

УДК 004.382.2, 004.4.42

СРЕДСТВА ВЫСОКОУРОВНЕВОГО СИНТЕЗА ПАРАЛЛЕЛЬНЫХ ПРОГРАММ ОБРАБОТКИ ДАННЫХ И УПРАВЛЕНИЯ ДЛЯ РЕКОНФИГУРИРУЕМЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

И.И. Левин

Южный федеральный университет
Россия, 344006, Ростов-на-Дону, ул. Б. Садовая, 105
E-mail: levin@superevm.ru

А.И. Дордопуло

Общество с ограниченной ответственностью «НИЦ супер-ЭВМ и нейрокомпьютеров»
Россия, 347900, Таганрог, пер. Итальянский, 106
E-mail: dordopulo@superevm.ru

В.А. Гудков

*Федеральное государственное автономное образовательное учреждение высшего образования
Южный федеральный университет*
Россия, 344006, Ростов-на-Дону, ул. Б. Садовая, 105
E-mail: gudkov@superevm.ru

А.А. Гуленок

Общество с ограниченной ответственностью «НИЦ супер-ЭВМ и нейрокомпьютеров»
Россия, 347900, Таганрог, пер. Итальянский, 106
E-mail: gulenok@superevm.ru

Ключевые слова: задачи управления, высокоуровневый синтез, HLS, трансляция программ, язык C, редукция производительности, реконфигурируемая вычислительная система, программирование многопроцессорных вычислительных систем.

Аннотация: В докладе рассматриваются средства высокоуровневого синтеза, преобразующие последовательную программу на языке высокого уровня C в конфигурационные файлы программируемых логических интегральных схем реконфигурируемой вычислительной системы. В отличие от известных средств, программный комплекс "Тесей" строит информационный граф последовательной программы и преобразует его в максимально-параллельную кадровую структуру, которая формальными методами в автоматическом режиме без участия пользователя адаптируется к заданной конфигурации реконфигурируемой вычислительной системы. Отличительной особенностью подхода является существенно меньшее по сравнению с распараллеливающими компиляторами число анализируемых вариантов, что позволяет сократить время преобразования (портации) последовательных программ. Разработанные средства синтеза применимы, в том числе, для прикладных задач управления, содержащих связанные информационной зависимостью подзадачи с разной степенью параллелизма. Для ряда прикладных задач представлены результаты синтеза решений на архитектуру различных реконфигурируемых вычислительных систем.

1. Введение

Программируемые логические интегральные схемы (ПЛИС или FPGA) в последнее время все чаще используются как самостоятельная вычислительная платформа [1], успешно конкурирующая в области высокопроизводительных вычислений с микропроцессорами и графическими ускорителями. Возможность создания в архитектуре ПЛИС специализированного вычислительного устройства, учитывающего информационные зависимости задачи, позволяет достигать многократного выигрыша во времени решения даже при существенной разнице с микропроцессорами в тактовой частоте работы. Программирование ПЛИС и реконфигурируемых вычислительных систем (РВС) на их основе [2] требует от прикладных программистов обширных и глубоких знаний схемотехники и архитектуры системы. Развитие методов эффективного программирования РВС, доступных для широкого круга разработчиков, во многом связано с созданием средств высокоуровневого синтеза [3-8], преобразующих разработанные для процессоров программы на языках высокого уровня (в том числе параллельные) в конфигурационные файлы ПЛИС.

