

TP 4. Les Processeurs actuels. Pipelining

1. Pipelining (1)

On considère le pipeline ci-dessous dédié aux instructions en virgule flottante (pour données FP Floating Point) :

<i>étage</i>	<i>pipeline P</i>
1	lit cache d'instructions
2	decode instruction
3	lit registres FP
4	execute FP / acces cache de donnees
5	execute FP / acces cache de donnees
6	execute FP / acces cache de donnees
7	ecrit registre FP

Le débit théorique maximum de ce pipeline est de 1 instruction/cycle.

On suppose un prédicteur de branchements parfait et les instructions déjà dans les caches.

Les instructions en virgule flottante (FP) sont pipelinées sur 3 cycles.

Une instruction FP dépendant de l'instruction immédiatement précédente doit attendre 2 cycles avant de rentrer dans le 1^{er} étage d'exécution.

On veut calculer l'expression : $ay^3 + by^2 + cy + d$. On propose 2 méthodes pour implémenter ce calcul :

- méthode 1 (de Horner) : $d + y(c + y(b + ya))$ opérations : 3 \times ; 3 +

- méthode 2 : $(d + cy) + ((y \times y) \times (ay + b))$ opérations : 4 \times ; 3 +

On remarquera que la méthode 2 demande 1 multiplication de plus que la méthode 1 (de Horner).

On supposera que y,a,b,c,d se trouvent déjà dans les registres flottants respectifs F0, F1, F2, F3, F4 et que le autres registres sont libres. On veut obtenir le résultat dans le registre F5.

a) Ecrire le pseudo-code assembleur pour les 2 méthodes.

b) Quelle méthode est la plus performante sur le pipeline considéré ?

2. Pipelining (2)

1. Pourquoi le pipelining améliore-t-il la performance ?
2. Quelles sont les limites de l'amélioration de performance apportée par le pipelining ?
3. Calcul du temps d'exécution d'une séquence d'instructions :

Exemple :

Rappel : latence = durée d'exécution, temps de traitement, d'une instruction

Supposons qu'un processeur non pipeliné possédant un temps de cycle (temps de traitement d'1 instruction) de 25 ns soit divisé en 5 étages de pipeline de latences respectives de 5, 7, 3, 6 et 4 ns. La latence de latch du pipeline (registre de sortie de chaque étage du pipeline) est de 1 ns. Le pipeline est supposé sans délais de branchement (prédiction de branchement parfaite, sans aléas).

- 3.1. Quelle est le temps de cycle (temps de traitement d'1 étage du pipeline) du processeur pipeliné ?
- 3.2. Quelle est la latence totale du pipeline (temps de traitement de tous les étages du pipeline) ?

Note :

Le temps d'exécution (en nombre de cycles) d'une séquence d'instructions d'un pipeline sans délais de branchement ni dépendance de données est :

nombre d'étages + nombre d'instructions – 1.
