

Computer organisatie

9 juni 2009 (18:30-21:15)

- Dit examen bestaat uit 7 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) We meten de performance van een processor met een onbekend benchmark programma. We ontdekken dat de CPI 0.9 is. Beantwoord nu de volgende vragen:

1. Wat betekent dat?
2. Is het realistisch dat een $CPI < 1$ is?
3. Zegt dat iets over de performance van een (ander) programma dat wij willen draaien op deze processor? Zo ja, wat? Zo nee, waarom niet?

(b) We hebben gemeten dat een programma dat draait op een bepaalde processor 'V1' 25% van zijn tijd spendeert in de divr instructie (de floating point division). De totale tijd die in floating point operaties wordt doorgebracht is 60%. De volgende generatie van de processor moet het programma beter ondersteunen. Er zijn twee opties:

- V2: maak de divr instructie 10x sneller;
- V3: maak alle floating point instructies 2x sneller.

Welke optie kiest u en waarom?

(c) Om de processor verder te versnellen wordt hij uitgerust met een pipeline. Er zijn drie opties:

- 5 pipeline stages (elke stage duurt 0.5ns, behalve stage 3 die 2ns duurt);
- 6 pipeline stages (elke stage duurt 1ns);
- 35 pipeline stages (elke stage duurt 0.95ns).

Welk ontwerp kiest u? Motiveer uw antwoord.

Vraag 2

In een code bestaan enkel de volgende codewoorden:

```
000000
000111
111000
111111
```

- (a) Wat is de Hamming afstand van deze code woorden?
- (b) (i) Hoeveel bitfouten kunnen met deze code gedetecteerd worden? (ii) Hoeveel gecorrigeerd?
- (c) Om geheugen aan te spreken wordt gebruik gemaakt van een bus met A adreslijnen 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 en INVD de inverse van D):

$$Q = ((A \text{ and } INVD) \text{ or } (A \text{ and } D \text{ and } INVB)) \text{ or } (C \text{ and } D)$$

(d) Geef de waarheidstabel voor dit circuit.

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

(f) In het geheugen hebben de eerste twee bytes de volgende waarden. De byte op adres 0 heeft de waarde 00001111 en de byte op adres 1 heeft de waarde 00000000. Welk 16 bits getal staat hier op een Little Endian machine en welk 16 bits getal op een Big Endian machine?

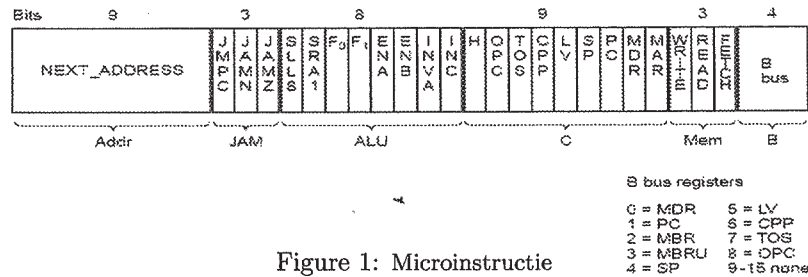


Figure 1: Microinstructie

Vraag 3

(a) De ILOAD instructie van de IJVM wordt gebruikt om een lokale variabele op de stack te plaatsen. De instructie heeft een 8 bits operand, maar door middel van de WIDE instructie kun je ook aangeven dat de instructie een 16 bits operand moet gebruiken. Daar is dan wel een extra instructie voor nodig. We denken dat we met dezelfde hardware complexiteit een processor kunnen bouwen waarbij de ILOAD instructie altijd een 16 bits operand gebruikt.

Welk ontwerp is beter? Leg uit.

De twee vormen van ILOAD staan hieronder nog eens opgenomen in 2 code-fragmenten (waarbij index1 1 byte is en index2 2 bytes).

Fragment 1	Fragment 2
ILOAD index1	WIDE
	ILOAD index2

(b) Zoals je weet wordt er in de microarchitectuur in het boek gebruik gemaakt van microinstructies waarbij elke instructie bepaalt welke instructie de volgende moet zijn (d.m.v. het `next_address` veld in de microinstructie - zie Figuur 1). Leg uit hoe JMPCL wordt gebruikt om ervoor te zorgen dat de juiste microinstructies worden uitgevoerd voor beide vormen van de ILOAD instructie.

(c) Waarom is het belangrijk dat instructie-*issue* sneller is dan instructie-*executie* in SuperScalar processors?

