Architecture des CPU

Revenons déjà sur le principe de fonctionnement et la différence entre processeur CISC et processeur RISC.

► CISC

Les processeurs x86 sont des CPU CISC (Complex Instruction Set Computing). Le CISC est une architecture de processeur qui utilise des instructions complexes et spécialisées pour effectuer des opérations spécifiques.

Voici quelques caractéristiques clés des processeurs CISC :

1. **Instructions complexes** : Les processeurs CISC utilisent des instructions qui peuvent effectuer plusieurs opérations en une seule étape, telles que des opérations arithmétiques, des opérations de déplacement de bits, des opérations de comparaison, etc.
2. **Instructions spécialisées** : Les processeurs CISC ont des instructions spécialisées pour des opérations spécifiques, telles que des instructions pour la gestion des chaînes de caractères, des instructions pour la gestion des tableaux, etc.
3. **Microcode** : Les processeurs CISC utilisent souvent un microcode pour implémenter les instructions complexes. Le microcode est un ensemble de instructions plus simples qui sont exécutées par le processeur pour effectuer les instructions complexes.
4. **Pipeline** : Les processeurs CISC ont souvent un pipeline plus complexe que les processeurs RISC (Reduced Instruction Set Computing), ce qui signifie que les instructions sont exécutées en plusieurs étapes et que les données sont traitées en parallèle.

Les avantages des processeurs CISC sont :

- **Meilleure utilisation des ressources** : Les processeurs CISC peuvent effectuer des opérations complexes en une seule étape, ce qui peut réduire le nombre d'instructions nécessaires pour effectuer une tâche donnée.
- **Meilleure performance** : Les processeurs CISC peuvent offrir de meilleures performances que les processeurs RISC pour certaines applications, en particulier celles qui nécessitent des opérations complexes et spécialisées.

Cependant, les processeurs CISC ont également des inconvénients, tels que :

- **Complexité** : Les processeurs CISC sont souvent plus complexes que les processeurs RISC, ce qui peut rendre leur conception et leur implémentation plus difficiles.
- **Consommation d'énergie** : Les processeurs CISC peuvent consommer plus d'énergie que les processeurs RISC, en particulier pour les opérations complexes.

► RISC

Les processeurs RISC (Reduced Instruction Set Computing) sont une architecture de processeur qui utilise des instructions simples et générales pour effectuer des opérations.

Voici quelques caractéristiques clés des processeurs RISC :

1. **Instructions simples** : Les processeurs RISC utilisent des instructions simples et générales qui peuvent être combinées pour effectuer des opérations complexes.
2. **Pas d'instructions spécialisées** : Les processeurs RISC n'ont pas d'instructions spécialisées pour des opérations spécifiques, mais plutôt des instructions générales qui peuvent être utilisées pour différentes opérations.
3. **Pas de microcode** : Les processeurs RISC n'utilisent pas de microcode pour implémenter les instructions, mais plutôt des circuits intégrés pour exécuter les instructions.
4. **Pipeline simple** : Les processeurs RISC ont souvent un pipeline simple, ce qui signifie que les instructions sont exécutées en quelques étapes et que les données sont traitées de manière séquentielle.

Les avantages des processeurs RISC sont :

- **Simplicité** : Les processeurs RISC sont souvent plus simples que les processeurs CISC, ce qui peut rendre leur conception et leur implémentation plus faciles.
- **Consommation d'énergie réduite** : Les processeurs RISC peuvent consommer moins d'énergie que les processeurs CISC, en particulier pour les opérations simples.
- **Meilleure scalabilité** : Les processeurs RISC peuvent être plus faciles à mettre à l'échelle pour les applications parallèles et les systèmes multiprocesseurs.

Cependant, les processeurs RISC ont également des inconvénients, tels que :

- **Moins de performances** : Les processeurs RISC peuvent offrir des performances inférieures à celles des processeurs CISC pour certaines applications, en particulier celles qui nécessitent des opérations complexes et spécialisées.
- **Plus d'instructions nécessaires** : Les processeurs RISC peuvent nécessiter plus d'instructions pour effectuer une tâche donnée que les processeurs CISC, ce qui peut augmenter le temps d'exécution.

En résumé, les processeurs RISC sont conçus pour offrir une simplicité, une consommation d'énergie réduite et une meilleure scalabilité, mais ils peuvent être moins performants que les processeurs CISC pour certaines applications.

Exemples de processeurs RISC :

- ARM (Advanced RISC Machines)
- PowerPC (Performance Optimization With Enhanced RISC)
- SPARC (Scalable Processor Architecture)
- MIPS (MIPS Instruction Set)

Il est important de noter que les processeurs modernes sont souvent des hybrides qui combinent des éléments de CISC et de RISC, et que les différences entre les deux architectures sont devenues moins nettes au fil des ans.

Pour exemple ou retrouvait les processeurs RISC dans les stations de calcul ou de travail dans le domaine de l'industrie.
Les processeur PowerPC par exemple se retrouvaient dans les stations de travail de CAO (Conception Assistée par Ordinateur) avec les première version des logiciels de conception 3D comme Catia (logiciel de Dassault) qui était utilisé pour la conception des avions, des automobiles … etc.
Les fameuses station Silicon Graphics qu'on pouvait trouver dans les années 1990 – 2000.



Silicon Graphics Octane 2 - 2003

Apple a également intégré des processeur PowerPC dans ses Mac avant de quitter cette architecture pour venir un temps sur le x86 avec Intel et ensuite de revenir sur le RISC avec ses propres processeurs basés sur ARM.

