7universum.com
UNIVERSUM:
ТЕХНИЧЕСКИЕ НАУКИ

БЛОК ФОРМИРОВАНИЯ ХЭШ-ФУНКЦИЙ КАК СРЕДСТВО ЛОКАЛИЗАЦИИ ВЫЧИСЛЕНИЙ В ПАРАЛЛЕЛЬНОЙ ПОТОКОВОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ

Змеев Дмитрий Николаевич
научный сотрудник, Институт проблем проектирования в микроэлектронике Российской академии наук,
124365, Россия, г. Москва, Зеленоград, ул. Советская, дом 3

Кузьмин Егор Никитович
инженер-исследователь, Институт проблем проектирования в микроэлектронике Российской академии наук,
124365, Россия, г. Москва, Зеленоград, ул. Советская, дом 3

Левченко Николай Николаевич
канд. техн. наук, заведующий отделом, Институт проблем проектирования в микроэлектронике Российской академии наук,
124365, Россия, г. Москва, Зеленоград, ул. Советская, дом 3

Окунев Анатолий Семенович
канд. техн. наук, ведущий научный сотрудник, Институт проблем проектирования в микроэлектронике Российской академии наук,
124365, Россия, г. Москва, Зеленоград, ул. Советская, дом 3
E-mail: oku@ippm.ru

A UNIT OF FORMING OF HASH-FUNCTIONS AS A MEANS FOR THE LOCALIZATION OF COMPUTATIONS IN THE PARALLEL DATAFLOW COMPUTING SYSTEM

Dmitry Zmejev
Research scientist,
Institute for Design Problems in Microelectronics of Russian Academy of Sciences,
124365, Russia, Moscow, Zelenograd, Sovetskaya Street, 3

Блок формирования хэш-функций как средство локализации вычислений в параллельной потоковой вычислительной системе //
Universum: Технические науки : электрон. научн. журн. Змеев Д.Н. [и др.]. 2015. № 10 (21) .
URL: http://7universum.com/ru/tech/archive/item/2672

Egor Kuzmin
Research engineer,
Institute for Design Problems in Microelectronics of Russian Academy of Sciences,
124365, Russia, Moscow, Zelenograd, Sovetskaya Street, 3

Nikolay Levchenko
Candidate of Engineering sciences, head of the department,
Institute for Design Problems in Microelectronics of Russian Academy of Sciences,
124365, Russia, Moscow, Zelenograd, Sovetskaya Street, 3

Anatoly Okunev
Candidate of Engineering sciences, a leading research scientist,
Institute for Design Problems in Microelectronics of Russian Academy of Sciences,
124365, Russia, Moscow, Zelenograd, Sovetskaya Street, 3

АННОТАЦИЯ

Современные вычислительные системы кластерного типа демонстрируют низкую реальную производительность на большом круге актуальных задач, что ставит вопрос об изменении модели вычислений. По мнению специалистов, изменение модели вычислений тем более необходимо для проектируемых суперкомпьютерных систем.

В Институте проблем проектирования в микроэлектронике Российской академии наук ведутся работы над проектом параллельной потоковой вычислительной системы (ППВС), которая реализует новую потоковую модель вычислений с динамически формируемым контекстом. Данная модель вычислений обладает рядом преимуществ по сравнению с применяемыми в традиционных вычислительных системах моделях вычислений и основана на активации вычислительных квантов по готовности данных.

Фактически основным методом управления вычислениями в ППВС является локализация вычислений по вычислительным ядрам, а также разбиение задачи на этапы (локализация во времени). Эти методы реализуются с помощью задаваемых пользователем хэш-функций, для вычисления которых необходима эффективная аппаратная поддержка.

Функция распределения задается пользователем вместе с программой — обычно это одна формула (зависящая от полей ключа), возможно, своя для каждого узла или группы узлов. Ее следует выбирать так, чтобы: а) минимизировать количество обменов между ядрами и б) обеспечить относительную равномерность загрузки ядер.

