

## ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ ОБЩЕГО НАЗНАЧЕНИЯ

**Волков В.С.**

**Научный руководитель – к. ф.-м. н., доцент Карпова Е.Д.**

*Сибирский федеральный университет*

В последнее время большое внимание уделяется проблеме использования видеокарт для программирования широкого круга задач, не связанных непосредственно с графикой. В результате вместо термина «видеокарта» все чаще говорят о GPCPU.

GPGPU (General-Purpose Graphics Processing Units, графический процессор общего назначения) – техника использования графического процессора видеокарты для общих вычислений, которые обычно проводит центральный процессор.

CPU (Central Processing Unit, центральное обрабатывающее устройство) – электронный блок или микросхема, исполнитель машинных инструкций (кода программ), главная часть аппаратного обеспечения компьютера или программируемого логического контроллера. Иногда называют микропроцессором или просто процессором.

**OpenCL** (Open Computing Language, открытый язык вычислений) – фреймворк для написания компьютерных программ, на различных графических и центральных процессорах. OpenCL обеспечивает параллелизм на уровне инструкций и на уровне данных и является реализацией техники GPGPU. В фреймворк OpenCL входит язык программирования, который базируется на стандарте Си99.

Отметим следующие особенности OpenCL:

- использование всех вычислительных ресурсов системы;
- GPU, CPU и другие процессоры;
- параллелизм на уровне данных и на уровне задач;
- эффективная модель параллельных вычислений;
- абстрагирование от оборудования;
- основан на языке Си;
- Специфицирует точность вычислений;
- режимы округления IEEE754;
- определена точность встроенных функций;
- возможность добавления расширений.

На рис.1 представлена модель работы системы OpenCL.

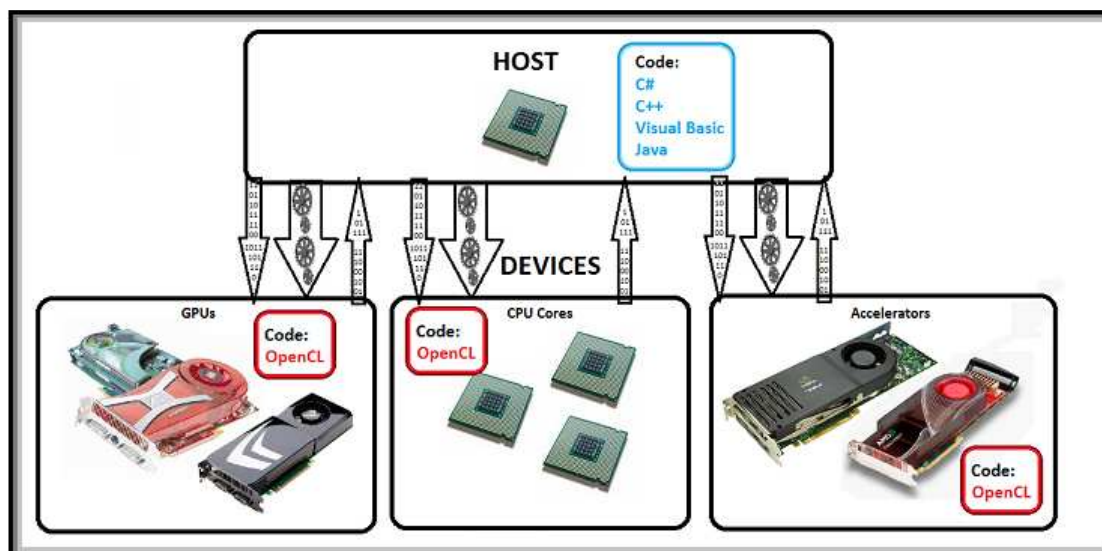


Рис.1. Модель работы системы OpenCL.

Цель данной работы использования GPGPU в сравнении с CPU на примере реализации алгоритма Хаффмана для сжатия данных.

В классической реализации алгоритма Хаффмана мы должны дважды прочитать файл: первый раз, чтобы посчитать вероятности вхождения каждого символа, а второй раз, выполняя сжатие файла. В работе рассмотрен алгоритм подсчета вхождения каждого символа в файле с использованием GPGPU и двух версий программ для CPU (последовательная и параллельная), написанных на языке программирования Си#.