On a pu trouver aussi ces processeurs au sein des consoles de jeu comme sur la Xbox 360, sur la Playstation 2.

Aujourd’hui les seuls processeurs RISC qui subsistent sont les ARM qu’on retrouve dans l’informatique embarquée, dans les smartphones et les tablettes.

► ARM

L'architecture ARM (Advanced RISC Machines) est une architecture de processeur RISC (Reduced Instruction Set Computing) développée par ARM Holdings. Elle est conçue pour être utilisée dans une large gamme d'applications, allant des systèmes embarqués aux serveurs et aux ordinateurs personnels.

Voici quelques caractéristiques clés de l'architecture ARM :

1. **RISC** : L'architecture ARM est basée sur la philosophie RISC, qui consiste à utiliser des instructions simples et générales pour effectuer des opérations.
2. **32 bits ou 64 bits** : L'architecture ARM peut être utilisée pour des processeurs 32 bits ou 64 bits, selon les besoins de l'application.
3. **Load/Store** : L'architecture ARM utilise un modèle de chargement et de stockage (Load/Store) pour accéder à la mémoire, ce qui signifie que les données sont chargées dans les registres du processeur avant d'être traitées.
4. **Registres** : L'architecture ARM utilise un ensemble de registres pour stocker les données et les adresses. Les registres sont divisés en deux catégories : les registres généraux et les registres spéciaux.
5. **Instruction Set** : L'architecture ARM utilise un ensemble d'instructions pour effectuer des opérations telles que les opérations arithmétiques, les opérations de déplacement de bits, les opérations de comparaison, etc.
6. **Pipeline** : L'architecture ARM utilise un pipeline pour exécuter les instructions, ce qui signifie que les instructions sont exécutées en plusieurs étapes et que les données sont traitées de manière séquentielle.
7. **Cache** : L'architecture ARM utilise un cache pour stocker les données et les instructions fréquemment utilisées, ce qui peut améliorer les performances du processeur.

Les avantages de l'architecture ARM sont :

- **Faible consommation d'énergie** : L'architecture ARM est conçue pour être économe en énergie, ce qui la rend idéale pour les applications mobiles et les systèmes embarqués.
- **Haute performance** : L'architecture ARM offre de hautes performances pour les applications qui nécessitent des opérations complexes.
- **Flexibilité** : L'architecture ARM peut être utilisée pour une large gamme d'applications, allant des systèmes embarqués aux serveurs et aux ordinateurs personnels.

Les inconvénients de l'architecture ARM sont :

- **Compatibilité** : L'architecture ARM peut avoir des problèmes de compatibilité avec les logiciels et les systèmes d'exploitation conçus pour les architectures x86.
- **Support** : L'architecture ARM peut nécessiter un support spécifique pour les logiciels et les systèmes d'exploitation.

Les processeurs ARM sont utilisés dans une large gamme d'applications, notamment :

- **Téléphones mobiles** : Les processeurs ARM sont utilisés dans la plupart des téléphones mobiles pour leur faible consommation d'énergie et leur haute performance.
- **Tablettes** : Les processeurs ARM sont utilisés dans les tablettes pour leur faible consommation d'énergie et leur haute performance.
- **Systèmes embarqués** : Les processeurs ARM sont utilisés dans les systèmes embarqués pour leur faible consommation d'énergie et leur haute performance.
- **Serveurs** : Les processeurs ARM sont utilisés dans les serveurs pour leur faible consommation d'énergie et leur haute performance.

En résumé, l'architecture ARM est une architecture de processeur RISC conçue pour être utilisée dans une large gamme d'applications, allant des systèmes embarqués aux serveurs et aux ordinateurs personnels. Elle offre de hautes performances, une faible consommation d'énergie et une grande flexibilité, mais peut avoir des problèmes de compatibilité et de support.

► Les jeux d’instruction

- **x86** : Le jeu d'instructions x86 est utilisé par les processeurs Intel et AMD. Il est connu pour sa complexité et sa grande variété d'instructions.
- **ARM** : Le jeu d'instructions ARM est utilisé par les processeurs ARM Holdings. Il est connu pour sa simplicité et sa faible consommation d'énergie.
- **MIPS** : Le jeu d'instructions MIPS est utilisé par les processeurs MIPS Technologies. Il est connu pour sa simplicité et sa grande flexibilité.
- **PowerPC** : Le jeu d'instructions PowerPC est utilisé par les processeurs IBM et Motorola. Il est connu pour sa grande performance et sa flexibilité.
- **SPARC** : Le jeu d'instructions SPARC est utilisé par les processeurs Oracle. Il est connu pour sa grande performance et sa sécurité.

Ces jeux d'instructions sont utilisés pour écrire des programmes qui peuvent être exécutés par les processeurs. Chaque jeu d'instructions a ses propres caractéristiques et avantages, et le choix du jeu d'instructions dépend de l'application et des besoins du système.

Voici quelques exemples d'instructions courantes :

- **ADD** : Additionne deux nombres
- **SUB** : Soustrait deux nombres
- **MUL** : Multiplie deux nombres
- **DIV** : Divise deux nombres
- **LOAD** : Charge une valeur en mémoire
- **STORE** : Stocke une valeur en mémoire
- **JUMP** : Saute à une adresse spécifique
- **CALL** : Appelle une fonction

Ces instructions sont utilisées pour écrire des programmes qui peuvent être exécutés par les processeurs. Les jeux d'instructions sont conçus pour être efficaces et flexibles, et pour permettre aux programmeurs de créer des programmes qui peuvent être exécutés sur une large gamme de systèmes.