

УДК 004

ПАРАЛЛЕЛЬНОЕ ВЫЧИСЛЕНИЕ С ПОМОЩЬЮ ГРАФИЧЕСКИХ ПРОЦЕССОРОВ

Асхатова Ляйсан Ильдаровна

канд. экон. наук

Казанский Федеральный университет, Казань

Галимов Эдвард Раифович

студент

Габдуллин Ильдар Масхутович

студент

Казанский национальный исследовательский технический
университет им. А.Н. Туполева, Казань

author@apriori-journal.ru

Аннотация. Рассматривается технология распараллеливания ресурсов между центральным и графическим процессорами.

Ключевые слова: процессор; видеокарта; сортировка.

PARALLEL COMPUTING ON THE GPU

Askhatova Laysan Ildarovna

candidate of economical sciences

Kazan Federal University, Kazan

Galimov Edward Raifovich

student

Gabdullin Ildar Maskhutovich

student

Kazan National Research Technical University, Kazan

Abstract. We consider the parallelization of resources between the CPU and GPU.

Key words: processor; video card; sorting.

С каждым годом человечество ставит перед собой все новые и новые задачи, как в прикладной, так и в научной сфере деятельности. В частности, сегодня это касается решения таких задач как обработка и рендер видеоданных в разрешение 4096 x 3112 (4K) пикселей, моделирование физических процессов и т.п.

Большинство как домашних, так и офисных компьютеров используются для рутинной работы редактирования текста, поиска в Интернете, просмотра видео, игр. Перед такими компьютерами не ставятся задачи, требующие сложных вычислений. Тем более, что таким компьютерам для выполнения сложных математических задач потребуется несколько дней, а может и недель. Для этих целей используются суперкомпьютеры или кластеры суперкомпьютеров, затраты на которые могут позволить не все компании.

В то же время передовые компании на рынке процессоров, такие как Intel, Nvidia, AMD, Samsung постоянно расширяют возможности своей продукции и предлагают нам новые технологии в производимых процессорах.

При выполнении программы, операционная система распределяет ресурсы между ядрами центрального процессора. Однако разработчики графических процессоров нашли другой способ распределения ресурсов. Решением стало технология распараллеливания ресурсов между центральным и графическим процессорами, которая позволила использовать ресурсы процессора видеокарты для решения неграфических задач.

Самыми распространенными представителями данной технологии являются NVidia со средством CUDA и AMD с ATI Stream Technology.

В данной статье, рассматривается технология NVidia Cuda на примере сортировки массива методом пузырька.

Алгоритм сортировки методом пузырька состоит из повторяющихся проходов по исходному массиву. За каждый проход элементы последовательно сравниваются попарно и, если порядок в паре неверный, выполняется обмен элементов. Проходы по массиву повторяются $N - 1$ раз

или до тех пор, пока на очередном проходе не окажется, что обмены больше не нужны, что означает – массив отсортирован. При каждом проходе алгоритма по внутреннему циклу, очередной наибольший элемент массива ставится на своё место в конце массива рядом с предыдущим «наибольшим элементом», а наименьший элемент перемещается на одну позицию к началу массива («всплывает» до нужной позиции как пузырёк в воде, отсюда и название алгоритма).

Теперь перейдем к сравнению решения задач только центральным процессором и технологией распараллеливания ресурсов. Для этого используем программу с 2 функциями (обычная сортировка и сортировка с помощью библиотеки NVidia Cuda) на трех компьютерах с разными видеокартами. Количество элементов массива после каждого выполнения каждой функции будет увеличиваться.

Возьмем видеокарту NVidia GTX 650 (Рис. 1) и NVidia GT 740M (Рис. 2).

Как видно из рисунков при различных видеокартах, результаты выполнения программы отличаются. Это связано с количеством ядер Cuda у графических адаптеров. У NVidia GTX 680 данный параметр равен 1536, у NVidia GTX 650 – 384, у NVidia GT 740M – 384.

Видеокарта NVidia GT 740M предназначена для ноутбуков, поэтому тактовая частота процессора меньше, в связи с этим результат выполнения программы отличается от видеокарты NVidia GTX 650 при одинаковом количестве ядер Cuda.

Реализация алгоритма метода пузырька с помощью центрального процессора и с технологией распараллеливания ресурсов показала, что время выполнения сокращается в несколько раз, но эффективна только для больших массивов. Это связано с временными затратами на пересылку данных из одной памяти в другую.

Преимущество данной технологии заключается в том, что даже на компьютерах с простыми видеокартами есть возможность реализации рассмотренной технологии.

N	Cuda	CPU
1	0.402000010	0.000000000
2	0.000000000	0.000000000
4	0.000000000	0.000000000
8	0.000000000	0.000000000
16	0.001000000	0.000000000
32	0.003000000	0.000000000
64	0.005000000	0.000000000
128	0.010000000	0.000000000
256	0.018999999	0.000000000
512	0.041000001	0.000000000
1024	0.077000000	0.001000000
2048	0.150999993	0.003000000
4096	0.273999989	0.012000000
8192	0.598999977	0.046000000
16384	1.281000018	0.180000007
32768	2.713999987	0.716000021
65536	6.230000019	2.864000082
131072	15.772999763	11.538999557
262144	45.355998993	46.085998535
524288	141.666000366	184.912994385
1048576	523.487976074	732.068969727

Рис. 1. Время выполнения сортировки методом пузырька при различных объемах массива (видеокарта NVidia GTX 650)

N	Cuda	CPU
1	0.856000006	0.000000000
2	0.000000000	0.000000000
4	0.004000000	0.000000000
8	0.000000000	0.000000000
16	0.004000000	0.000000000
32	0.008000000	0.000000000
64	0.012000000	0.000000000
128	0.024000000	0.000000000
256	0.052000001	0.000000000
512	0.104000002	0.004000000
1024	0.203999996	0.000000000
2048	0.412000000	0.008000000
4096	0.851999998	0.020000000
8192	1.761000037	0.075999998
16384	3.808000088	0.328000009
32768	8.654999733	1.307999969
65536	22.597999573	5.281000137
131072	66.027000427	21.204999924
262144	229.746994019	85.504997253
524288	799.041992188	341.123992920
1048576	2357.133056641	1013.562011719

Рис. 2. Время выполнения сортировки методом пузырька при различных объемах массива (видеокарта NVidia GT 740M)

К сожалению, графический процессор видеокарты, на сегодняшний момент, не может полностью заменить центральный, но зато с легкостью может его дополнять при реализации сложных неграфических алгоритмов.

Эффективное использование технологии распараллеливания ресурсов между графическим и центральным процессором может быть достигнуто только с мощными графическими адаптерами.