

GIF-17455 – Architecture des microprocesseurs

Examen Final

Jeudi le 18 décembre 2003

Durée: 13h30-15h20

Notes : Pour toutes les questions, la réponse seule n'obtient aucun point. Selon les problèmes, **vous devez montrer vos calculs et/ou justifier vos réponses.**

Les points attribués à chaque question sont donnés à titre indicatif et peuvent changer légèrement lors de la correction.

Question 1 (25 points)

Un système a les caractéristiques suivantes:

- 95% des accès à la mémoire sont en cache
- 10% des accès à la mémoire sont des écritures
- la cache a des blocs de deux mots
- un bloc est alloué en cache lors d'un accès manqué (en lecture ou écriture)

Faites de plus l'hypothèse qu'un accès en écriture est l'écriture d'un seul mot.

- a) Pour 100 accès à la mémoire, calculez le nombre de transferts de **mots** fait entre la cache et la mémoire (dans les deux directions)
 - pour le cas d'une cache de *type ré-écriture (write-back)* Faites l'hypothèse que tous les blocs sont propres.
 - pour le cas *d'une cache à écriture simultanée (write-through)*.
- b) Sans faire le calcul, quel(s) élément(s) de votre réponse en a) changerait si certains blocs étaient « sales » pour la cache avec ré-écriture ?
- c) Si 25% des instructions sont des lectures de données, quel est le pourcentage des instructions qui sont des écritures de données ?
- d) Soit une cache à deux niveaux, i.e., L1 et L2, avec inclusion de L1 dans L2. Quel est l'avantage d'avoir une cache avec écriture simultanée (write-through) pour L1 ? Quel est l'avantage d'avoir une cache avec ré-écriture (write-back) pour L2 ?

Question 2 (15 points)

Un processeur a des adresses virtuelles de 35 bits et des adresses physiques de 32 bits. Les adresses virtuelles sont traduites en adresses physiques avant de faire accès à la cache.

Les pages ont 64 KO (2^{16}), et 19 bits sont utilisés comme « tag » lors de l'accès à la cache TLB.

Huit (8) bits sont utilisés pour indexer la cache et 19 bits pour vérifier le « tag » des entrées de la cache. Que pouvez-vous dire de :

- a) le type de cache utilisé pour la TLB
- b) la dimension de la TLB (en KO)
- c) la grandeur des blocs de la cache
- d) la dimension de la cache (en KO)

Question 3 (15 points)

Nous avons un système de deux processeurs avec mémoire centralisée. Chaque processeur a sa cache. Les caches utilisent la politique de ré-écriture (write-back) et les caches font de l'allocation lors d'un accès manqué. Toutes les entrées des caches sont initialement invalides.

Le programme exécuté sur le système génère les actions suivantes :

Pour $i = 1$ à K :

```
{  
    Processeur P1 écrit à l'adresse A1  
    Processeur P2 lit l'adresse A1  
    Processeur P2 écrit à l'adresse A1  
    Processeur P1 lit l'adresse A1  
}
```

Pour un protocole d'invalidation comme nous avons vu en classe, combien y aura-t-il d'accès manqués dans les caches ? Donnez la réponse en fonction de K .

Question 4 (20 points)

Pour ce problème, le processeur a un pipeline de 5 étages, IF/ID/EX/MEM/WB, émission simple, exécution dans l'ordre et passage des données, évaluation des conditions de branchement au cycle ID, et plusieurs unités d'exécution avec les délais suivants :

Opération arithmétique logique sur des entiers : 1 cycle en EX

Opération arithmétique sur nombres en point flottant : 8 cycles en EX

Calcul d'adresse : 1 cycle en EX

Faites l'hypothèse que seulement les obstacles de données peuvent causer des arrêts du pipeline. Il n'y a donc pas de conflits sur les ressources, les branchements ne causent aucun arrêt du pipeline, et tous les accès mémoires sont en cache et prennent 1 cycle.

Soit la boucle suivante :

```
A :  L.D. F0, 0(R1)
      L.D. F2, 0(R2)
      MUL.D F4, F2, F0
      S.D. F4, 0(R1)
      DADDIU R1,R1,-16
      DADDIU R2,R2,-16
      BNEZ R1, A
```

Lors de la compilation, cette boucle est exécutée avec la technique du pipeline logiciel. Négligeant le code d'initialisation et de terminaison, la nouvelle boucle est:

```
A :  S.D. F4, 32(R1)
      MUL.D F4, F2, F0
      L.D. F2, 0(R2)
      L.D. F0, 0(R1)
      DADDIU R1,R1,-16
      DADDIU R2,R2,-16
      BNEZ R1, A
```

- Identifiez pourquoi cette boucle ne peut s'exécuter sans arrêt du pipeline (lorsque le branchement est pris).
- Déroulez la boucle le nombre **minimum** de fois pour pouvoir l'implanter en pipeline logiciel sans arrêt du pipeline. Donnez le nouveau code du pipeline logiciel (sans le code d'initialisation et de terminaison).

Question 5 (25 points)

Ces questions font référence à un processeur super-scalaire, avec exécution dans le désordre et spéculation:

- a) Expliquez l'impact de ces différents éléments micro-architectural sur le CPI :
 - 1. Nombre d'étages du pipeline
 - 2. Dimension des caches
 - 3. % de bonnes prédictions d'un prédicteur de branchement
- b) Identifiez un élément micro-architectural (qui n'est pas en a) et qui n'est pas lié aux caches) et qui a une influence importante sur le CPI. Expliquez comment cet élément influence le CPI.
- c) Pourquoi le traitement multi-processus simultané (SMT) est-il si simple à ajouter à ce type de processeur ?
- d) Question bonus : Le Pentium-M (Centrino) fait de la fusion de micro-opérations. Expliquez pourquoi cette fusion de micro-opérations peut réduire le CPI.