Колпаков А.А.
Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: kaf-eivt@yandex.ru

Структурная схема архитектуры гетерогенной системы «центральный процессор – графический процессор»

Как показано в известных работах, например в [1], увеличение производительности вычислительных систем напрямую зависит от организации работы вычислительных элементов (процессоров). В общем случае алгоритмы разбиваются на последовательные и параллельные ветви, каждая из которых может выполняться с использованием разного количества вычислителей. Каждая ветвь вычислений может описываться рядом параметров. При этом высокопроизводительные вычислительные системы могут функционировать как в мультипрограммном режиме, когда происходит выполнение одновременно множества задач в рамках одной вычислительной системы, так и в многопоточном режиме, когда в рамках одной программы запускается множество вычислительных потоков. Во втором случае часто встречается применение специализированных вычислительных модулей, имеющих высокую производительность в многопотоковом режиме. В таком случае центральный процессор выступает в роли управляющего блока. В качестве специализированных вычислительных модулей чаще всего выступают кластерные системы, цифровые сигнальные процессоры, а также в последнее время видеокарты или специализированные устройства на их основе.

В связи с этим возникает задача разработки методов увеличения производительности вычислительных систем с архитектурой, построенной с использованием центрального процессора в роли управляющего ядра и массива дополнительных вычислительных модулей в качестве основного вычислителя. В качестве массива дополнительных вычислительных модулей в данной работе рассматриваются однородные графические процессоры. Основной проблемой в описанной выше вычислительной системе представляется разработка модели памяти данной системы, что требует соответствующих исследований.

В общем случае гетерогенную вычислительную систему на основе специализированных вычислительных модулей можно представить в виде схемы, изображенной на рисунке 1. В составе специализированных вычислительных модулей можно выделить следующие ключевые элементы:

1. Мультипроцессоры – это основные вычислительные устройства, работающие по принципу SIMD.

2. Специализированная графическая оперативная память (GRAM), в которой следует выделить:

a. память констант, которая является кэшируемой, однако доступна из графического процессора только для чтения. Память констант наиболее подходит для хранения часто используемых данных, которые являются общими для всех нитей.

b. глобальную память, которая является очень емкой, однако, и наименее производительной. Глобальная память наилучшим образом подходит для хранения результатов вычислений.

Массив мультипроцессоров мультипроцессоров не имеет прямого доступа к оперативной памяти компьютера, поэтому данные, предназначенные для обработки на графическом процессоре, необходимо с применением специализированных команд и паттернов транслировать из оперативной памяти компьютера в GRAM в соответствии со структурной схемой на рисунке 1. При этом операции записи в память констант и в глобальную память производятся отдельно.

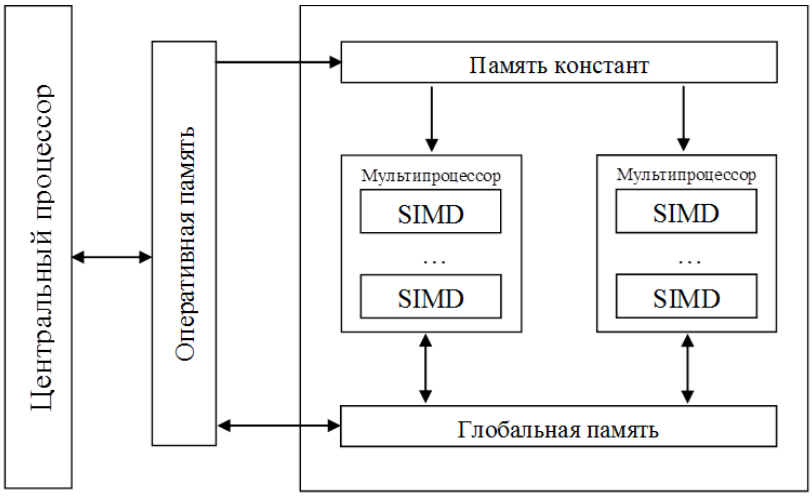

Рисунок 1 – Структурная схема архитектуры гетерогенной системы «Центральный процессор – Графический процессор»

Как можно увидеть из рис. 1, модель архитектуры «Центральный процессор – Графический процессор» представляет собой модель вычислительной системы с общей памятью, т.к. имеется объем памяти, который является доступным как для центрального процессора, так и для массива мультипроцессоров. Основными моделями с общей памятью являются:

- PRAM (англ. parallel random access memory) [2];
- BSP (англ. bulk synchronous parallel);
- LogP (англ. Latency, overhead, gap, Processors) [3];
- MapReduce [4,5].