Аппаратура блока формирования хэш-функций поддерживает аппаратную выработку следующих хэш-функций, используемых для конкретных актуальных задач: ZIP, NORM, BLK, STD, FLD.

Для исследования работы различных хэш-функций было реализовано несколько вариантов RTL-описаний этих хэш-функций.

В дальнейшем будет продолжено расширение функциональности блока формирования хэш-функций для поддержки многозадачного режима работы, созданы аппаратные реализации новых хэш-функций, что позволит увеличить производительность вычислительной системы.

ABSTRACT

Modern clusters demonstrate low real performance on a large range of actual tasks. It raises the question of changing the computing model. According to experts, changing of computing model is especially necessary for the planned supercomputer systems.

The Institute for Design Problems in Microelectronics of Russian Academy of Sciences is working on a project of the parallel dataflow computing system (PDCS) which implements a new dataflow computing model with dynamically formed context. This computing model has a number of advantages compared to a traditional ones used in computer systems and is based on the activation of computational quantums by data availability.

Actually, the main method of computation management in the PDCS is a localization of computations by computational cores, and partitioning tasks into stages (localization by time). These methods are implemented by using user-defined hash-functions. And the effective hardware support is needed to calculate these functions.

The distribution function is defined by the user with the program — usually it is a one formula (which depends on the fields of key), probably, is different for each node or group of nodes. It should be selected so as to: a) minimize the number of exchanges between the cores, and b) provide a relatively uniform loading of cores.

A unit of forming of hash-functions supports a hardware computation of the following hash-functions used for specific actual tasks: ZIP, NORM, BLK, STD, FLD.

To study the work of various hash-functions there were implemented several RTL-descriptions of these hash-functions.

In the future, the functionality of unit of forming of hash-functions will be expanded to support the multitasking, and a hardware realization of new hash-functions will be implemented, which will allow to increase the performance of the computing system.

Ключевые слова: вычислительное ядро ППВС «Буран», блок формирования хэш-функций, локализация вычислений, узлы программной и аппаратной выработки хэш-функций.

Keywords: computation core of PDCS "Buran", unit of forming of hash-functions, localization of computations, nodes of hardware and software forming of hash-functions.

Введение

Современные вычислительные системы кластерного типа демонстрируют низкую реальную производительность на большом круге актуальных задач, что ставит вопрос об изменении модели вычислений. По мнению специалистов, изменение модели вычислений тем более необходимо для проектируемых суперкомпьютерных систем.

В ИППМ РАН ведутся работы над проектом параллельной потоковой вычислительной системы (ППВС) «Буран» [2—6], которая реализует новую потоковую модель вычислений с динамически формируемым контекстом.

Данная модель вычислений обладает рядом преимуществ по сравнению с применяемыми в традиционных вычислительных системах моделях вычислений [1] и основана на активации вычислительных квантов по готовности данных. Вычислительный квант — это программа, которая выполняется до конца без привлечения дополнительной информации, то есть без приостановки процесса вычисления на подкачку дополнительных внешних данных. Различные вычислительные кванты между собой взаимодействуют и сохраняют состояние только через отправку сообщений, активирующих новые кванты. В узле-отправителе определяется и передаваемое значение, и адрес получателя. В этом состоит принципиальное отличие предлагаемого подхода от традиционного.

В состав каждого вычислительного ядра ППВС «Буран» входят: исполнительное устройство, процессор сопоставления, коммутатор токенов, блок хэширования. Между ядрами в системе передаются единицы информации в виде токенов. Токеном называется структура, в состав которой входит данное, контекст, определяющий положение операнда в виртуальном адресном пространстве задачи, а также набор служебных полей. Токены являются основными информационными объектами, с которыми работает аппаратура вычислительной системы. Коммутация между ядрами осуществляется по номеру вычислительного ядра. Этот номер вырабатывается в блоке хэширования на основе содержимого полей токена и настраиваемой программистом функции распределения.

Потоковая модель вычислений с динамически формируемым контекстом обладает рядом свойств, которые позволяют преодолевать проблемы, особенно остро проявляющиеся при повышении производительности компьютеров свыше петафлопса. Основным препятствием к использованию этой модели вычислений является необходимость в разработке специализированной аппаратуры.