Для тестирования использовалось следующее оборудование:

- процессор(CPU): AMD PHENOM X4 955 Black Edition 4-ядра;
- видеокарта: MSI GeForce GTS 450.

Для замера скорости работы программ, были выбраны два файла один размером 9,92 МБ (10 407 936 байт), второй 284 МБ (297 947 136 байт). Для измерения времени используются класс Stopwatch(), доступный в языке Си#. Точность замера времени зависит от частоты процессора и теоретически на используемом компьютере равна  $1/32902109016524975 = 3,0e-16$ .

Замеры производились по десять раз для каждой версии программы и для каждого файла. На рис. 2–3 представлены графики, демонстрирующие среднее время на основании замеров.

В табл. 1 и на рис. 2 представлена информация для файла размером 9,92 МБ. Из рисунка и таблицы видно, что на этом тесте OpenCL превосходит по скорости CPU(Sync) в шесть раз, а CPU(Async) в четыре раза.

В табл. 2 и на рис. 3 представлена информация для файла размером 284 МБ. Из рисунка и таблицы видно, что на этом тесте OpenCL превосходит по скорости CPU(Sync) в девяносто семь (!) раз, а CPU(Async) в сорок восемь раз.

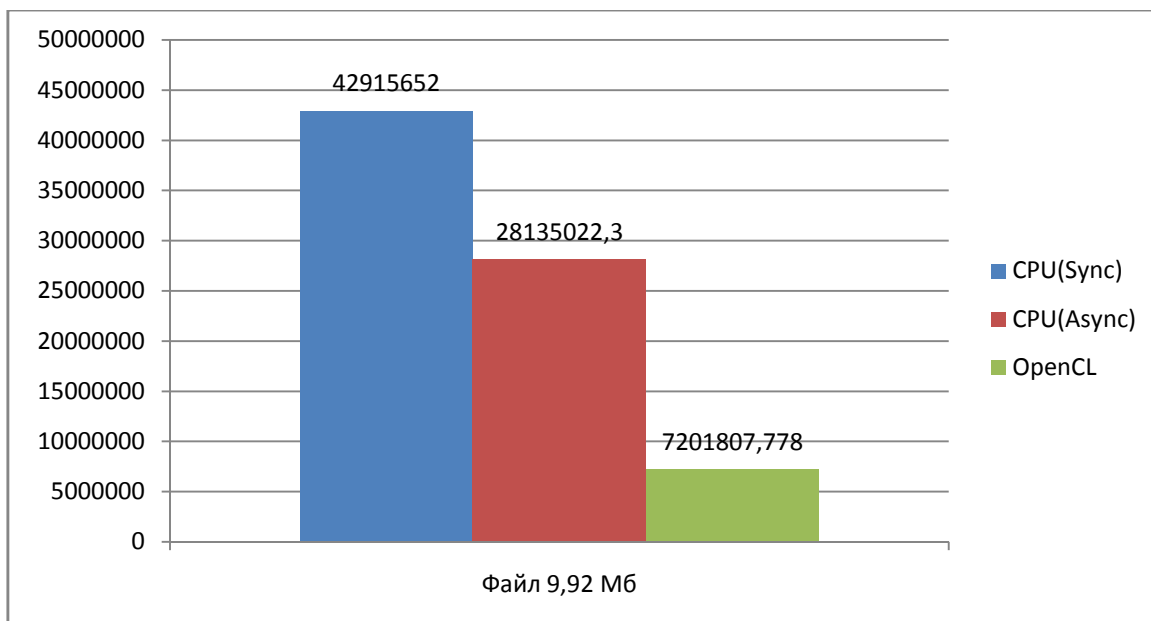


Рис.2. Среднее время выполнения программы на файле размером 9,92 МБ.

Таблица 1. Время выполнения программы на файле размером 9,92 МБ.

