Кирюша Б.А., Горбик А.В.
УНК "Институт прикладного системного анализа" НТУУ "КПИ", Киев, Украина

Сравнительный анализ ведущих технологий вычислений общего назначения на графических процессорах

В последнее время возрастает интерес к гетерогенным вычислениям с использованием ранее не задействованных вычислительных мощностей, в частности, к вычислениям общего назначения на графических процессорах (**GPGPU** – **G**eneral **P**urpose computation on **G**raphical **P**rocessor **U**nit).

Основными особенностями использования графических процессоров являются: высокая степень параллелизма по данным, латентность обращения к внешней памяти и особенности архитектуры, оптимизированной для обработки больших массивов данных (*«crunch numbers»*).

GPGPU успешно применяется в моделировании сложных физических явлений, в трудоёмких процессах обработки изображений в медицине и криптографии (генерация ключей, подбор паролей).

Архитектура *GPU* на практике позволяет уменьшать время выполнения отдельных алгоритмов в 35-60 раз [1]. Такой уровень ускорения дает возможность говорить о потенциально высокой эффективности использования устройств и таких преимуществ, как:

- количество операций на единицу стоимости (Flops/dollar ratio) [2];
- количество операций на единицу потребляемой мощности (Flops/Watt ratio).

Высокое значение первого показателя привело к появлению *GPU* кластеров [3], специализированных графических процессоров *Fermi* и *Tesla* от *NVidia*, программных средств для поддержки работы на распределённых *GPU* системах (*C++ AMP*, *Virtual OpenCL*). Второй показатель позволяет частично решить проблему охлаждения мощных вычислительных узлов, а также ведет к внедрению технологий *GPGPU* на переносимые/мобильные устройства.

На ранних этапах развития *GPGPU* использовался *OpenGL*, где вычисления реализовывались в контексте логики стандарта. Разработчикам было необходимо разбираться в механизмах использования *API* без поддержки отладки и профилирования программ. С появлением библиотек, ориентированных на математические вычисления, ускорился процесс написания программ и эффективность использования устройств, появились реализации с возможностью отладки выполнения и профилирования.

Технология *CUDA C++* отличается широким набором реализованных примитивов, оптимизированными математическими библиотеками (*cuSparse, cuBlast, cuFFT*), специфическими расширениями и развитой архитектурой самих *GPU* (серии графических процессоров *Tesla* и *Fermi*). Архитектура *CUDA* используется для построения мощных кластеров (20-ая серия видеокарт *Tesla* позволяет достичь 500 гигафлопс для операций над числами с плавающей запятой двойной точности).

Главным конкурентом технологии от *NVIDIA* является открытый стандарт *OpenCL*, развитием которого занимается Khronos group. В основе *OpenCL* лежит полная поддержка гетерогенных вычислений – возможность эффективного использования *CPU* и *GPU*. Компания *AMD* выпускает реализацию *OpenCL* для своих архитектурных решений (ускоренные процессорные устройства - выполненные на одном кристалле центральный процессор и графический ускоритель), последние релизы *ATI Stream SDK* эффективно используют особенности *APU*, достигая скорости передачи данных между процессорами до 15 GB/s. В 2012 ожидается выпуск *OpenCL 2.0*.

На конференции AMD Fusion 11 компания Microsoft объявила выпуск нового открытого стандарта *C++ AMP (Accelerated Massive Parallelism)*, основными преимуществами использования которого являются простота использования (расширение стандарта C++, полная интеграция в Visual Studio 12) и заявленная поддержка облачных вычислений. Microsoft объявила о намерении поддерживать реализации стандарта другими компаниями (облачные сервисы Amazon и Salesforce будут поддерживать стандарт).

Таблица 1. Сравнение технологий вычислений общего назначения на графических процессорах

	CUDA C++ [4]	OpenCL [5]	C++ AMP [6]	OpenGL [7]
Доступность технологии	Проприетарная технология NVidia	Открытый стандарт, *Khronos group*	Открытый стандарт, *Microsoft*	Открытый стандарт, *Khronos group*
Наличие программных средств для разработки	SDK, NSight – debugger и profiler	Стандарт не подразумевает реализации (AMD предоставляет SDK, debugger)	Интеграция с Microsoft Visual Studio следующего релиза (debugger, profiler)	Стандарт не подразумевает конкретной реализации
Использование нескольких устройств	Да	Да. Использование GPU и CPU		Нет
Основные недостатки технологии перед конкурентами	Ориентирована на поставщика – закрытый стандарт	Реализация зависит от поставщика – необходимость дополнительной оптимизации	Новый стандарт, поддержка в тестовом режиме	Технология используется штучно – предназначена для обработки графики
Основные преимущества	Наиболее развитое API и множество расширений, специализированная архитектура процессоров	Открытый стандарт, поддержка гетерогенных вычислений (CPU + GPU)	Декларированная поддержка вычислений в облачных системах	Поддержка всеми основными поставщиками

Использование *GPU* в вычислениях общего назначения для многих алгоритмов дает значительный прирост скорости их выполнения, позволяя решать некоторые классы задач эффективней однородных многопроцессорных систем по ряду показателей. При выборе конкретной технологии *GPGPU* необходимо учитывать не только особенности программных моделей, а также эффективность использования самих устройств и возможность утилизации разных аппаратных платформ. В процессе проектирования решения задач необходимо принимать во внимание несколько важных факторов, в частности, переносимость программного кода, поддержку распределенных вычислений и использование возможностей узкоспециализированных платформ.

Литература. **1.** T. Preis, "GPU accelerated Monte Carlo simulation of the 2D and 3D Ising model," in *Journal of Computational Physics*, Volume 228, P. Virnau, W. Paul, J. Schneider, 2009, pp .4468–4477. **2.** Z.Fan et al, "GPU Cluster for High Performance Computing" in *ACM / IEEE Supercomputing* Conference, F. Qiu, A. Kaufman, S. Yoakum-Stover, 2004, Pittsburgh, PA. **3.** F. Chinchilla (2004 December) *Parallel N-Body Simulation using GPUs* [Online] Available: http://wwwx.cs.unc.edu/~tgamblin/gpgpu/GPGPfinalReport.pdf **4.** Official website [Online]. Available: http://developer.nvidia.com/cuda-tools-ecosystem **5.** Official website [Online]. Available: http://www.khronos.org/opencl/ **6.** MSDN website [Online]. Available: http://msdn.microsoft.com/en-us/library/hh265137%28v=vs.110%29.aspx **7.** Official website [Online]. Available: http://www.khronos.org/opengl