Одним из определяющих факторов повышения производительности ППВС является выбор способа распределения виртуальных узлов по ядрам для каждой

конкретной задачи [5]. Фактически основным методом управления вычислениями в ППВС «Буран» является локализация вычислений по вычислительным ядрам, а также разбиение задачи на этапы (локализация во времени). Эти методы реализуются с помощью задаваемых пользователем хэш-функций, для вычисления которых необходима эффективная аппаратная поддержка.

Основной источник взаимодействий в ППВС — это передача токенов между ядрами. Токен посылается в ядро, номер которого вычисляется посредством функции распределения на основе адреса вычислительного узла, в который направлен токен.

Функция распределения задается пользователем вместе с программой — обычно это одна формула (зависящая от полей ключа), возможно, своя для каждого узла или группы узлов. Ее следует выбирать так, чтобы: а) минимизировать количество обменов между ядрами и б) обеспечить относительную равномерность загрузки ядер.

Блок формирования хэш-функций

В составе блока формирования хэш-функций (БХФ) (рис. 1) предусматривается реализация как узла программной выработки хэш-функций (ХФ), так и узел аппаратной выработки хэш-функций. Также предусмотрен быстрый выбор и смена нужной хэш-функции для каждого выдаваемого токена.

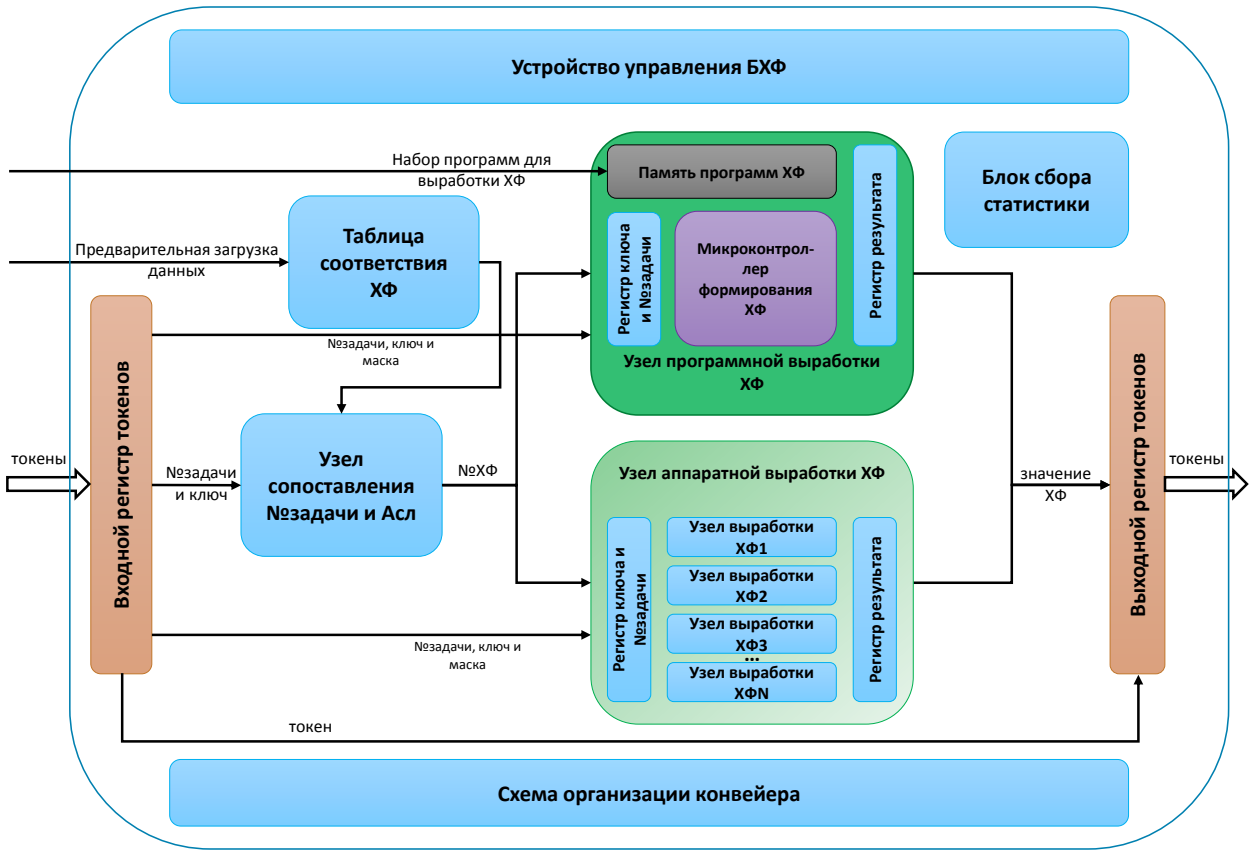

Рисунок 1. Функциональная схема блока хэш-функций

Перед стартом задачи происходит загрузка требуемыми значениями блоков «Таблица соответствия ХФ» и «Память программ ХФ» узла программной выработки хэш-функции. В «Таблицу соответствия ХФ» записываются номера задач, подзадач, номера программных узлов и соответствующие им номера аппаратных и программных хэш-функций. В «Память программ ХФ» помещаются микропрограммы выработки хэш-функций, которые при необходимости предварительно создает разработчик для своей программы, если базовый, реализованный аппаратно набор хэш-функций не позволяет достичь требуемой эффективности выполнения программы.

Работа блока хэш-функций начинается с прихода токена на «входной регистр токенов». Поля «№ задачи» и «Ключ» передаются в узел сопоставления, в котором происходит определение номера хэш-функций (№ ХФ) для данного токена. Согласно этому номеру хэш-функций поля «№ задачи», «ключ» и «маска» поступают либо в «Узел программной выработки ХФ», либо в «Узел аппаратной выработки ХФ». Если токен

