DOI: 10.17117/na.2015.11.03.138

<http://ucom.ru/doc/na.2015.11.03.138.pdf>

Поступила (Received): 13.11.2015

Егунов В.А., Каменнов А.А. Аппаратная реализация преобразования отражения с использованием HDL – языков

Egunov V.A., Kamennov A.A.
Hardware implementation of reflection
transformations using HDL – languages

В статье рассматриваются вопросы аппаратной реализации преобразования отражения на программируемых логических интегральных схемах. В качестве базовых алгоритмов предлагаются алгоритмы дискретных линейных преобразований. Анализируются вопросы использования HDL – языков для реализации подобных преобразований

Ключевые слова: ПЛИС, дискретные линейные преобразования, HDL. VHDL, Verilog

Егунов Виталий Алексеевич

Кандидат технических наук, доцент
Волгоградский государственный технический университет
г. Волгоград, пр. им. Ленина, 28

Каменнов Александр Александрович

Студент
Волгоградский государственный технический университет
г. Волгоград, пр. им. Ленина, 28

The article discusses the issues of hardware implementation of reflection transformations on programmable logic devices. Algorithms of discrete linear transformations are proposed as base. Analyzes the use of HDL languages to implement such transformations

Key words: PLD, discrete linear transformations, HDL. VHDL, Verilog

Egunov Vitaly Alekseevich

Candidate of Engineering Sciences, Associate Professor
Volgograd state technical university
Volgograd, Lenina ave., 28

Kamennov Alexander Alexandrovich

Student
Volgograd state technical university
Volgograd, Lenina ave., 28

Преобразование отражения Хаусхолдера является одним из наиболее востребованных в вычислительной математике, т.к. является эффективным методом выполнения QR – разложения, которое может быть использовано при решении ряда ключевых задач. К таким задачам, например, можно отнести поиск собственного и сингулярного разложений, решение систем линейных алгебраических уравнений и ряда других. В многочисленной литературе можно найти массу программ реализации преобразования отражения на различных алгоритмических языках программирования. В связи с тем, что в последние годы все большее распространение получают параллельные вычислительные системы, актуальным становится вопрос разработки соответствующих параллельных ал-

горитмов. Алгоритмы реализации преобразования отражения достаточно хорошо распараллеливаются, т.к. имеют матричную природу, следовательно можно получить эффективную программную реализацию данных алгоритмов, например, для многоядерных процессоров, вычислительных кластеров и т.д.

Также в последние годы все шире стали применяться ускорители, построенные на базе микросхем программируемой логики (ПЛИС). Достигать ускорения вычислений они позволяют в основном за счет специализации вычислений под конкретную задачу и возможности строить для них очень эффективные конвейеры. Применение таких ускорителей ограничивается рядом сложностей, среди которых можно выделить, прежде всего, доступность таких устройств, сложность сопряжения их с настольными системами и сложность создания специализированных схем для решения задач.

Одно из препятствий к использованию подобных ускорителей является то обстоятельство, что не любой алгоритм может быть реализован в виде эффективного аппаратного ускорителя. Для этого алгоритм должен удовлетворять ряду требований.

Базовый алгоритм отражения в целом не удовлетворяет требованиям, предъявляемым к алгоритмам, эффективно реализуемым в виде ускорителей на ПЛИС. Известен ряд реализаций преобразования отражения в виде дискретных линейных преобразований (ДЛП) [1, 2]. Под ДЛП – реализацией линейного оператора A понимается разложение исходного линейного оператора в бесконечное произведение матриц, таких что их произведение стремится к исходному оператору. Матрицы ДЛП подбираются таким образом, что преобразование в основном сводится к последовательности сложений и сдвигов, что значительно упрощает аппаратную реализацию данного преобразования. Число ДЛП – итераций выбирается таким образом, чтобы достигалась требуемая точность результатов. Подобные алгоритмы часто называются алгоритмами «цифра за цифрой», т.к. на каждой итерации уточняется очередная двоичная цифра результата.

С лета 2008 года ведётся разработка стандарта межплатформенного параллельного языка программирования OpenCL. Цель разработки – перенос параллельных алгоритмов с платформы на платформу с минимальными затратами. В основу OpenCL положен язык C с добавлением специфических для параллельных программ расширений. Фактически OpenCL ориентирован на использование сопроцессоров. Один из ведущих мировых производителей ПЛИС – компания Altera – в ноябре 2012 года опубликовала первые результаты разработки компилятора OpenCL для создания ядер в своих ПЛИС. Разработка OpenCL ядер для реализации алгоритмов ДЛП на ПЛИС описан в [4, 5]. В работах приводится программный код OpenCL, который использовался для загрузки ПЛИС семейства CycloneII и StratixV, что связано с доступностью лицензий и особенностями использованных средств синтеза. Анализировались конвейерные реализации и реализации без конвейера, получены параметры сложности и пропускной способности полученных устройств.

Другим вариантом разработки ПЛИС является использованием HDL (hardware description language) языков – языков описания аппаратуры. К наиболее популярным HDL – языкам можно отнести VHDL (Very high speed integrated circuits) и Verilog. Данные языки являются международным стандартом и используются в качестве основы лингвистического обеспечения как в системах анализа (моделирования), так и в системах синтеза цифровой аппаратуры. Ведущие мировые САПР БИС поддерживают описания на HDL – языках.

Для аппаратной реализации преобразования на ПЛИС с использованием HDL – языка программирования необходимо выполнить следующие действия:

- разработать модификацию исходного преобразования, удовлетворяющую требованиям к аппаратурно – ориентированным алгоритмам;
- используя полученный алгоритм, синтезировать схему соответствующего преобразователя;
- описать полученный преобразователь с помощью HDL – языка; при этом необходимо учитывать особенности используемых САПР и ПЛИС;
- осуществить моделирование, доработку проекта.

Список используемых источников:

1. Духнич Е.И., Егунов В.А., Деревенсков С.О. Универсальный матричный алгоритм, ориентированный на систолическую реализацию // Автоматиз. процессов упр. и обраб. информ. Вологда, 1998. С. 54-58.
2. Духнич Е.И., Егунов В.А. Алгоритм быстрого преобразования Хаусхолдера // Концептуальное проектирование в образовании, технике и технологии. Волгоград, 1999. С. 71-75.
3. Andreev A., Egunov V., Zharikov D., Shapovalov O., Artuh S., Doukhnitch E. Evaluation of hardware implementations of cordic-like algorithms in fpga using opencl kernels // Communications in Computer and Information Science. 2014. T. 466 CCIS. С. 228-242.
4. Egunov V.A., Artyukh S.V. FPGA as coprocessors in solving problems of linear algebra // Инновационные информационные технологии. 2014. № 2. С. 93-97.