

# Computer organisatie

## 7 february, 2005 (18:30-21:30)

- Dit examen bestaat uit 5 vragen met subvragen. Elke subvraag is 10 punten waard.
- Begin elke vraag op een nieuwe bladzijde. Zet overal je naam en studentnummer boven.
- Gebruik van boeken of ander studiemateriaal is niet toegestaan, een rekenmachine wel.
- Houd je antwoorden bondig. Schrijf niet met potlood of rode inkt.

### Vraag 1

(a) (i) Leg in woorden uit wat de 'CPI' en de 'CPU-tijd' van een programma zijn. (ii) Druk de CPU-tijd (in seconden) uit in CPI en IC (waarbij  $IC = \text{het aantal uitgevoerde instructies}$ ). Een processor heeft een kloksnelheid van 1 GHz en de specificatie geeft de volgende timing informatie:

| Type                           | CPI            |
|--------------------------------|----------------|
| ALU operations                 | 1              |
| floating point (FP) operations | $x$ (onbekend) |
| Branches                       | 3              |
| Loads and stores               | 2              |

(b) In een bepaald programma zijn 30% van de operaties ALU instructies, 5% FP instructies, 25% Branches, 20% Loads en 20% stores. Geef een formule (uitgedrukt in  $x$ ) om de gemiddelde CPI te berekenen uit de CPI's uit de tabel.

Het volgende programma wordt nu uitgevoerd (gegeven in pseudo-assembly):

| Code                           | Description                                                                              |
|--------------------------------|------------------------------------------------------------------------------------------|
| 1. <code>LOAD R1, #10</code>   | load in register R1 de waarde 10                                                         |
| 2. <code>LI: SQR R1, R1</code> | bereken de (floating point) vierkantsoverl van de waarde in (floating point register) R1 |
| 3. <code>SUB R1, R1, #1</code> | trek 1 af van de inhoud van R1, resultaat wordt in R1 gezet                              |
| 4. <code>BRZ R1, L1</code>     | als inhoud van R1 niete gelijk is aan 0, spring naar L1                                  |

Neem aan dat F1 initieel de waarde 100.0 bevat.

- (c) (i) Geef van elk van de instructies aan hoe vaak deze wordt uitgevoerd. (ii) Beschrijf *in één zin* wat dit programma doet.
- We zijn nieuwsgierig naar  $x$ , de waarde van de CPI voor de FP operaties. Ga er vanuit dat SQR de enige FP instructie is. Neem tevens aan dat de code wordt uitgevoerd op een processor met een kloksnelheid van 1 GHz. Om  $x$  te vinden moeten we de CPU time van bovenstaand programma. Deze blijkt 7 seconden te zijn.

(d) Bepaal, gebruikmakend van uw eerdere antwoorden, een waarde voor  $x$  (uitgedrukt in  $T$ ).

### Vraag 2

In een code bestaan enkel de volgende codewoorden:

000000  
000111  
111000  
111111

- (a) Wat is de Hamming distance van deze code woorden?
- (b) (i) Hoeveel bitfouten kunnen met deze code gedetecteerd worden? (ii) Hoeveel gecorrigeerd?
- (c) Om gebelgen aan te spreken wordt gebruik gemaakt van een bus met  $A$  adresslijnen en  $D$  datalijnen. (i) Kan  $A$  toenemen zonder dat  $D$  toeneemt (leg uit)? (ii) Wat zijn de voordelen als  $A$  toeneemt? (iii) Wat zijn de voordelen als  $D$  toeneemt?

We willen een digitaal circuit bouwen dat de volgende functie uitdrukt (alle waarden zijn 1 bit en INVB betekent de inverse van B):  $Q = ((A \text{ and } B) \text{ or } (A \text{ and } D \text{ and } INVB)) \text{ or } (C \text{ and } D)$

(d) Geef de waarheidstabel voor dit circuit.

(e) Teken het circuit. Gebruik alleen NAND and NOR gates.

### Vraag 3

Zie de microinstructie in Figuur 1.

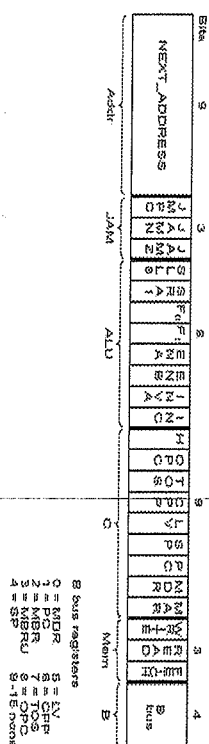


Figure 1: Microinstruction

(a) Uit de figuur blijkt dat er 4 bits gebruikt worden voor de selectie van wat er op de B bus gezet moet worden, terwijl er 9 bits gebruikt worden voor de selectie van registers die moeten lezen van de C bus. Verklaar dit verschil.

(b) Leg (in maximaal 10 regels) uit hoe steeds, nadat een microinstructie is uitgevoerd, de volgende microinstructie wordt geselecteerd.

### Vraag 4

Gegeven een byte-adresseerbare machine met 32 bits adressen, 8 bytes in een woord, en 32 woorden in een cacheline. De grootte van de cache is 64 KByte en de cache is direct-mapped. Initieel is de cache leeg.

(a) Gegeven het decimale adres 768. In welke cacheline komt de byte die zich op dit adres bevindt? Leg uit. Geef ook de tag die erbij hoort.

(b) Leg uit wat er gebeurt als hierna de volgende (hexadecimale) adressen worden gelezen (geef aan of de memory operatie een 'hit' of een 'miss' oplevert en in het geval van een miss, of het resulteert in een 'replacement'): (i) 770, (ii) 66304, (iii) 770.

(c) Zullen een of meer van de cache misses uit je vorige antwoord bits worden als er een 2-way set associatieve cache wordt gebruikt (zelfde cache grootte, 8 byte woorden, 32 woorden in een cache line)? Ga uit van leest recently used replacement. Zouja, welke en waarom? (Neem opnieuw aan dat de eerste read operatie plaatsvindt op adres 768.)

(d) We vergelijken direct-mapped caches met two-way set associatieve caches. Leg uit: (i) welke volgens u de hoogste 'miss rate' heeft, (ii) welke volgens u de hoogste 'hit time' heeft.

### Vraag 5

(a) Wat zijn de verschillen tussen superscalars en VLIW (of EPIC) processoren?

(b) In het boek wordt uitgelegd hoe de Intel IA-64 *prediction* ondersteunt: alle instructies zijn conditioneel. Leg uit *m.b.v. een programma-voorbeeld* hoe deze conditionele instructies kunnen leiden tot snellere executie.

(c) (i) Wat is het verschil tussen: sequential consistency en processor consistency? (ii) Stel: we gebruiken momenteel een processor-consistente multiprocessor om een applicatie  $X$  te draaien. Op een dag besluiten we een nieuwe multiprocessor te kopen en die is sequentieel consistent. Moeten we de bestaande code van  $X$  aanpassen? (Leg uit.)