Nghiên cứu và tìm hiểu kiến trúc GPU Nvidia

# Tổng quan về điện toán GPU (GPU computing)

## GPU là gì?

GPU viết tắt cho Graphics Processing Unit được gọi là Đơn vị xử lý đồ họa ( đôi khi được gọi VPU – Visual Processing Unit) là một bộ vi xử lý chuyên dụng nhận nhiệm vụ tăng tốc, xử lý đồ họa cho bộ vi xử lý trung tâm CPU.

Các GPU hiện đại có năng suất rất cao trong xử lý đồ họa máy tính. Với cấu trúc mang tính xử lý song song mạnh mẽ của mình, GPU cho thấy nó hiệu quả hơn CPU rất nhiều trong nhiều thuật toán phức tạp.

GPU được sử dụng trong các hệ thống nhúng, điện thoại di động, máy tính cá nhân, máy trạm, máy chơi game v…v.. Trong máy tính cá nhân, một GPU có thể xuất hiện ở card đồ họa, hoặc nó cũng có thể được gắn trên mainboard.

## Lịch sử phát triển của điện toán GPU (GPU computing)

GPU được đưa ra và giới thiệu lần đầu tiên vào 31-8-1999 bởi NVIDIA và được xem như bộ xử lý song song phổ biến nhất ngày nay. Dựa trên những mong muốn chưa được thỏa mãn về khả năng đồ họa như thật, thời gian thực, GPU đã phát triển trở thành bộ xử lý có thể biểu diễn các phép tính dấu chấm động (floating-point) cũng như khả năng lập trình. Những GPU ngày nay đã thật sự vượt qua các CPU trong khả năng tính toán số học và băng thông bộ nhớ, khiến chúng trở thành bộ xử lý lý tưởng để tăng tốc đa dạng các ứng dụng xử lý dữ liệu song song.

Các nỗ lực để khai thác GPU cho các ứng dụng không mang tính đồ họa đã được tiến hành từ năm 2003. Đặc biệt trong khoa học máy tính, cùng với các nhà nghiên cứu trong các lĩnh vực như hình ảnh y khoa và điện từ, họ đã bắt đầu sử dụng GPU để chạy các ứng dụng điện toán đa dụng (general purpose). Họ đã nhận thấy sự xuất sắc trong khả năng biểu diễn các phép tính dấu chấm động của các GPU, dẫn đến tăng hiệu suất rất lớn cho 1 loạt ứng dụng khoa học. Từ đây dẫn đến sự ra đời của GPGPU – General Purpose computing trên các GPU.

Tuy mô hình GPGPU đã chứng minh khả năng tăng tốc tuyệt vời của mình, nhưng nó vẫn phải đối mặt với một số vấn đề. Thứ nhất, nó đòi hỏi các lập trình viên phải có kiến thức sâu sắc về các giao diện lập trình ứng dụng đồ họa (graphics APIs) và kiến trúc của GPU. Thứ hai, các vấn đề phải được thể hiện trong các thuật ngữ của các chương trình xác định tọa độ đỉnh (**vertex coordinate)**,kết cấu ( **textures)** và đổ bóng (**shader),** qua đó tăng sự phức tạp lên rất nhiều. Thứ ba, các tính năng lập trình cơ bản như truy xuất đọc, viết ngẫu nhiên vào bộ nhớ đã không được hỗ trợ, giới hạn mô hình lập trình rất nhiều.

Nhưng bên cạnh đó cũng nảy sinh một số vấn đề mà GPU phải đối mặt. Thứ 1 đó là GPGPU đòi hỏi sử dụng các ngôn ngữ lập trình đồ họa như OpenGL, Cg, DirectX và nhiều thuật toán xử lý dữ liệu song song khác nhau để lập trình GPU. Điều này đã giới hạn khả năng tiếp cận đến hiệu suất to lớn của GPU dành cho khoa học vì nó đòi hỏi các lập trình viên phải có kiến thức sâu sắc về các giao diện lập trình ứng dụng đồ họa (graphics APIs) và kiến trúc của GPU.

NVIDIA realized the potential to bring this performance to the larger scientific community and decided to invest in modifying the GPU to make it fully programmable for scientific applications and added support for high-level languages like C, C++, and Fortran. This led to the **CUDA architecture** for the GPU.

## Những lĩnh vực ứng dụng của GPU

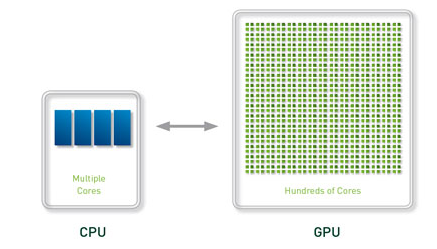
# Kiến trúc GPU

## Kiến trúc G80

## Kiến trúc GT200

## Kiến trúc Fermi

# GPU và CPU



# Các thuật toán song song áp dụng trên GPU

Chúng em sẽ nghiên cứu thuật toán **Parallel Prefix Sum (Parallel Scan)** và demo một bài toán trên GPU có áp dụng thuật toán này.

Hướng nghiên cứu của chúng em như vậy có ok không thầy? Mong thầy cho chúng em thêm ý kiến để chúng em có thể tiếp tục đề tài của mình. Cám ơn thầy.