Синхронное время для CPU	Асинхронное время для CPU	Время OpenCL
00:00:04.2614014	00:00:02.6437079	00:00:00.7211382
00:00:04.2766471	00:00:03.1778067	00:00:00.7214607
00:00:04.2730737	00:00:02.5877035	00:00:00.7191261
00:00:04.5423760	00:00:02.5935329	00:00:00.7195999
00:00:04.2565692	00:00:03.2328503	00:00:00.7203116
00:00:04.2498611	00:00:02.6367112	00:00:00.7216355
00:00:04.2638007	00:00:02.6473304	00:00:00.7189992
00:00:04.2706906	00:00:02.7049616	00:00:00.7212668
00:00:04.2650615	00:00:03.2787918	00:00:00.7195983
00:00:04.2561707	00:00:02.6316260	00:00:00.7196289

Таблица 2. Время выполнения программы на файле размером 284 МБ.

Асинхронное время для CPU	Синхронное время для CPU	Время OpenCL
00:01:07.6638835	00:02:18.7945631	00:00:02.3442649
00:01:07.5787202	00:02:28.9771076	00:00:02.2658456
00:01:09.0680060	00:02:18.9736316	00:00:02.2664110
00:01:08.8866856	00:02:19.7737336	00:00:02.2724685
00:01:08.0746633	00:02:18.9537366	00:00:02.2694669
00:01:08.1586118	00:02:17.3537366	00:00:02.2695683
00:01:08.3605142	00:02:19.9245873	00:00:02.2647699
00:01:08.0265990	00:02:19.1421356	00:00:02.2718034
00:01:09.1984674	00:02:18.7498245	00:00:02.2649339
00:01:18.9827208	00:02:18.9423213	00:00:02.2668215

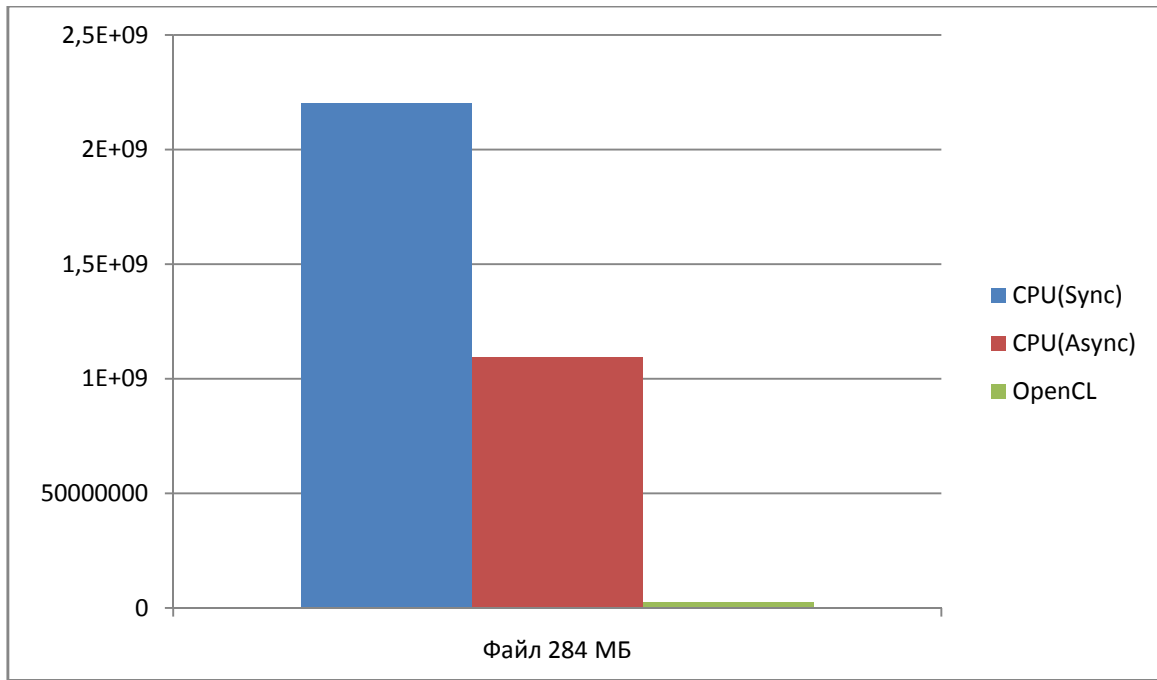


Рис.3. Среднее время выполнения программы на файле размером 284 МБ.

На основании проведенных численных экспериментов можно сделать вывод, что существует класс задача, для решения которых, архитектура GPGPU адаптирована наилучшим образом, особенно в сравнении с CPU.