Vraag 4

Geef van de volgende uitspraken aan of ze goed of fout zijn (JA/NEE):

1. Als twee programma's beurtelings op dezelfde processor draaien moeten we steeds bij het wisselen de inhoud van de TLB 'flushen' (in elk geval alle entries die bij het niet langer actieve programma horen).
2. Een (2,2)-bits branch predictor is vooral zo goed omdat hij naast informatie over de lokale branch gebruik maakt van de 'global history' gebruikt om de branch prediction nauwkeuriger te maken.
3. Het 'relocation' probleem bij het linken kan worden opgelost door gebruik te maken van position independent code (PIC).
4. Een 2-way set associative cache heeft over het algemeen een lagere miss-rate dan een direct-mapped cache, maar als de cache groter wordt maakt het niet meer zoveel uit.
5. Een voordeel van scheduling in hardware in superscalars (vergeleken met VLIW processoren) is dat het makkelijker is om compilers te schrijven.
6. WAR hazards kunnen vaak opgelost worden door middel van register renaming, maar leveren meestal nog wel een korte 'stall' van 1 cycle in de pipeline op.
7. Voor cache memory wordt meestal gebruik gemaakt van Static RAM.
8. Pass 2 van een two-pass assembler is bedoeld om het programma dat uit meerdere modules bestaat aan elkaar te 'linken'.
9. Een verschil tussen een macro en een functie-aanroep is dat bij een functie-aanroep de parameters op de stack gezet (kunnen) worden. Bij macros is dat nooit het geval.
10. Om externe fragmentatie te reduceren is het beter om pages zo groot mogelijk te maken.

Vraag 5

Gegeven een machine die zowel segments als pages ondersteunt. Een adres bestaat uit twee delen: een 8-bits segment selector en een 24-bits ge-paged adres binnen dat segment (zie ook Fig. 2.a). De page size is 4 Kbyte.

(a) Hoeveel entrees telt de page table?

Een schematisch overzicht van de segment descriptor table en een page table staan in Figuur 2.b.

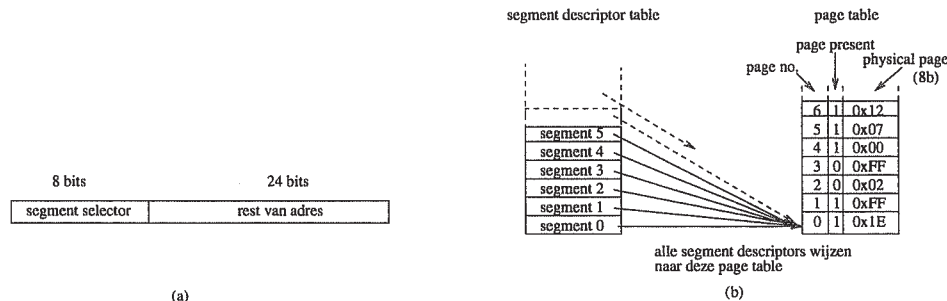


Figure 2: Segment descriptor table en page table

(b) (i) Hoeveel elementen telt de segment descriptor table? (ii) Afgaande op de figuur, gebruikt het besturingssysteem echt segmentatie? Waarom wel/niet?

(c) (i) Hoeveel *fysieke* pages heeft deze computer (hint: zie page table)? (ii) Hoeveel bits telt een fysiek adres in totaal? (iii) Hoeveel fysiek geheugen is er dus aanwezig?

Zoals gezegd, de segment selector is 8 bits lang en het adres binnen dat segment is 24 bits. Stel dat de segment selector de waarde 3 bevat en het 24 bits adres binnen het segment bevat de decimale waarde 4419 (= hexadecimaal: 0x1143 = binair: 000000000001000101000011).

(d) Wat wordt het fysieke adres? Leg ook uit hoe het bijbehorende fysieke adres wordt gevonden.

(e) Stel je hebt een machine met 4 fysieke pages van 4KB elk. Geef een programma waarvoor LRU page replacement ongewenst gedrag vertoont.

Vraag 6

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 hits worden als er een 2-way set associative cache wordt gebruikt (zelfde cache grootte, 8 byte woorden, 32 woorden in een cache line)? Ga uit van least recently used replacement. Zoja, 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 associative caches. Leg uit: (i) welke volgens u de hoogste 'miss rate' heeft, (ii) welke volgens u de hoogste 'hit time' heeft.

(e) Register renaming is niet geschikt voor het verwijderen van stalls die het gevolg zijn van RAW (read after write) hazards. Noem een techniek die soms helpt om RAW stalls te voorkomen.

Vraag 7

Bedenk zelf een leuke tentamenvraag voor dit vak en geef het antwoord. Inzichtvragen krijgen de volledige 10 punten (mits correct) en kennisvragen krijgen 5 punten (mits correct).