Иванов Дмитрий Игоревич

гр. 328

1. Принципы фон Неймана в графических процессорах.

Графический процессор (англ. graphics processing unit, GPU) — программируемое вычислительное устройство, изначально предназначенное для обработки графической информации.

Работа с графикой предполагает большой объём обрабатываемой информации и параллелизм задачи, поэтому графический процессор изначально создавался как многоядерная структура. Современные GPU имеют довольно схожую архитектуру. Они содержат набор одинаковых вычислительных устройств (потоковых процессоров), работающих с общей памятью графического процессора (видеоОЗУ). Все потоковые процессоры синхронно исполняют одну и ту же команду, что позволяет отнести GPU к классу SIMD. Система команд потоковых процессоров включает арифметические команды для вещественных и целочисленных вычислений, команды управления (ветвления и циклы), а также команды обращения к памяти. Из-за высоких задержек команды доступа к оперативной памяти выполняются асинхронно. С целью сокрытия задержек в очереди выполнения GPU может одновременно находиться несколько сотен потоков, и если текущий поток блокируется по доступу к памяти, на исполнение ставится следующий.

Таким образом, о выполнении принципов фон Неймана в графических процессорах можно сказать следующее:

* Принцип последовательного выполнения нарушается из-за асинхронности работы с памятью и возможности переключения GPU между программными потоками.
* Остальные принципы фон Неймана очевидно выполняются.

2. Единицы измерения пропускной способности информационных каналов.

 Пропускная способность информационного канала — метрическая характеристика, показывающая соотношение предельного количества проходящих единиц информации в единицу времени через канал, то есть наибольшую возможную в данном канале скорость передачи информации. Пропускная способность канала есть скорость передачи информации при использовании «наилучших» (оптимальных) для данного канала источника, кодера и декодера, поэтому она характеризует только канал.

 Так как размерности пропускной способности информационного канала и скорости передачи информации совпадают, рассмотрим единицы измерения последней.

 Скорость передачи информации измеряется в битах в секунду, бодах и их производными.

 Бит в секунду, бит/с (англ. bits per second, bps) — базовая единица измерения скорости передачи информации, используемая на физическом уровне сетевой модели OSI или TCP/IP. На более высоких уровнях сетевых моделей, как правило, используется более крупная единица — байт в секунду (Б/c или Bps, от англ. bytes per second) равная 8 бит/c.

Производные образуются прибавлением приставок СИ: килобит в секунду (Кбит/с, Кbps, Kbit/s или Kb/s), мегабит в секунду (Мбит/с, Mbps, Mbit/s или Mb/s), гигабит в секунду (Гбит/с, Gbps, Gbit/s или Gb/s) и так далее. Аналогично для байта.

 Существует разница в значении производных единиц для измерения компьютерных и телекоммуникационных систем, обусловленная в различии смысла двоичных приставок. Для компьютерных систем 1 Кб = $2^{10}$ бит. Каждая следующая приставка увеличивает показатель степени на десять. Для телекоммуникационных систем 1 Кб = $10^{3}$ бит, а каждая следующая приставка увеличивает показатель степени на три.

 Бод (англ. baud) — единица измерения символьной скорости, количество изменений информационного параметра несущего периодического сигнала в секунду. То есть бод равен количеству информации, передаваемой за один периодический сигнал.

Например, в современных модемах используется квадратурная амплитудная манипуляция (КАМн), и одним изменением уровня сигнала может кодироваться несколько (до 16) бит информации.

Производные образуются при помощи приставок СИ.

Обычно в бодах измеряют полную пропускную способность канала (то есть включая служебные символы), а в битах в секунду — эффективную скорость канала (не включающую служебные символы).

3. Особенности вычисления на ЭВМ арифметических выражений.

 Представление чисел в ЭВМ дискретно и конечно. Как следствие, для хранения конкретного вещественного числа его приходится округлять. Это влечёт за собой погрешность вычислений. Для её уменьшения требуется производить арифметически эквивалентные преобразования над вычисляемым выражением. Рассмотрим некоторые из них.

