

Параллельные вычисления с использованием OpenCL

науч. рук. Дегтярев А.

OpenCL (*Open Computing Language*) – открытый стандарт параллельного программирования. Он был разработан фирмой Apple и утвержден 9 декабря 2008 года организацией Khronos Group. OpenCL включает в себя собственный язык и интерфейс прикладного программирования. Этот стандарт создавался, прежде всего, как инструмент для параллельных вычислений, поэтому он одинаково эффективен для решения далеко не каждой задачи. Язык OpenCL имеет ряд встроенных математических функций, для которых можно задать точность. Такая гибкость расширяет область применения технологии, позволяя выбирать между скоростью работы и верностью результатов.

OpenCL – мультиплатформенный стандарт. Он совместим со всеми современными семействами операционных систем, такими как Windows, Linux и Mac. OpenCL также поддерживает как центральные так и графические процессоры. Все устройства связаны единым контекстом, а взаимодействие с ними осуществляется посредством командных очередей. В такой связи устройства выступают в роли серверов, а пользовательская программа в роли клиента.

Для программирования в OpenCL используется язык C, адаптированный под параллельные вычисления. Программа на таком языке состоит из одной глобальной функции, называемой ядром (kernel). Через специальный интерфейс ядро запускается на выполнение в цикле на выбранном устройстве. Итерации цикла выполняются параллельно. Таким образом использование OpenCL принесет выгоду только при эффективном разбиении исходной задачи на ряд независимых, решаемых одновременно.

Типовое решение	Решение с OpenCL
<pre>void scalar_mul(int n, const float* a, const float* b, float* result) { int i=0; for (int i=0; i<n; i++) result[i] = a[i]*b[i]; }</pre>	<pre>kernel void dp_mul(int n, global const float* a, global const float* b, global float* result) { int id=get_global_id(0); result[id] = a[id]*b[id]; }</pre>

Работа с графикой всегда была связана с решением большого количества математически емких задач, с которыми процессор не мог справиться в одиночку. Это была одна из причин появления дискретных графических плат. Большинство транзисторов на таких платах предназначены для обработки данных, а не для кэширования или управления потоком выполнения. Число ядер на видеокарте превосходит число ядер у центрального процессора на порядок, хотя они проигрывают по частоте в разы. Перечисленные преимущества позволяют видеокартам одновременно выполнять максимальное число итераций, получая значительный выигрыш в скорости.

Для тестирования технологии OpenCL была выбрана задача построения графиков функций, заданных неявно. Для таких функций не существует однозначного отношения координат, поэтому необходимо проверить существование графика в каждой точке некоторой заданной области размера NxN. Сложность такого алгоритма составляет N^2 (без учета сложности самой функции). Для реализации параллельного решения заданную область необходимо разбить на

несколько частей, обрабатываемых независимо. Теоретически, используя OpenCL и видеокарту, можно устремить сложность задачи к N.

Устройство	Количество ядер	Частота, Гц	Макс. потоков	Производительность
NVIDIA 8800GT	112	1,5	10752	≈2000
Intel E7400	2	2,8	2	1

Тестирование проводилось на следующем оборудовании, указанном в таблице. В качестве функции была выбрана формула $\sin(x \cdot \sin(x)) = \cos(y \cdot \cos(y))$. График строился на областях от 0,5 до 8 млрд точек. Результаты тестов показали, что видеокарта выполняет алгоритм почти в 200 раз быстрее процессора. Область из 8 млрд точек процессор проанализировал за 30 минут, а видеокарта всего за 8 секунд. Практический прирост производительности хоть и велик, но все же меньше теоретического на порядок. Это объясняется ветвлением алгоритма, необходимого для «сжатия» в памяти большого количества точек графика, и особенностью распределения параллельных задач на видеокарте.

OpenCL это новый стандарт, который требует реализации как аппаратно, так и программно. На данный момент список совместимых устройств невелик: 10 карт NVIDIA и 2 карты ATI Radeon. Программную поддержку получили практически все современные языки, такие как Java, C, C++, Ruby, Python, а также языки, связанные с платформой .NET.

У OpenCL есть свои соперники в лице CUDA и DirectCompute. Преимуществами перед ними можно считать открытость и поддержку рядом крупных компаний. OpenCL – гибкая технология, имеющая широкую область применения. Она подходит как для создания реалистичных спецэффектов, так и для решения сложных проблем, связанных с наукой.

Источники

1. OpenCL Parallel Computing for Heterogenous Devices, Khronos Group, 2009
2. The OpenCL Specification, Khronos OpenCL Working Group, 2009
3. khronos.org