2. Средства высокоуровневого синтеза

Помимо традиционных инструментов схемотехнического проектирования, таких как Xilinx Vivado или Intel Quartus Prime, в последнее время все большую популярность приобретают средства высокоуровневого синтеза (High Level Synthesis) или HLS-компиляторы [3,4], которые преобразуют программу на одном из языков высокого уровня в конфигурационные файлы специализированных аппаратных средств на языках описания аппаратуры HDL. Подробный обзор возможностей, ограничений и отличительных характеристик академических (DWARV, BAMBU, LEGUP) и коммерческих (CatapultC, Vivado HLS[6], Vivado Vitis[7]) средств высокоуровневого синтеза представлен в [3]. Большинство рассмотренных в [3,4] HLS-компиляторов преобразуют вычислительно трудоемкий фрагмент программы на языке C в сложный функциональный блок (IP-ядро) – цифровой автомат или специализированный процессор. Выигрыш во времени вычислений для синтезированного решения по сравнению с универсальным процессором обеспечивается за счет свойств архитектуры ПЛИС и особенностей реализации информационных зависимостей. Применение HLS-компиляторов не гарантирует [8] эффективной реализации вычислений в ПЛИС для любой программы на языке C. Как правило, HLS-компиляторы предназначены для быстрой разработки проектов, где важна скорость создания конфигурационных файлов ПЛИС, а не эффективность кода и/или сокращение аппаратных затрат. Автоматизация создания IP-ядра облегчает портацию вычислений в архитектуру ПЛИС, но организация потоков данных возлагается на программиста, поскольку средства масштабирования и синхронизации решения (хотя бы в пределах одного кристалла) отсутствуют. Поэтому для повышения эффективности синтезируемого решения используется выполняемая программистом ручная разметка кода (например, директивы `#pragma` в Vivado HLS и Vitis), что требует от разработчика глубоких знаний о синтезируемом решении. Для многокристальных РВС масштабирование и согласование IP-ядер для поиска рационального решения многократно усложняются и становятся похожими на автоматическое распараллеливание, вычислительная сложность которого очень велика. Поэтому для РВС, содержащих множество ПЛИС, связанных пространственной коммутационной системой [1, 2, 8], разработан комплекс средств высокоуровневого синтеза «Тесей», обеспечивающий преобразование входной программы без ручной разметки кода директивами `#pragma` и автоматическую

синхронизацию информационных и управляющих сигналов для многокристалльных решений.

3. Выскоуровневый синтез решений в программном комплексе «Тесей»

Схема взаимодействия программ в комплексе средств высокоуровневого синтеза «Тесей» представлена на рис. 1.

