычислительных модулей обеспечивает программную поддержку функций непосредственного доступа к высокоскоростному аппаратному интерфейсу, поддерживающему пакетные режимы работы и обеспечивающему механизмы прямого доступа к физической памяти управляющего компьютера.

Для удаленного доступа и управления вычислительными ресурсами PBC разработана система удаленного доступа, которая состоит из сервера, обрабатывающего удаленные заявки на использование вычислительных ресурсов и поддерживающего очередь заявок, и клиента, формирующего заявки на основе команд пользователя. К функциям системы удаленного доступа относятся функции включения, выключения, остановки и запуска как отдельных вычислительных модулей, так и стоек и PBC в целом.

Созданный комплекс программного обеспечения позволяет создавать эффективные прикладные программы для PBC при решении задач различных предметных областей, обеспечивает удобство программирования и сокращает время разработки прикладного решения в 3–

5 раз, обеспечивая при этом автоматизированный перенос структурного решения с одной архитектуры PBC на другую.

## ЗАКЛЮЧЕНИЕ

PBC являются перспективным направлением развития высокопроизводительной вычислительной техники, которое, в отличие от кластерных суперЭВМ, предоставляет пользователю возможность создавать в базовой архитектуре виртуальные специализированные вычислители, структура которых адекватна структуре решаемой задачи. Это, в свою очередь, обеспечивает высокую эффективность вычислений и близкий к линейному рост производительности при наращивании вычислительного ресурса.

Следует отметить, что переход на новую компоновку модулей, позволившую сосредоточить в пределах вычислительных модулей «Орион-5» и «Ригель» высотой 1U мощный вычислительный ресурс на основе ПЛИС, обеспечивает удельную производительность PBC на уровне лучших мировых показателей для суперЭВМ с кластерной архитектурой.

Анализ тенденции развития ПЛИС и построения реконфигурируемых вычислительных систем на их основе показывает, что PBC обладают высокой удельной производительностью при решении задач различных классов и могут служить основным средством для создания суперЭВМ нового поколения.

## СПИСОК ЛИТЕРАТУРЫ

1. **Левин И. И.** Реконфигурируемые вычислительные системы с открытой масштабируемой архитектурой // Параллельные вычисления и задачи управления, РАСО'2010: Тр. 5-й Междунар. конф.. М.: Учреждение Рос. акад. наук Институт проблем управления им. В. А. Трапезникова. РАН, 2010. С. 83–95.
2. **Каляев А. В., Левин И. И.** Модульно-наращиваемые многопроцессорные системы со структурно-процедурной организацией вычислений. М.: Янус-К, 2003. 380 с.
3. Реконфигурируемые мультиконвейерные вычислительные структуры / И. А. Каляев [и др.]; под общ. ред. И. А. Каляева. Ростов-на-Дону: Изд-во ЮНЦ РАН, 2009. 344 с.
4. **Каляев И. А., Левин И. И.** Семейство реконфигурируемых вычислительных систем с высокой реальной производительностью // Параллельные вычислительные технологии (ПАВТ'2009): Тр. Междунар. науч. конф. Нижний Новгород: электронное издание НГУ имени Н. И. Лобачевского, 2009. С. 186–196.
5. Развитие аппаратной платформы реконфигурируемых вычислительных систем / Н. Н. Дмитренко [и др.] // Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Тр. Междунар. суперкомпьютерн. конф. М.: Изд-во МГУ, 2010. С. 315–320.

## ОБ АВТОРАХ

**Каляев Игорь Анатольевич**, дир. науч.-иссл. ин-та многопроцес. выч. систем Южного федер. ун-та. Дипл. инж.-системотехн. (ТРТИ, 1980). Член-кор. РАН (НИИ МВС ТРТУ, 2003), д-р техн. наук, проф. Иссл. в обл. многопроцессорных вычислительн. и управляющих систем, интеллектуальных систем управления робототехническими комплексами.

**Левин Илья Израилевич**, зам. директора по науке того же ин-та. Дипл. инж.-констр.-технол. ЭВА (ТРТИ, 1984). Д-р техн. наук (НИИ МВС ТРТУ, 2005). Иссл. в обл. высокопроизводительных многопроцессорных вычислительных и управляющих систем, систем обработки информации, принятия решений и управления специального назначения.

**Семерников Евгений Андреевич**, зав. лаб. Южного научного центра РАН. Дипл. инж.-электрик (ТРТИ, 1976). Канд. техн. наук (НИИ МВС ТРТИ, 1986). Иссл. в обл. высокопроизводительных многопроцессорных вычислительных и управляющих систем, систем обработки информации, принятия решений и управления специального назначения.

**Дордопуло Алексей Игоревич**, ст. науч. сотр. Южного научного центра РАН. Дипл. инженер по организации и технологии защиты информации (ТРТУ, 2000). Канд. техн. наук (ТРТУ, 2003). Иссл. в обл. высокопроизводительных многопроцессорных вычислительных и управляющих систем, систем обработки информации, принятия решений и управления специального назначения.