использует аппаратную хэш-функций, то в «Узле аппаратной выработки ХФ» по номеру выбирается хэш-функция, которая преобразует исходные данные в значение, соответствующее номеру вычислительного ядра или номеру этапа в программе. В случае использования программной хэш-функций, из «Памяти программ ХФ» выбирается соответствующая программа, по которой микроконтроллер формирует значение, соответствующее номеру вычислительного ядра или номеру этапа в программе. Результирующее значение записывается в «Выходной регистр токенов» в поле, которое в зависимости от функции, выполняемой блоком выработки хэш-функции, может быть как «номером этапа» при работе внутри ядра, так и «номером вычислительного ядра» при пересылке токена между ядрами.

Работа блока выработки хэш-функции конвейеризирована для обеспечения высокого темпа приема и выдачи токенов.

Хэш-функции, использующиеся в системе ППВС

Ранее в работе [5] уже были рассмотрены некоторые хэш-функции. Аппаратура блока формирования хэш-функций поддерживает аппаратную выработку следующих хэш-функций, используемых для конкретных актуальных задач:

- функция ZIP. Принимает два двоичных аргумента и скрещивает их разряды: биты из первого аргумента идут на четные места результирующего аргумента, второго — на нечетные;
- функция NORM. «Нормализует» первый аргумент n и затем берет k старших разрядов результата. Под нормализацией понимается сдвиг аргумента влево так, чтобы его старшая единица вышла за пределы разрядной сетки числа;
- функция BLK. Обеспечивает наилучшую равномерность распределения однородного массива элементов с индексами i от 0 до N-1 на всех уровнях коммуникационной иерархии (в предположении, что иерархическая близость соответствует арифметической, иначе говоря, модули с близкими номерами близки коммуникационно);

- функция STD. Стандартная функция равномерно распределяет данные по вычислительным ядрам системы. Использует в своей работе все поля контекста;
- функция FLD. Обеспечивает распределение данных равными порциями (по N/K, где N — размерность данных задачи, а K — число вычислительных ядер) по вычислительным ядрам.