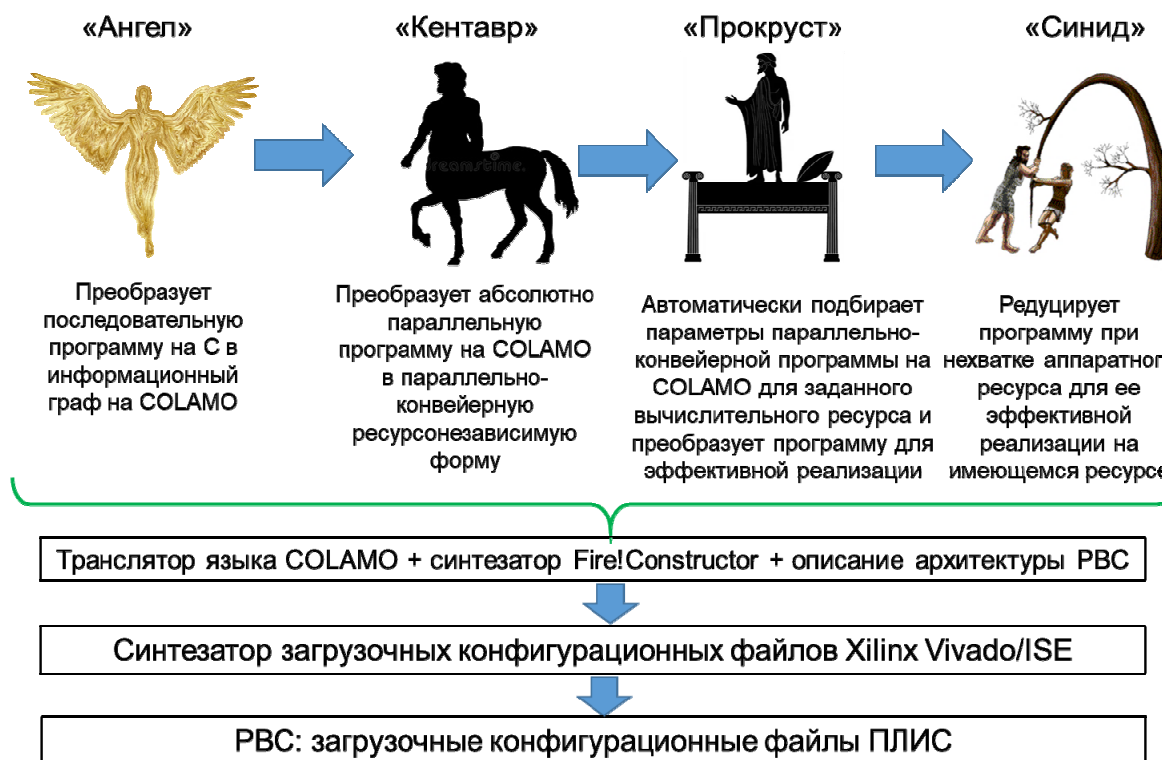


Рис. 1 Структура взаимодействия компонентов программного комплекса.

Входная последовательная программа на языке C (стандарт ISO/IEC 9899:1999) последовательно преобразуется в информационный граф и кадровую структуру, которая адаптируется компонентами комплекса к области доступного аппаратного ресурса PBC [5].

Транслятор «Ангел» преобразует входную программу на языке C в максимально параллельную форму - информационный граф, анализирует информационные зависимости, устранение нарушения правил однократного присваивания с помощью растягивания и расщепления переменных и формирует генераторы чтения-записи потоков данных. «Кентавр» преобразует максимально параллельную кадровую структуру в масштабируемую параллельно-конвейерную форму и выделяет функциональные графы, параметризуемые аппаратным ресурсом. «Прокруст» подбирает параметры быстродействия кадровой структуры для ее рациональной реализации в архитектуре заданной PBC. «Синид» выполняет редукцию производительности при нехватке аппаратного ресурса для структурной реализации базового подграфа.

Результатом трансляции является программа на языке высокого уровня COLAMO, соответствующая доступному ресурсу и заданному уровню реальной

производительности. Разработанные ранее транслятор языка программирования COLAMO и синтезатор многокристальных решений Fire!Constructor транслируют эту программу на языке COLAMO в конфигурационные файлы ПЛИС многокристальных PBC. Синтезатор системы автоматизированного проектирования Xilinx Vivado создает загрузочные конфигурационные файлы (*.bit) для каждого кристалла.

4. Экспериментальные результаты преобразования прикладных задач

Оценка эффективности разработанного комплекса средств высокоуровневого синтеза проводилась сравнением результатов портации модельных задач, полученных прикладными программистами, и работающим в автоматическом режиме комплексом. Сравнение проводилось для пяти модельных задач (таблица 1): симметричный блочный алгоритм DES, хеш-функции MD5 и SHA-1 и задачи решения систем линейных алгебраических уравнений методом Гаусса и методом Якоби для трехдиагональных матриц. Заданный для программного комплекса уровень реальной производительности составлял 0,6, а достигнутый уровень определялся как отношение числа подграфов в решениях, полученных комплексом и прикладными программистами.

Таблица 1. Результаты преобразования модельных задач.

