

## Architectures de processeurs et sécurité matérielle

### TD APSM23-2 – Architecture superscalaire "Out-of-Order"

On s'intéresse dans ce TD à des microarchitectures superscalaires simplifiées, fonctionnant (sauf si précisé autrement) en mode "Out-of-Order". Les processeurs considérés sont basés sur un modèle d'exécution chargement-rangement, exécutant des instructions de calcul arithmétique ou logique de type  $Rx \text{ op } Ry \rightarrow Rz$ , où  $Rx$ ,  $Ry$  et  $Rz$  sont trois registres banalisés appartenant à un banc de registres. Tous les registres architecturaux et microarchitecturaux de ces processeurs sont sensibles au même front d'horloge.

#### Partie 1 – Exécution de calculs

On considère dans un premier temps la microarchitecture illustrée en Figure 1. Les caractéristiques principales sont :

- Deux instructions mémorisées en étage de lecture, avec lecture de 2 instructions par cycle au maximum (selon l'existence d'aléas)
- Accès mémoire en cache L1 en 1 cycle d'horloge
- Registres architecturaux (banc de registres) sur 32 bits. Le banc de registres permet de réaliser quatre lectures et deux écritures par cycle ; un registre donné peut être connecté à plusieurs sorties au même cycle (lectures multiples possibles)
- 3 unités de traitement : U1 et U2 pour les calculs et une unité pour les accès en mémoire données (LOAD et STORE, incluant les calculs d'adresse en fonction du mode d'adressage)
- U1 permet de réaliser les opérations  $A + B$  et  $A - B$
- U2 permet de réaliser les opérations  $A + B$  et  $A + 1$
- Pipelines sur 4 étages pour chacune des trois unités : (1) lecture avec mise à jour du compteur ordinal, (2) décodage et allocation des unités de traitement avec transfert des opérandes, (3) exécution dans chacune des unités et (4) rangement des résultats sous contrôle de l'unité de complétion
- Aucun mécanisme de gestion des aléas au niveau des pipelines des unités de traitement
- Pas d'élément de mémorisation dans l'unité de complétion

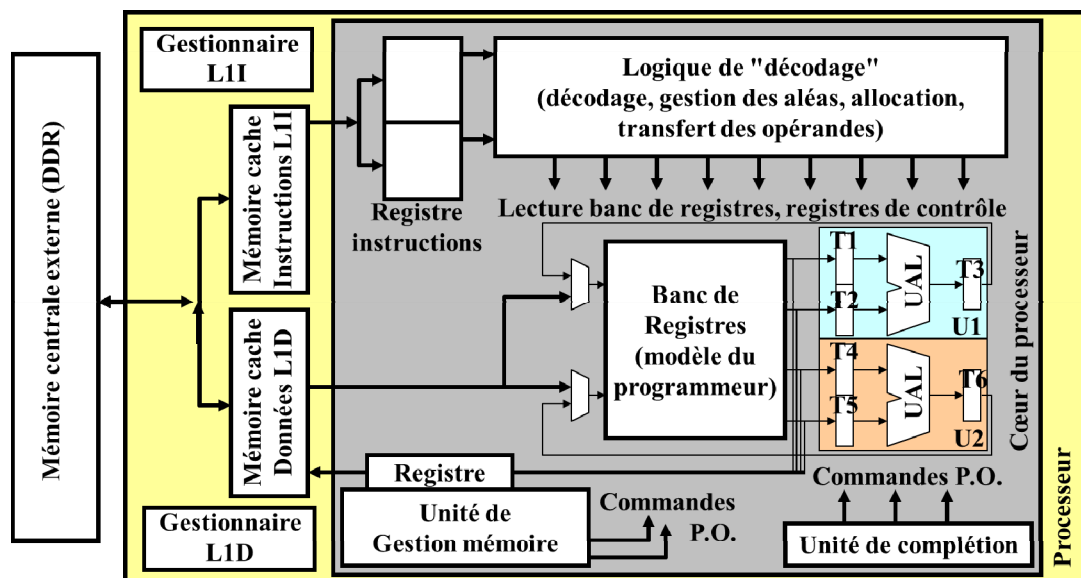


Figure 1 – Microarchitecture simplifiée d'un processeur superscalaire à 2 voies avec une hiérarchie mémoire à 2 niveaux (les registres de contrôle des pipelines et la plupart des interconnexions internes ne sont pas représentés)

- 1.1. Dans le cas idéal où aucun aléa d'exécution ne survient, combien faut-il de cycles d'horloge pour exécuter six instructions avec l'architecture proposée ? Combien en faut-il pour exécuter cinquante mille instructions ?