Наиболее подходящей для описания архитектуры «Центральный процессор – Графический процессор» является абстрактная идеализированная модель PRAM (parallel random-access memory – память с параллельным произвольным доступом). Эта модель подразумевает, что весь массив памяти вычислительной системы доступен для чтения и записи для всех вычислителей в равной мере [1, с.14]. В данной работе в модели PRAM используются следующие допущения:

- объем устройств, выполняющих вычисления (q) является теоретически бесконечным;
- весь массив общей памяти вычислительной системы доступен для чтения и записи для всех вычислителей, размер общей памяти является теоретически бесконечным;
- отсутствует конкуренция по ресурсам;
- вычислители работают в режиме SIMD.

Работа вычислительных устройств синхронизирована, выполнение каждой инструкции занимает 1 такт.

В модели PRAM алгоритмы представляются в виде ациклического ориентированного графа «операции-операнды», по которому находятся значения основных характеристик, определенных в рамках данной модели [6].

Модель PRAM для всех сценариев является идеализированной, поэтому для описания реально существующих гетерогенных вычислительных систем ее необходимо дополнить. В работе [2] показано, что из всех имеющихся модификаций PRAM ни одна не подходит для оценки производительности гетерогенных вычислительных систем, поэтому базовую абстрактную модель PRAM необходимо модифицировать, чтобы привести ее в соответствие с гетерогенной архитектурой вычислительных систем на основе графических процессоров.

Как видно из схемы, приведенной на рис. 1, основными блоками вычислительного модуля на основе графического процессора являются мультипроцессоры. Мультипроцессор представляет собой массив скалярных вычислительных процессоров, а также разделяемую память. Разделяемая (shared) память – это сегмент оперативной памяти GPU, выделенный для определенного скалярного процессора, при этом данные в этом сегменте доступны только для этого скалярного процессора. Это реализовано для того, чтобы была возможность работы с промежуточными переменными, имеющими одинаковое обозначения, но являющимися

разными для разных нитей. Также внутри вычислительного модуля имеется некоторый массив глобальная оперативная память (SpRAM), который является общим для всех мультипроцессоров. SpRAM имеет объем, намного больший, чем разделяемая память мультипроцессоров, однако эта память является самой медленной из используемых в вычислительного модуля на основе графического процессора. Центральный процессор ведет обмен данными из оперативной памяти (RAM) с SpRAM. При этом обращение к разделяемой памяти осуществляется гораздо быстрее, поэтому целесообразно перед началом обработки переносить необходимые данные из SpRAM непосредственно в разделяемую память.

Исходя из этого, можно сделать вывод, что PRAM-модель подходит для описания гетерогенной вычислительной системы на основе графических процессоров только в случае ее модификации по правилам, приведенным ниже.

1. Применить для всех мультипроцессоров сценарий PRAM – CREW (одновременное чтение всеми вычислительными устройствами, операция записи осуществляется одновременно только одним вычислительным устройством).

2. Необходимо ввести значение q_{max}, которое будет обозначать максимальный объем скалярных процессоров в одном мультипроцессоре. Если требуется выполнение числа потоков, большего q_{max}, то происходит разбиение массива вычислительных нитей на пучки по q_{warp} скалярных процессоров. Пучки выполняются последовательно, тогда как внутри пучка выполнение потоков происходит параллельно.

3. Вводится значение M_s байт, которое обозначает максимальный объем разделяемой памяти, выделенной одному мультипроцессору [7,8].

4. Все скалярные процессоры в графическом процессоре являются однородными, поэтому следует ввести значение скорости выполнения элементарной операции скалярным процессором – S_{GPU} элементарных операций в секунду. Также подразумевается, что все скалярные процессоры работают по архитектуре SIMD.

5. Необходимо ввести операции чтения и записи данных в SpRAM вычислительного модуля на основе графического процессора со стороны центрального процессора. Время операций чтения и записи определяется значением K элементарных операций, которое требуется для обращения к одному 32-х разрядному числу в SpRAM.

Уточненная и дополненная новая модель PRAM представлена на рис. 2.