Задачи	DES	MD5	SHA-1	Якоби	Гаусс
PBC «Тайгета»					
Время портации, с	10836,62	10938,86	10841,66	13429,90	12316,38
Выигрыш	6,31	6,25	6,31	5,09	5,55
Уровень реальной производительности	0,63	0,63	0,86	0,86	0,86
PBC «Терциус-3»					
Время портации, с	25239,02	25344,79	25241,76	25914,02	26991,12
Выигрыш	3,28	3,27	3,28	3,20	3,07
Уровень реальной производительности	0,65	0,67	1,0	0,85	0,80

Каждая задача портировалась на персональном компьютере с процессором Intel (R) Core (TM) i7-8750H @2,2 ГГц и 16 Гб оперативной памяти, работающим под управлением операционной системы Windows 10 Pro, на аппаратные платформы PBC «Тайгета» [9] и «Терциус-3» [9]. При портации измерялись время преобразования кадровой структуры и время портации задачи. Время преобразования кадровой структуры комплексом определялось временем работы программы «Прокруст», а для прикладных программистов, исходя из практики решения задач, было принято равным двум восьмичасовым рабочим дням. Время портации учитывает время синтеза загрузочного конфигурационного файла ПЛИС, которое зависит от логической емкости ПЛИС и уровня заполнения кристалла. При заполнении кристаллов на уровне 90% оно составляет не менее трех часов для «Тайгеты» и семи часов для «Терциуса-3». Полученные программным комплексом решения проверялись на работоспособность запуском на соответствующей PBC.

5. Заключение

Применение разработанного комплекса средств высокоуровневого синтеза позволяет значительно сократить время преобразования кадровой структуры для многокристалльных РВС и находить рациональное решение (с эффективностью не ниже 50% от результатов прикладных программистов) за существенно меньшее (по сравнению с распараллеливающими компиляторами) число преобразований. В отличие от известных HLS-компиляторов, входная программа на языке C преобразуется автоматически, без ручной разметки кода или иных указаний пользователя, а результатом является многокристалльное решение с автоматической синхронизацией информационных и управляющих сигналов.

Список литературы

1. Kalyaev I.A., Levin I.I., Dordopulo A.I., Slasten L.M. FPGA-based reconfigurable computer systems // Proc. Science and Information Conference (SAI 2013). 2013. P. 148-155.
2. Levin I., Dordopulo A., Fedorov A., Kalyaev I. Reconfigurable computer systems: from the first FPGAs towards liquid cooling systems // Supercomputing Frontiers and Innovations. 2016. Vol. 3(1). P. 22-40 DOI: 10.14529/jsfi160102.
3. Nane R., et al. A Survey and Evaluation of FPGA High-Level Synthesis Tools // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2016. Vol. 35, No. 10. P. 1591-1604. DOI: 10.1109/TCAD.2015.2513673.
4. Numan M.W., Phillips B.J., Puddy G.S., Falkner K. Towards Automatic High-Level Code Deployment on Reconfigurable Platforms: A Survey of High-Level Synthesis Tools and Toolchains // IEEE Access. 2020. Vol. 8. P. 174692-174722. DOI: 10.1109/ACCESS.2020.3024098.
5. Дордопуло А.И., Левин И.И., Гудков В.А., Гуленок А.А. Программные средства высоко-уровневого синтеза для многокристалльных реконфигурируемых вычислительных систем // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2022. Т. 11, № 3. С. 5-21. DOI: 10.14529/cmse220301.
6. Make Slow Software Run Fast with Vivado HLS <https://www.xilinx.com/publications/xcellonline/run-fast-with-Vivado-HLS.pdf> (дата обращения 11.01.2024).
7. Vitis Unified Software Platform Documentation. Application Acceleration Development. https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_2/ug1393-vitis-application-acceleration.pdf (дата обращения 11.01.2024).
8. Тарасов И. Проектирование для ПЛИС Xilinx с применением языков высокого уровня в среде Vivado HLS // Компоненты и технологии. 2013. № 12. <https://kit-e.ru/fpga/vivado-hls/>.
9. Вычислительные блоки НИЦ супер-ЭВМ и нейрокompьютеров (<http://superevm.ru/index.php?page=vychislitelnye-bloki>) (дата обращения 11.01.2024).