- 1.2. On souhaite exécuter le programme suivant, obtenu par compilation à partir d'un langage de haut niveau et décrit ici en "pseudo-assembleur" :

Instruction I1 :  $R2 - R23 \rightarrow R13$   
Instruction I2 :  $R23 + R5 \rightarrow R8$   
Instruction I3 :  $R23 + 1 \rightarrow R24$   
Instruction I4 :  $R12 + 1 \rightarrow R12$   
Instruction I5 :  $R25 + R12 \rightarrow R3$   
Instruction I6 :  $R6 + R7 \rightarrow R23$

Combien de cycles d'horloge vont être nécessaires pour l'exécution de ce programme (depuis le chargement de I1, jusqu'au rangement du résultat de I6), en supposant que les instructions sont présentes en mémoire cache ? On précisera les étapes de chaque instruction terminées à chaque cycle, et les unités de calcul utilisées.

- 1.3. On suppose pouvoir intervenir sur l'ordre des instructions dans la mémoire programme. Sans modifier la fonctionnalité, quel ré-ordonnement des instructions du programme de la question 1.2 pourrait être envisagé pour améliorer les performances ?
- 1.4. D'une façon plus générale, indépendamment de la microarchitecture considérée, quels types de dépendances vont conduire à limiter la réorganisation dynamique des instructions dans l'extrait de programme suivant ?

Instruction I1 :  $R4 + R5 \rightarrow R3$   
Instruction I2 :  $R3 + R21 \rightarrow R6$   
Instruction I3 :  $R2 + R7 \rightarrow R4$   
Instruction I4 :  $R2 + R3 \rightarrow R30$   
Instruction I5 :  $R5 + R1 \rightarrow R30$

## **Partie 2 – Microarchitecture du bloc de gestion mémoire et évitement de bulles**

- 2.1. Proposez une organisation/structure du pipeline de l'unité de gestion mémoire (avec interconnexions aux autres éléments du processeur), en se limitant au mode d'adressage à base registre, afin que les instructions "LOAD Rx, \$D", et "STORE Rx, \$D" puissent être terminées en 4 cycles, sans induire de bulle dans le processeur (en plus des bulles éventuelles liées à la lecture simultanée de deux instructions de ce type ou des dépendances de données).

Rappel du TD APSM23-1 : En mode d'adressage à base registre, l'adresse d'accès \$Ad à la mémoire donnée correspond à la somme du contenu d'un registre architectural prédéfini (par exemple, R31) et du déplacement \$D fourni en valeur immédiate dans le code de l'instruction.

- 2.2. Quelle modification de structure serait possible pour réduire au maximum le nombre de bulles dans le cas des deux extraits de programme suivant ? Quel est le nombre de bulles minimum (on se placera dans le pire cas) ?

Extrait 1

Instruction I1 :  $R4 + R5 \rightarrow R3$   
Instruction I2 :  $R3 + R21 \rightarrow R6$

Extrait 2

Instruction I1 : "LOAD R3, \$D"  
Instruction I2 :  $R3 + R21 \rightarrow R6$

## **Partie 3 – Comparaison "In-Order" et "Out-of-Order"**

On considère maintenant la microarchitecture illustrée en Figure 2. Les caractéristiques principales sont :

- Deux instructions mémorisées en étage de lecture, avec lecture de 2 instructions par cycle au maximum (selon l'existence d'aléas)
- Accès mémoire en cache L1 en 1 cycle d'horloge
- Registres architecturaux (banc de registres) sur 64 bits, utilisés pour stocker aussi bien des nombres entiers (registres R0 à R31) que des nombres flottants (registres RF0 à RF31). Le banc de registres permet de réaliser quatre lectures et deux écritures par cycle ; un registre donné peut être connecté à plusieurs sorties au même cycle (lectures multiples possibles)
- 3 unités de traitement : IU et FPU pour les calculs entiers et flottants, et une unité LSU pour les accès en mémoire données (LOAD et STORE, incluant les calculs d'adresse en fonction du mode d'adressage)