Исследование работы хэш-функций

Для исследования работы различных хэш-функций было реализовано несколько вариантов RTL-описаний этих хэш-функций, причем описания были созданы как на комбинаторной, так и на синхронной логике. При аппаратной реализации хэш-функций на синхронной логике входные сигналы для хэш-функций поступают одновременно, то есть не требуется дополнительно решать вопрос задержки того или иного входного сигнала относительно другого.

Для исследования работы хэш-функций были написаны тестовые последовательности, варианты реализаций хэш-функций были промоделированы в Quartus II и Icarus-Verilog. Ниже на рис. 2 и 3 приведены варианты временных диаграмм работы для хэш-функций zip и norm.

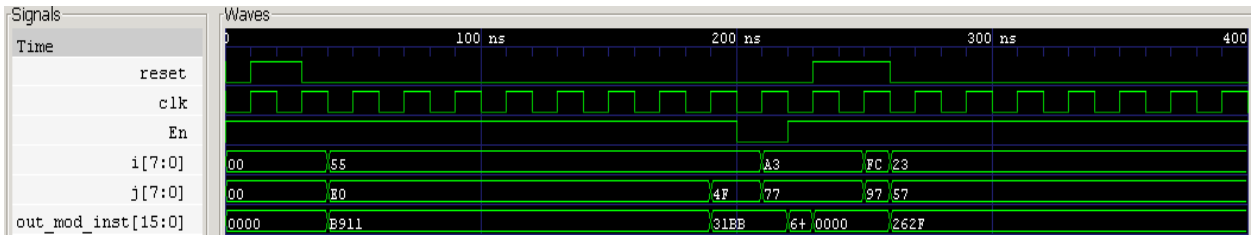

Рисунок 2. Временная диаграмма работы хэш-функции zip (на комбинаторной логике)

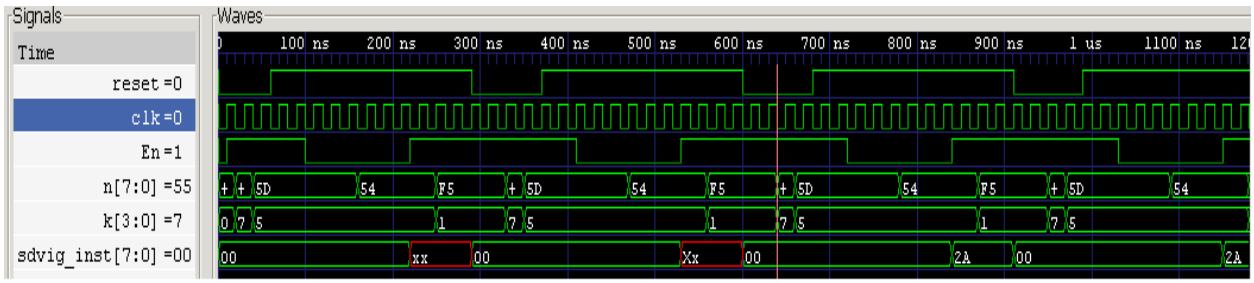

Рисунок 3. Временная диаграмма работы хэш-функции norm (на синхронной логике)

Временные диаграммы продемонстрировали правильность работы созданных схем. Таким образом, были реализованы (на языке описания аппаратуры Verilog) и протестированы основные хэш-функции, которые в перспективе будут использоваться в качестве базовых.

В дальнейшем будет продолжено расширение функциональности блока формирования хэш-функций для поддержки многозадачного режима работы, созданы аппаратные реализации новых хэш-функций. Всё это позволит увеличить производительность вычислительной системы и повысить эффективность загрузки вычислительных ресурсов ППВС «Буран».

