Architectures des processeurs

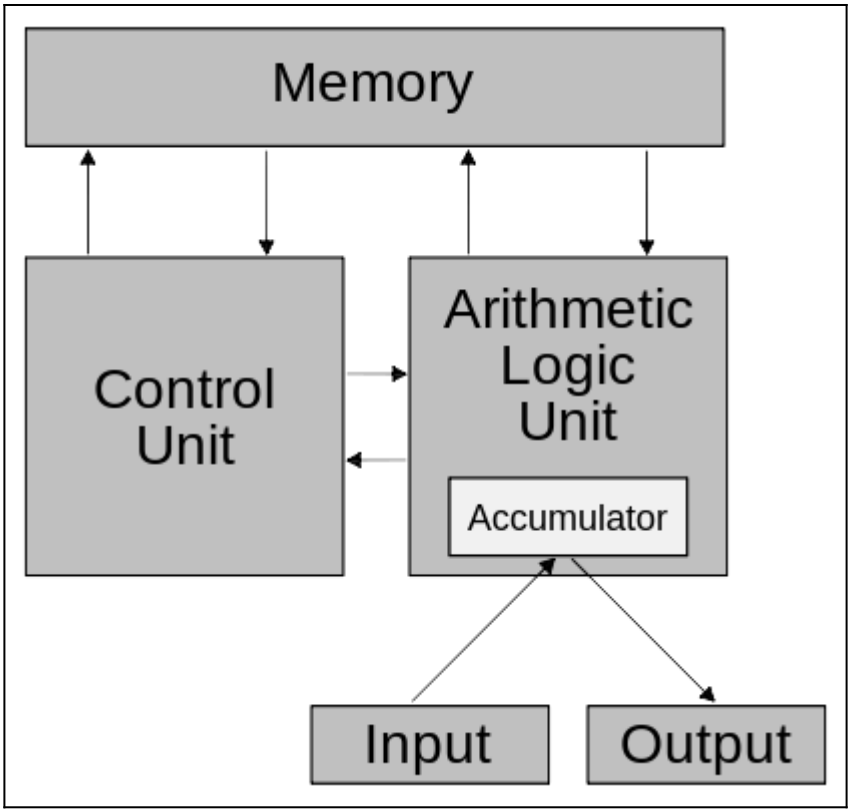
L'architecture matérielle des GPU est très différente des architectures matérielles que vous avez peut-être pu rencontrer par le passé. Pour bien appréhender leur utilisation et le portage d'applications sur ces plates-formes, il est nécessaire de bien comprendre ces différences et quels vont être les impacts sur le parallélisme réalisable sur ces architectures.

Comparaison entre l'architecture des CPU et des GPU

Si vous vous référez au titre des sections sur l'architecture des CPU et des GPU, vous pourriez penser qu'elles sont très proches. En fait, le schéma global est dû à l'origine de toutes ces puces qui est l'architecture dite de von Neumann. Dans ces deux cas, les puces sont constituées d'unités de calcul qui sont pilotées par un séquenceur et alimentées par des bus mémoire. Mais en dehors de ces points communs globaux, les possibilités sont très différentes.

Architecture des CPU

Schéma global



Architecture CPU

L'architecture des CPUs actuels est issue tout droit des travaux de von Neumann . Aucune des autres achitectures (par exemple l'architecture de Harvard) n'a eu autant de succès que celle-ci. Il y a naturellement eu des variations depuis la mise en place de cette architecture qui seront exposées dans les parties associées.

Séquenceur

Le séquenceur est le chef d'orchestre d'un processeur. Il est chargé de lire les instructions, de les décoder et de les exécuter. Pour cela, c'est lui

qui interagira avec la mémoire et qui pilotera les unités de calcul.

Les séquenceurs modernes possèdent des systèmes complexes comme les pipelines, permettant d'exécuter progressivement les instructions, des ordonnanceurs d'instructions, permettant de changer l'ordre d'exécution d'instructions élémentaires, et, de plus en plus de registres intermédiaires, permettant de changer le thread en cours à peu de frais (on change les registres utilisés pour d'autres sans avoir à stocker leur contenu en mémoire).

Toutes ces techniques font que les CPUs actuels sont tout-terrain et peuvent effectuer tout type de tâche efficacement.

Unités de calcul

Il existe deux principaux types d'unités de calcul : les calculs entiers et les flottants. Si les premiers sont à peu près stables depuis le début des CPU, les seconds ont vu une évolution totalement différente.

Les calculs entiers sont à la base d'un processeur. Ces unités vont permettre de calculer des adresses mémoire que ce soient pour les données ou pour les instructions, et naturellement exécuter des calculs scientifiques entiers.

Les calculs flottants ont longtemps été émulés par les unités entières. Intel introduisit des unités spécifiques dans les coprocesseurs 387 et 487 puis sur les 386 DX et 486 DX. Le fonctionnement était très particulier car il fallait transmettre les données sur le coprocesseur, exécuter les instructions puis récupérer les données. Ce mode de fonctionnement est resté en place jusqu'à l'arrivée des unités vectorielles MMX puis SSE. Actuellement, on peut utiliser les unités de calculs flottants sans avoir à transmettre les données ou les placer dans des registres dédiés et on peut surtout effectuer plusieurs calculs identiques sur des données différentes en même temps (instructions vectorielles).