- IU dispose d'un pipeline interne sur 2 étages (exécution IE et rangement IR)
- FPU dispose d'un pipeline interne sur 4 étages (3 étages d'exécution FE1, FE2, FE3 avant rangement FR)
- LSU dispose d'un pipeline interne sur 2 étages comme dans le cas des parties 1 et 2 du TD (exécution LE et rangement LR)
- Des mécanismes matériels gèrent toutes les dépendances de données, dans le domaine du possible (cf. question 2.2)
- Pas d'élément de mémorisation dans l'unité de complétion

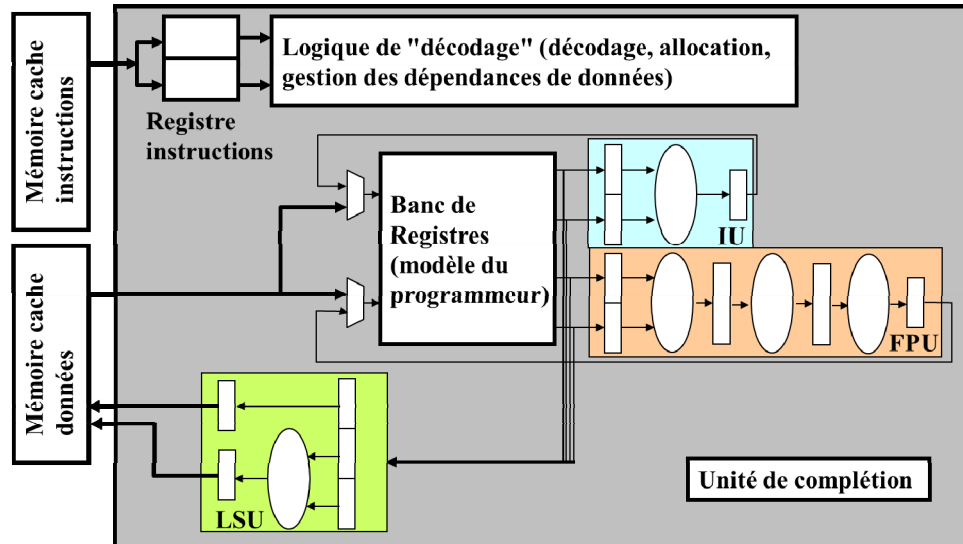


Figure 2 – Autre microarchitecture simplifiée d'un processeur superscalaire à 2 voies

- 3.1. On souhaite exécuter le programme suivant, en supposant que les données sont disponibles en cache L1 ou dans les registres internes :

Instruction I1 : LOAD R3, \$10  
 Instruction I2 : LOAD R8, \$11  
 Instruction I3 : R3 + R8 -> R3  
 Instruction I4 : RF2 + RF4 -> RF6  
 Instruction I5 : R7 + R9 -> R10  
 Instruction I6 : R11 + R12 -> R13

Quel est le déroulement temporel de cet extrait de programme dans le cas d'un processeur "In-Order" ? L'unité de complétion a-t-elle une utilité si le processeur n'est pas "Out-of-Order" ?

- 3.2. Quel est le déroulement temporel de cet extrait de programme dans le cas d'un processeur "Out-of-Order" ?

#### **Partie 4 – Impact des paramètres de l'unité de complétion**

- 4.1. Si l'unité de complétion est dotée d'une mémoire permettant le stockage intermédiaire des résultats à ranger, et peut utiliser un port d'écriture spécifique sur le banc de registres (en plus des ports d'écriture représentés sur la Figure 2), quelle est la différence de déroulement par rapport à la question 3.2. ?
- 4.2. Si l'unité de complétion est dotée d'une mémoire permettant le stockage intermédiaire des résultats à ranger, mais doit utiliser l'un des deux ports existant du banc de registres, quelle est la conséquence ?

#### **Partie 5 – Exemple d'évaluation rapide de la puissance de calcul : MIPS crête**

- 5.1. Un processeur superscalaire dispose de 4 unités de traitement, chacune composée d'un pipeline entre 2 et 8 étages. L'unité de lecture d'instructions peut stocker jusqu'à 6 instructions. L'unité de complétion dispose d'une mémoire FIFO par unité de traitement, et de 4 accès parallèles en écriture sur le banc de registres. La fréquence de l'horloge est 1 GHz. Quelle est l'influence de chaque paramètre sur les performances crête du processeur ? Quel est le nombre de MIPS crête ? Quelle est la durée minimum d'exécution d'un programme nécessitant d'exécuter 6000 instructions ?
- 5.2. Quelle hypothèse supplémentaire doit être explicitée pour garantir la réponse précédente ?