

Architecture des ordinateurs et systèmes d'exploitation

Corrigé du TD 9: Architectures superscalaires

Marcel Bosc

Christophe Dehlinger
Benoît Meister

Arnaud Giersch
Nicolas Passat

Mathieu Haefele

1. Opérations dépendantes

Quelle est la plus longue chaîne d'opérations dépendantes du fragment de programme suivant ?

```
LD r7, (r8)      ! stocker dans r7 la valeur se trouvant à l'adresse mémoire r8
SUB r10, r11, r12 ! r10 = r11 - r12
MUL r13, r7, r11 ! r13 = r7 * r11
ST (r9), r13     ! stocker en mémoire à l'adresse r9 la valeur du registre r13
ADD r13, r2, r1  ! r13 = r2 + r1
LD r5, (r6)      ! stocker dans r5 la valeur se trouvant à l'adresse mémoire r6
SUB r3, r4, r5   ! r3 = r4 - r5
```

Correction : La chaîne de dépendances la plus longue fait quatre instructions de long.

```
LD r7, (r8)
MUL r13, r7, r11
ST (r9), r13
ADD r13, r2, r1
```

2. Limites du parallélisme

Si le fragment de code de l'exercice précédent était exécuté sur un processeur superscalaire doté d'un nombre infini d'unités d'exécution avec une latence d'un cycle pour toutes les opérations, combien de temps faudrait-il pour émettre toutes les instructions de la séquence ? La question peut aussi se formuler ainsi : quelles limitations les dépendances du fragment de programme apportent-elles au temps d'émission de ses instructions ?

Correction : Avec un nombre infini d'unités d'exécution, la capacité du processeur à émettre les instructions en parallèle n'est plus limitée que par la longueur des chaînes d'instructions dépendantes du programme. La plus longue chaîne de dépendances a été identifiée dans l'exercice précédent et la deuxième chaîne la plus longue est une chaîne de deux instructions. Si toutes les dépendances dans la plus longue chaîne étaient des dépendances LAE, les instructions dans la chaîne devraient être émises en séquence, ce qui donnerait un temps d'émission de quatre cycles. Toutefois, l'une des dépendances correspond à une dépendance EAL, or les instructions entretenant une dépendance EAL peuvent être émises durant le même cycle. Ceci permet aux instructions ST (r9), r13 et ADD r13, r2, r1 d'être émises au même moment et réduit le temps d'émission total à 3 cycles.

3. Exécution IO (In Order)

Combien de temps faut-il pour émettre les instructions du fragment de code suivant sur un processeur superscalaire IO doté de 4 unités d'exécution pouvant exécuter chacune n'importe quel type d'instruction ? Les opérations mémoire (LD, ST) possèdent une latence de 2 cycles et toutes les autres opérations une latence d'un cycle.

```
ADD r1, r2, r3
SUB r5, r4, r5
LD r4, (r7)
MUL r4, r4, r4
```

```

ST (r7), r4
LD r9, (r10)
LD r11, (r12)
ADD r11, r11, r12
MUL r11, r11, r11
ST (r12), r11

```

Correction : Cette séquence de code prendrait 9 cycles à être émise.

Cycle	Unité 1	Unité 2	Unité 3	Unité 4
1	ADD r1, r2, r3	SUB r5, r4, r5	LD r4, (r7)	
2			latence	
3	MUL r4, r4, r4			
4	ST (r7), r4	LD r9, (r10)	LD r11, (r12)	
5	latence	latence	latence	
6	ADD r11, r11, r12			
7	MUL r11, r11, r11			
8	ST (r12), r11			
9	latence			

4. Exécution IO (In Order) avec unités d'exécution restreintes

Combien de temps faut-il pour émettre les instructions du fragment de code suivant sur un processeur superscalaire IO doté de deux unités d'exécution dont l'une ne peut exécuter que des opérations mémoire et l'autre que des opérations non mémoire ? Les opérations mémoire (LD, ST) possèdent une latence de 2 cycles et toutes les autres possèdent une latence d'un cycle.

```

LD r4, (r5)
LD r7, (r8)
ADD r9, r4, r7
LD r10, (r11)
MUL r12, r13, r14
SUB r2, r3, r1
ST (r2), r15
MUL r21, r4, r7
ST (r22), r23
ST (r24), r21

```

Correction : Le programme prendrait 13 cycles à être émis.

Cycle	Unité 1 (non mémoire)	Unité 2 (mémoire)
1		LD r4, (r5)
2		latence
3		LD r7, (r8)
4		latence
5	ADD r9, r4, r7	LD r10, (r11)
6	MUL r12, r13, r14	latence
7	SUB r2, r3, r1	
8	MUL r21, r4, r7	ST (r2), r15
9		latence
10		ST (r22), r23
11		latence
12		ST (r24), r21
13		latence

5. Exécution OOO (Out Of Order)

Combien de temps faudrait-il pour émettre les instructions du fragment de code suivant sur un processeur superscalaire OOO doté de deux unités d'exécution et dont toutes les instructions possèdent une latence d'un cycle ? Nous supposons que chaque unité d'exécution peut exécuter n'importe quel type d'instruction. Nous supposons par ailleurs que la fenêtre d'instruction du processeur est assez grande pour comprendre le fragment de code entier.

```
LD r1, (r2)
SUB r4, r5, r6
ADD r3, r1, r7
MUL r8, r3, r3
ST (r11), r4
ST (r12), r8
ADD r15, r14, r13
SUB r10, r15, r10
ST (r9), r10
```

Correction : Le programme prendrait 5 cycles à être émis.

Cycle	Unité 1	Unité 2
1	LD r1, (r2)	SUB r4, r5, r6
2	ADD r3, r1, r7	ST (r11), r4
3	MUL r8, r3, r3	ADD r15, r14, r13
4	ST (r12), r8	SUB r10, r15, r10
5	ST (r9), r10	

6. Exécution OOO (Out Of Order) avec unités d'exécution restreintes

Supposons que le processeur de l'exercice précédent possède une unité d'exécution qui ne peut exécuter que les instructions mémoire et une autre qui ne traite que les instructions non mémoire. Si tous les paramètres du processeur restent les mêmes, combien de temps faut-il pour émettre le fragment de code de l'exercice précédent ?

Correction : Le programme prendrait 6 cycles à être émis.

Cycle	Unité 1 (non mémoire)	Unité 2 (mémoire)
1	SUB r4, r5, r6	LD r1, (r2)
2	ADD r3, r1, r7	ST (r11), r4
3	MUL r8, r3, r3	
4	ADD r15, r14, r13	ST (r12), r8
5	SUB r10, r15, r10	
6		ST (r9), r10

7. Renommage de registres

Combien de registres matériels sont requis pour permettre au renommage de registres de rompre toutes les dépendances EAL et EAE du jeu d'instruction suivant ?

```
LD r1, (r2)
ADD r3, r4, r1
SUB r4, r5, r6
MUL r7, r4, r8
OR r8, r9, r10
SUB r11, r8, r12
DIV r12, r13, r14
ST (r15), r12
```

Correction : Le fragment de code utilise 15 registres architecturaux. En outre, il existe trois dépendances EAL entre ADD r3, r4, r1 et SUB r4, r5, r6, entre MUL r7, r4, r8 et OR r8, r9, r10, et entre SUB r11, r8, r12 et DIV r12, r13, r14. Il n'y a pas de dépendance EAE dans ce code. En conséquence, 18 registres matériels

sont nécessaires pour que le renommage des registres puisse rompre toutes les dépendances du programme (15 pour les 15 registres architecturaux et 3 pour renommer chacun des registres concernés par les dépendances EAL).