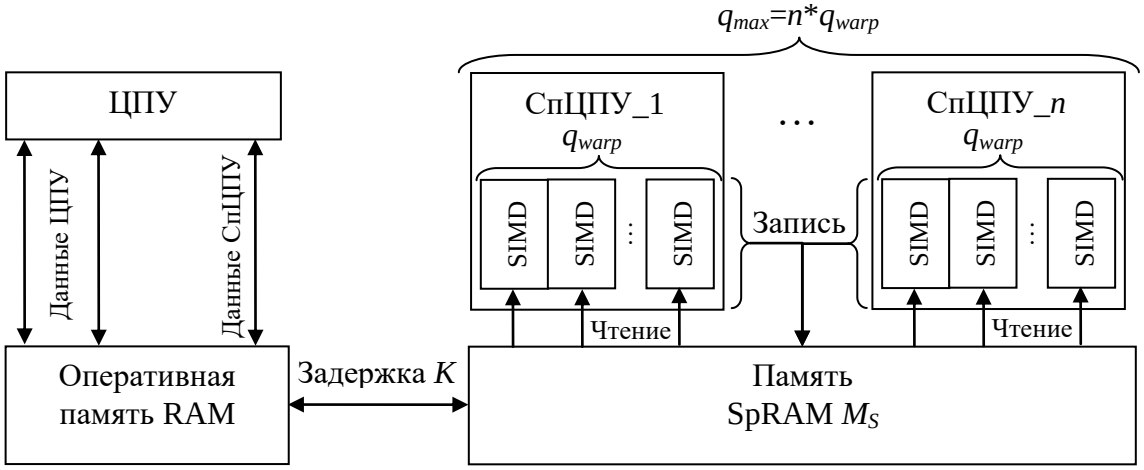

Рисунок 2 – Структурная схема модифицированной модели PRAM

Таким образом, применение перечисленных уточнений и дополнений позволяет использовать PRAM модель для описания гетерогенные вычислительные системы на основе графических процессоров. Однако, следует заметить, что приведенная выше модель все равно является идеализированной, т.к. в ней не рассмотрено применение специализированных типов памяти графического процессора, таких, как текстурная память и регистры.

В настоящий момент гетерогенные вычислительные системы на основе графических процессоров являются одним из самых быстро развивающихся сегментов высокопроизводительных вычислительных систем. Это связано с их относительной дешевизной, простотой освоения, компактностью и доступностью, что при этом сочетается с

достаточно высокой производительностью. При этом вышеописанные вычислительные системы имеют некоторые особенности, отличающие их от ранее существующих высокопроизводительных систем обработки данных, что накладывает определенные ограничения на применение уже существующих моделей параллельных систем для описания гетерогенные вычислительные системы на основе графических процессоров. В данной работе разработана структурная схема архитектуры гетерогенной вычислительной системы «Центральный процессор – Графический процессор», которая отражает особенности построения компьютерных вычислительных систем с использованием специализированных вычислительных модулей. На основе представленной архитектуры проведено уточнение и модификация модели с общей памятью PRAM, что позволяет применить ее для описания гетерогенных вычислительных систем на основе графических процессоров. В дальнейшем планируется использование разработанной модели в качестве основы для методики оценки производительности алгоритмов программного обеспечения для гетерогенные вычислительные системы на основе графических процессоров.

Литература

1. Баканов В.М. Параллельные вычисления: Учебное пособие. / В.М. Баканов. – М.: МГУПИ, 2006. – 124 с.

2. Капустин Д.С. Ржеуцкая С.Ю. Модификация абстрактной модели параллельных вычислений PRAM с учетом существенных особенностей графических процессоров / Д.С. Капустин, С.Ю. Ржеуцкая // Естественные и технические науки, №5(55). – М.: Спутник+, 2011. – С. 336-342.

3. Кропотов Ю.А., Кульков Я.Ю. Аппроксимация закона распределения вероятности амплитуд речевого сигнала / Ю.А. Кропотов, Я.Ю. Кульков // Радиотехника, 2006. – №11. – С.63-66.

4. Кропотов Ю.А., Проскуряков А.Ю., Белов А.А., Колпаков А.А. Методы проектирования телекоммуникационных информационно – управляющих систем аудиообмена в сложной помеховой обстановке / Ю.А. Кропотов, А.Ю. Проскуряков, А.А. Белов, А.А. Колпаков // Системы управления, связи и безопасности, 2015. – №2. – С.165-183.

5. Kropotov Y.A., Ermolaev V.A. Algorithms for processing acoustic signals in telecommunication systems by local parametric methods of analysis [Electronic resource]/ Y.A. Kropotov, V.A. Ermolaev // 2015 International Siberian Conference on Control and Communications (SIBCON) – Proceedings. – 2015. – Access mode: http://ieeexplore.ieee.org/document/7147109/

6. Кропотов Ю.А., Парамонов А.А. Методы проектирования алгоритмов обработки информации телекоммуникационных систем аудиообмена: моногр. / Ю.А. Кропотов, А.А. Парамонов. -М.-Берлин: Директ-Медиа, 2015. – 226 с.

7. Proskuryakov A. Y. Processing and forecasting of time series in systems with dynamic parameters [Electronic resource]/ A.Y. Proskuryakov // 2017 International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2017 – Proceedings. – 2017. DOI: 10.1109/ICIEAM.2017.8076366 – Access mode: https://ieeexplore.ieee.org/document/8076366/

8. Proskuryakov A. Intelligent System for Time Series Forecasting [Electronic resource]/ A. Proskuryakov // Procedia Computer Science, 2017. – Volume 103, Pages 363-369. DOI: 10.1016/j.procs.2017.01.122 – Access mode: https://www.sciencedirect.com/science/article/pii/S1877050917301230/