Список литературы:

1. Климов А.В., Левченко Н.Н., Окунев А.С. Модель вычислений с управлением потоком данных как средство решения проблем больших распределенных систем // Материалы Второй Всероссийской научно-технической конференции «Суперкомпьютерные технологии» (СКТ-2012), 24—29 сентября 2012 года, с. Дивноморское, Геленджикский район. — С. 303—307.
2. Климов А.В., Левченко Н.Н., Окунев А.С. и др. Использование архитектуры потока данных для создания сверхвысокопроизводительных вычислительных систем // Материалы Второй Всероссийской научно-технической конференции «Суперкомпьютерные технологии» (СКТ-2012), 24—29 сентября 2012 года. — Дивноморское, Геленджикский район, 2012. — С. 64—68.
3. Левченко Н.Н., Окунев А.С. Об одном подходе к применению векторного функционального устройства в ППВС // Материалы Международной научно-технической конференции «Суперкомпьютерные технологии: разработка, программирование, применение» (СКТ-2010). — Таганрог — Москва, 2010. — Т. 1. — С. 124—126.

4. Стемпковский А.Л., Левченко Н.Н., Окунев А.С. Архитектура высокопроизводительной вычислительной системы с высокой реальной производительностью // Материалы Международной научно-технической конференции «Суперкомпьютерные технологии: разработка, программирование, применение» (СКТ-2010). — Таганрог—Москва, 2010. — Т. 1. — С. 153—157.
5. Стемпковский А.Л., Левченко Н.Н., Окунев А.С. Архитектура сверхпетафлопной вычислительной системы с высокой реальной производительностью, базирующейся на нетрадиционной модели вычислений // Материалы научной конференции «Результаты целевых ориентированных фундаментальных исследований и их использование в российской промышленности». — Таганрог, Изд-во ТТИ ЮФУ, 2010. — С. 68—72.
6. Яхонтов Д.Е., Левченко Н.Н., Окунев А.С. Принципы работы блока специальных операций модуля ассоциативной памяти параллельной потоковой вычислительной системы ППВС // Материалы Международной научно-технической конференции «Суперкомпьютерные технологии: разработка, программирование, применение» (СКТ-2010), Таганрог — Москва, 2010. — Т. 1. — С. 166—170.

References:

1. Klimov A.V., Levchenko N.N., Okunev A.S. Dataflow model of computations as a means of decisions the problems of large distributed systems. Superkomp'juternye tehnologii (SKT-2012). [Supercomputer technologies (SCT-2012)]. Divnomorscoe, 2012, pp. 303—307. (In Russian).
2. Klimov A.V., Levchenko N.N., Okunev A.S., Stempkovskiy A.L. Dataflow architecture usage for ultra performance computing systems development. Superkomp'juternye tehnologii (SKT-2012). [Supercomputer technologies (SCT-2012)]. Divnomorscoe, Russia, 2012, pp. 64—68. (In Russian).

3. Levchenko N.N., Okunev A.S. About one approach to vector functional unit application in PDCS. Superkomp'juternye tehnologii: razrabotka, programmirovanie, primenenie (SKT-2010). [Supercomputer technologies: design, programming, application (SCT-2010)]. Taganrog — Moscow, 2010, vol. 1, pp. 124—126. (In Russian).
4. Stempkovskiy A.L., Levchenko N.N., Okunev A.S. High-performance computer systems architecture with high real performance. Superkomp'juternye tehnologii: razrabotka, programmirovanie, primenenie (SKT-2010). [Supercomputer technologies: design, programming, application (SCT-2010)]. Taganrog — Moscow, 2010, vol. 1, pp. 153—157. (In Russian).
5. Stempkovskiy A.L., Levchenko N.N., Okunev A.S. Above petaflop’s computer systems architecture with high real performance, based on non-traditional computing model. Rezul'taty celevyh orientirovannyh fundamental'nyh issledovanij i ih ispol'zovanie v rossijskoj promyshlennosti [Results of the target-oriented fundamental researches and their usage in russian industry]. Taganrog, TTI IuFU Publ., 2010, pp. 68—72. (In Russian).
6. Yahontov D.E., Levchenko N.N., Okunev A.S. Principles of work of the special operations unit of the associative memory module of parallel dataflow computing system. Superkomp'juternye tehnologii: razrabotka, programmirovanie, primenenie (SKT-2010). [Supercomputer technologies: design, programming, application (SCT-2010)]. Taganrog — Moscow, 2010, vol. 1, pp. 166—170. (In Russian).