

УДК 004.312.46

НЕСПЕЦИАЛИЗИРОВАННЫЕ ВЫЧИСЛЕНИЯ НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ КАК АЛЬТЕРНАТИВА ВЫЧИСЛЕНИЯМ НА CPU

А.Б. Дедюрин, студент группы ИТб-121,
Научный руководитель: Алексеева Г.А., старший преподаватель

Кузбасский государственный технический университет
имени Т.Ф.Горбачева
г. Кемерово.

С возрастанием возможностей компьютеров человечество находит им применения в прикладной и научной сфере деятельности. Особенно

требовательны к аппаратным ресурсам такие задачи, как обработка и рендеринг потокового аудио-визуального контента высокого качества, моделирование физических процессов и явлений, обработка экспериментальных данных, воссоздание поведенческих особенностей и подражание мозговой деятельности. Обычному пользовательскому компьютеру с двух или четырехядерным CPU может понадобиться для таких расчетов несколько дней, а иногда недель или месяцев. Чаще всего нам необходимо оперативно получать результаты обработки (например, для обработки сейсмических данных или кодирования видео в режиме online). Для повышения производительности постоянно разрабатываются новые технологии и улучшаются характеристики оборудования. Одной из технологий является использование GPGPU.

GPGPU (General-purpose computing for graphics processing units) – техника использования графического процессора (GPU, англ. Graphics Processing Unit) видеокарты для выполнения не графических вычислений. Архитектура видеокарты позволяет быстро и без проблем “распараллелить” обработку данных, т.к. GPU обладает большим количеством ядер в отличие от CPU (англ. Central Processing Unit), что может затмить скорость вычислений даже самых мощных процессоров для многих параллельных задач обработки.

GPGPU используется для выполнения задач, которые раньше были прерогативой мощных процессоров, таких как шифрование/дешифрование, физические расчеты, научные вычисления, генерация криптовалют, таких как Bitcoin, финансовые расчеты, обработка сейсмических данных и тд. Важный момент при использовании GPU заключается в том, что он реализует принцип SIMD (single instruction, multiple data- одиночный поток команд, множе-

ственный поток данных), принцип компьютерных вычислений, позволяющий обеспечить параллелизм на уровне данных.

Реализации:

AMD FileStream- технология GPGPU, позволяющая реализовать алгоритмы, выполнимые на графических процессорах ускорителей ATI.

CUDA(англ. Compute Unified Device Architecture)- технология GPGPU, разработанная компанией NVIDIA, позволяющая реализовывать алгоритмы на языках C/C++/C# алгоритмы на графических процессорах ускорителей GeForce восьмого поколения и выше.

OpenCL- технология связанная с параллельными компьютерными вычислениями на различных типах графических и центральных процессоров.

OpenACC(Open Accelerators)- программный стандарт для параллельного программирования. Описывает набор директив компилятора, предназначенных для упрощения создания разнородных параллельных программ, задействующий как центральный так и графический процессоры.

C++ AMP(C++ Accelerated Massive Parallelism)- библиотека, использующая DirectX 11, и открытая спецификация, созданные Microsoft для реализации параллельных программ для гибридных систем на языке C++. Позволяет переносить вычисления на GPU, без внесения большого количества изменений в код программы.

DirectCompute от компании Microsoft представляет новый API для вычислений на GPU, работающий с текущей версией архитектуры параллельных вычислений CUDA от NVIDIA. Предназначен для выполнения вычислений общего назначения на графических процессорах.

Различия GPU и CPU:

CPU

Основная задача CPU, если говорить простыми словами, это выполнение цепочки инструкций за максимально короткое время. CPU спроектирован таким образом, чтобы выполнять несколько таких цепочек одновременно или разбивать один поток инструкций на несколько и, после выполнения их по отдельности, сливать их снова в одну, в правильном порядке. Каждая инструкция в потоке зависит от следующих за ней, и именно поэтому в CPU так мало исполнительных блоков, а весь упор делается на скорость выполнения и уменьшение простоев, что достигается при помощи кэш-памяти и конвейера.

GPU

Основная функция GPU — рендеринг 3D графики и визуальных эффектов, следовательно, в нем все немного проще: ему необходимо получить на входе полигоны, а после проведения над ними необходимых математических и логических операций, на выходе выдать координаты пикселей. По сути, работа GPU сводится к оперированию над огромным количеством независимых между собой задач, следовательно, он содержит большой объем памяти, но не

такой быстрой, как в CPU, и огромное количество исполнительных блоков: в современных GPU их 2048 и более, в то время как у CPU их количество может достигать 48, но чаще всего их количество лежит в диапазоне 2-8.