Avec l'arrivée des Pentium 4, Intel propose maintenant des processeurs avec Hyperthread, ce qui signifie que les certaines unités de calcul

peuvent être utilisées par un autre thread lors quelles ne sont pas utilisées. AMD a un système différent où les unités de calculs flottants sont partagés par différentes cœurs. D'autres constructeurs permettent d'utiliser plusieurs threads par cœur, partageant les unités de calculs entre ces threads. L'objectif de tous ces systèmes est de permettre une utilisation de toutes les unités de calcul, afin de maximiser le rapport calculs/watt.

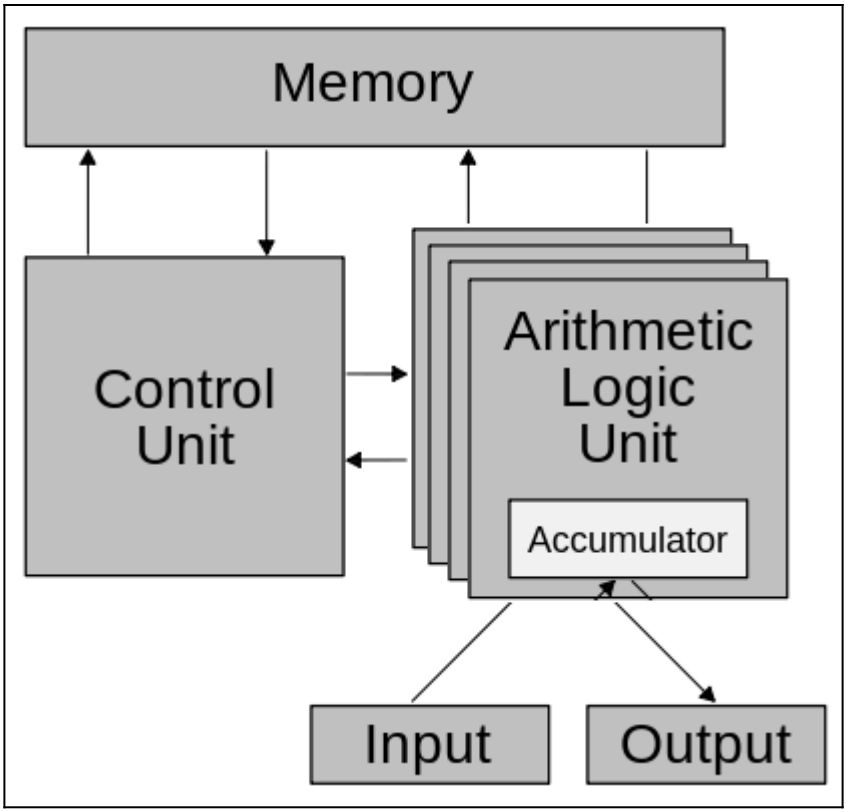
Accès mémoire

Les accès mémoire d'un CPU ont engendré une hiérarchie complexe. Les processeurs ayant gagné plus rapidement en vitesse que les systèmes d'accès à la donnée, il a été nécessaire d'ajouter des étapes intermédiaires permettant de stocker les données les plus utilisées.

Le premier système utilisé dans les processeurs est les registres. Les CPU x86 ont peu de registres, contrairement à d'autres architectures comme les ARM, ils doivent donc souvent accéder à des données dans un système plus lent. Le système le plus lent est les disques (disque dur, DVD...) et c'est aussi celui qui peut contenir le plus de données. Une fois les données chargées depuis les disques, elles sont stockées dans la mémoire principale, à l'extérieur du processeur. Celui-ci va accéder aux informations dans cette mémoire et la stocker dans des “caches” successifs. Actuellement, il existe 3 niveaux de cache, les niveaux les plus rapides (L1 et L2) étant généralement dédiés pour un cœur, le L3 étant partagé au niveau du processeur.

Architecture des GPU

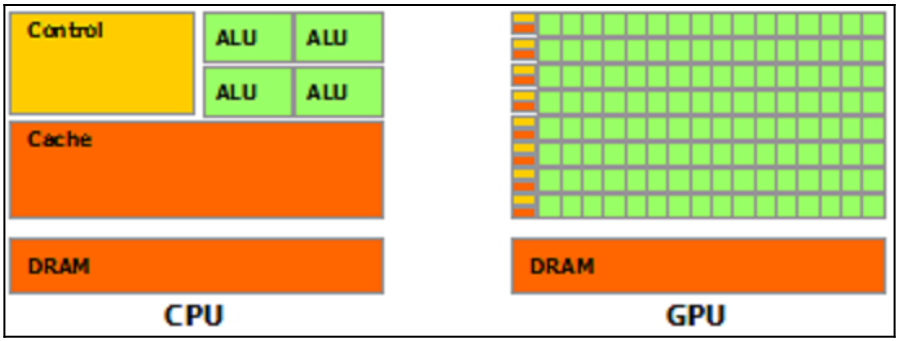
Schéma. Non détaillé selon les évolutions des GPU



Architecture GPU

Programmation hétérogène

Comparaison CPU et GPU (low latency processor)



Comparaison CPU et GPU

	Cpu (low latency)	**Gpu (high troughput)**
Control	Oui (prediction branch, data forward)	Non
Alu	Peu nombreux	Nombreux, pipeline
Cache	Large	Small
Utilisation	Séquentiel	parallèle