

ARCHITECTURE DES ORDINATEURS
Examen Décembre 2010
3H - Tous documents autorisés
Les questions sont indépendantes

On utilise le jeu d'instructions NIOS II et le jeu d'instructions ARM.

PROGRAMMATION ASSEMBLEUR

Soit le code C

```
int a, b, c, d; // a, b, c et d sont initialement dans R1, R2, R3 et R4.
while (a != b) {
  if (a < b) c++;
  else d++;
  a += ((b - a) / 2);}

```

Question 1) Ecrire le code assembleur NIOS correspondant au code C, en essayant de minimiser le nombre de branchements.

Question 2) Ecrire le code assembleur ARM correspondant au code C

CACHES

Soit un cache de 4 Ko à correspondance directe, avec des lignes (blocs) de 32 octets. Le processeur a des registres de 32 bits et des adresses de 32 bits.

Question 3) Quels sont les nombres de bits nécessaire pour l'index, l'étiquette et le déplacement (adresse dans la ligne) ?

Soit le code C travaillant sur des flottants en double précision (64 bits).

```
double A[1024]; int i ;
for (i = 0; i < 512; i++)
  A[i] = A[i] + A[i+512];

```

Figure 1

Question 4) Pour un cache à écriture simultanée et écriture non allouée, quel est le nombre de défauts de cache lors de l'exécution du code C (figure 1). On rappelle qu'avec l'écriture non allouée, il n'y a pas de défauts de cache en écriture.

Question 5)

a) Donner une technique logicielle (modification du code) et une technique matérielle (modification du cache, sans changer la taille de ligne ou taille du cache) qui permettraient de diminuer le nombre de défauts de cache.

b) Après modification, quel serait le nombre de défauts de cache total pour le programme de la figure 1 ?

Question 6) Avec le cache initial (sans modification logicielle ou matérielle), mais à réécriture et allocation d'écriture, quel est le nombre de défauts de cache pour le code de la figure 1 ? On rappelle qu'avec l'écriture allouée, il y a des défauts de cache en écriture.

EXECUTION DE BOUCLES

On ajoute au jeu d'instructions NIOS II des instructions flottantes simple précision (32 bits) (Figure 2) et 32 registres flottants F0 à F31 (F0 est un registre normal).

Les additions, soustractions et multiplications flottantes sont pipelinées. Une nouvelle instruction peut démarrer à chaque cycle. Les latences sont de 2 cycles pour LF et de 3 cycles pour les instructions flottantes.

Les branchements ne sont pas retardés.

LF	2	LF ft, déplac(rs)	ft ← MEM [rs +SIMM]
SF	1	SF ft, déplac(rs)	ft → MEM [rs + SIMM]
FADD	3	FADD fd, fs,ft	fd ← fs + ft (addition flottante simple précision)
FMUL	3	FMUL fd, fs,ft	fd ← fs * ft (multiplication flottante simple précision)
FSUB	3	FSUB fd, fs,ft	fd ← fs - ft (soustraction flottante simple précision)

Figure 2 : Instructions flottantes ajoutées (Ce ne sont pas les instructions NIOS)

Soit le programme assembleur P1, qui travaille sur un tableau de flottants simple précision (float) X[N] rangé en mémoire, avec N = 512. L'adresse de X[0] est initialement contenue dans le registre R1.

```

ADDI R2, R1, 1024
Boucle : LF F1, 0(R1)
         LF F2, 1024(R1)
         FADD F3,F1,F2
         SF F3, 0 (R1)
         FSUB F2,F2,F1
         SF F2, 1024(R1)
         ADDI R1,R1,4
         BLT R1,R2, Boucle
    
```

Question 7) Donner le code C correspondant au programme P1

Question 8) En optimisant, montrer l'exécution cycle par cycle du programme assembleur P1 et donner le nombre de cycles par itération de la boucle du programme assembleur en supposant qu'il n'y a ni pénalité de branchement, ni défauts de cache. Quel est le temps d'exécution total en cycles d'horloge ?

Question 9) Quel est le nombre de cycles par itération de la boucle initiale avec déroulage d'ordre 4, en supposant qu'il n'y ni pénalité de branchement, ni défauts de cache ?

PREDICTION DE BRANCHEMENT

Soient trois branchements b1, b2 et b3 appartenant à une boucle interne, qui est exécutée de nombreuses fois à l'intérieur d'une boucle externe. La boucle interne est exécutée 10 fois, et les branchements ont le comportement donné Figure 3

	1	2	3	4	5	6	7	8	9	10
B1	N	N	N	N	P	N	P	P	P	P
B2	N	P	P	P	N	P	N	N	N	N
B3	P	P	P	P	P	P	P	P	P	N

Figure 3 : comportement des branchements (P=pris ; N = non pris).

Question 10) En supposant que la boucle interne ait déjà été exécutée un certain nombre de fois, quel est le nombre de bonnes prédictions pour les trois branchements avec

- a) un prédicteur statique « toujours pris »
- b) un prédicteur dynamique « 1 bit »
- c) un prédicteur dynamique « 2 bits ».