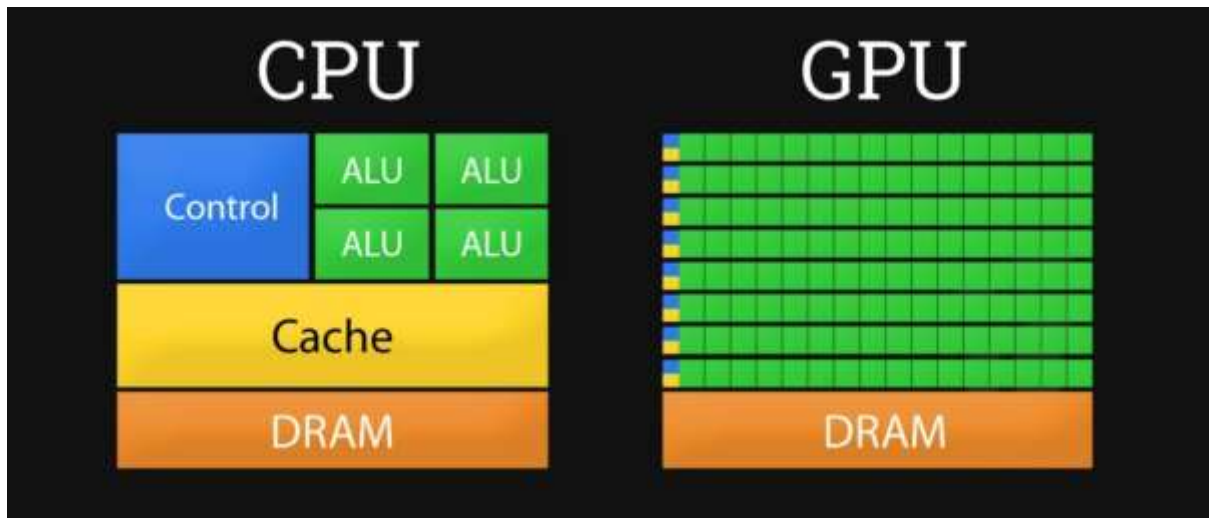


рис.1. Отличительная особенность GPU и CPU

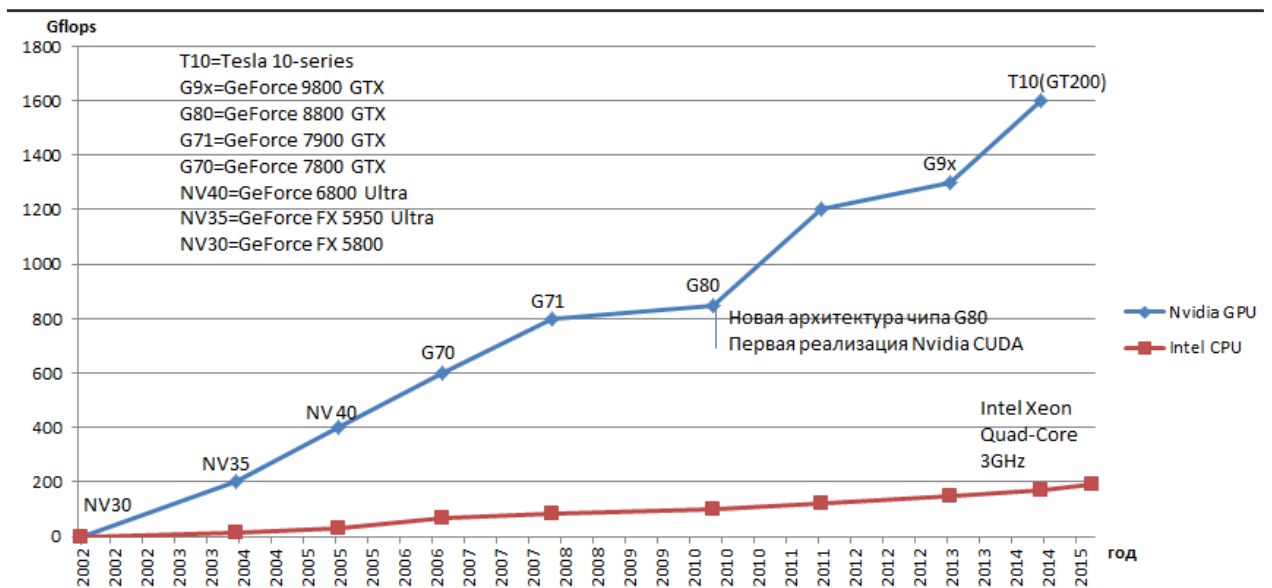


рис.2. Сравнение производительности.

Вывод: Выполнение расчётов на GPU показывает отличные результаты в алгоритмах, использующих параллельную обработку данных. То есть, когда одну и ту же последовательность математических операций применяют к большому объёму данных. При этом лучшие результаты достигаются, если отношение числа арифметических инструкций к числу обращений к памяти достаточно велико. Это предъявляет меньшие требования к управлению исполнением (flow control), а высокая плотность математики и большой объём данных отменяет необходимость в больших кэшах, как на CPU.

GPU оптимизирован для параллельной обработки данных, в отличие от последовательной обработки, выполняемой CPU. Графический процессор не

является заменой центральному, а скорее выполняет роль вычислительного блока и работает одновременно с ним. В настоящее время существуют разработки в которых оба процессора объединены в одно устройство, называемое APU.

Список использованной литературы:

1. Боресков А.В., Харламов А.А. и др. «Параллельные вычисления на GPU. Архитектура и программная модель CUDA.» Учеб. пособие / Предисл.: В. А. Садовничий. - М.: Издательство Московского университета, 2012. - 336 с.

2. Сайт Nvidia [Электронный ресурс] URL:

<http://www.nvidia.ru/object/cuda-parallel-computing-ru.html>

3. Сайт GROMACS [Электронный ресурс] URL:

http://www.gromacs.org/Documentation/Installation_Instructions_4.5/GROMACS-OpenMM

4. Хабрахабр [Электронный ресурс] URL:

<https://habrahabr.ru/>