* Если необходимо произвести сложение или вычитание длинной последовательности чисел, то начинать надо с наименьших чисел. Это обусловлено тем, что числа с плавающей точкой, хранимые в ЭВМ, распределены на числовой оси неравномерно: чем больше экспонента в нормированном представлении, тем они реже.
* При умножении нескольких чисел необходимо умножать большие числа на маленькие, так как умножение друг на друга нескольких маленьких чисел может привести к появлению машинного нуля.
* Необходимо сводить к минимуму число необходимых арифметических операций, чтобы не копилась погрешность при записи промежуточных результатов.
* В некоторых случаях имеет смысл выполнить предварительные преобразования вычисляемого выражения. Например, вместо $(a+b) ^{2}$для случая, когда одно число намного меньше другого (и при сложении результат будет равен большему числу), лучше использовать выражение: $a^{2}+2ab+b^{2}$ (произведение большого и маленького числа 2ab может превзойти погрешность).
* Если числа a и b почти равны, то вычитание этих чисел следует производить раньше, чем деление или умножение. К примеру, следует отдать предпочтение выражению $(a-b)/c$ вместо $a/c-b/c$. Во-первых в первом варианте меньше операций, а во-вторых, после деления числа станут меньше и еще меньше будут отличаться друг от друга, что может привести к обращению разности в машинный нуль.

4. Представление данных в современных вычислениях.

a) Представление текстовых данных.

Каждому символу ставится в соответствие неотрицательное число, называемое кодом символа. Конкретное соответствие между символами и их кодами называется системой кодировки. В современных ЭВМ, в зависимости от типа операционной системы и конкретных прикладных программ, используются 8-разрядные и 16-разрядные коды символов. Примеры кодировок: ASCII, КОИ-8, ISO, UNICODE.

b) Представление изображений.

Все известные форматы представления можно разделить на растровые и векторные. В векторном формате изображение разделяется на примитивы — прямые линии, многоугольники, окружности и сегменты окружностей, параметрические кривые, залитые определенным цветом или шаблоном, связные области, набранные определенным шрифтом отрывки текста и т. д. Каждый примитив описывается своими геометрическими координатами. Точность описания в разных форматах различна, нередко используются числа с плавающей точкой двойной точности или с фиксированной точкой и точностью до 16-го двоичного знака.

В растровом формате изображение разбивается на прямоугольную матрицу элементов, называемых пикселами. Матрица называется растром. Для каждого пиксела определяется его яркость и, если изображение цветное, цвет. Размер матрицы называется разрешением растрового изображения. Вторым параметром растрового изображения является разрядность одного пиксела, которую называют цветовой глубиной.

Наиболее широко используемые цветовые модели — это RGB (Red, Green, Blue — красный, зеленый, синий), CMY (Cyan, Magenta, Yellow — голубой, пурпурный, желтый, дополнительные к RGB) и CMYG — те же цвета, но с добавлением градаций серого.

Цветовая модель RGB используется в цветных кинескопах и видеоадаптерах, CMYG — в цветной полиграфии.

В различных графических форматах используется разный способ хранения пикселов. Два основных подхода — хранить числа, соответствующие пикселам, одно за другим, или разбивать изображение на битовые плоскости — сначала хранятся младшие биты всех пикселов, потом — вторые и так далее. Обычно растровое изображение снабжается заголовком, в котором указано его разрешение, глубина пиксела и, нередко, используемая цветовая модель.

c) Представление звуковой информации.

Метод FM (Frequency Modulation) основан та том, что теоретически любой сложный звук можно разложить на последовательность простейших гармонических сигналов разных частот, каждый из которых представляет собой правильную синусоиду, а, следовательно, может быть описан числовыми параметрами, то есть кодом. Данный метод обеспечивает компактный код.

Метод таблично волнового (Wave-Table) синтеза соответствует современному уровню развития техники. В заранее подготовленных таблицах хранятся образцы звуков для множества различных музыкальных инструментов. Числовые коды выражают тип инструмента, номер его модели, высоту тона, продолжительность и интенсивность звука, динамику его изменения, некоторые параметры среды, в которой происходит звучание, а также прочие параметры, характеризующие особенности звучания. Поскольку в качестве образцов исполняются реальные звуки, то его качество получается очень высоким и приближается к качеству звучания реальных музыкальных инструментов.

d) Представление видео.

Видео — это сочетание звуковой и графической информации. Первый кадр хранится целиком (ключевой), а в следующих хранятся лишь отличия от начального кадра (разностные кадры). В видео существует много ситуаций, связанных со сменой действия, когда первый кадр новой сцены настолько отличается от предыдущего, что его проще сделать ключевым, чем разностным. Регулярное расположение подобных кадров в потоке обеспечивает оперативный доступ к любому фрагменту видео. Кроме того, указанная мера позволяет эффективно восстановить изображение при сбоях или при пропуске отдельных кадров на медленных ЭВМ.