

Student:

Collegekaartnummer:

Tentamen Computersystemen voor AI programmeurs

baiCOSY06 2e jaar bachelor AI, 1e semester 23 oktober 2012 12u-14u REC-M 3.01,

Het is niet toegestaan communicatieapparatuur zoals tablets en telefoons te gebruiken: zet de mobiele telefoon uit!

Gebruik van een rekenmachine en de boeken behorende bij dit vak (Computer Systems, en Van 0 en 1 tot processor) is toegestaan. Succes!

vraag 1: MIPS machine

Stel dat in een MIPS Single Cycle machine de verschillende fases in het uitvoeren van een instructie de volgende tijden vergen:

Instruction Fetch 0,1 ns
Instruction Decode 0,15 ns
Execution 0,35 ns
Data Memory Access 0,1 ns
Register Write Back 0,1 ns

- a) Wat mag de frequentie van de klok van deze processor maximaal zijn? Wat is de throughput (in GIPS) ?

De throughput van deze processor kan worden verhoogd door er een 5-stage pipeline machine van te maken. De tussengevoegde pipeline registers hebben 50 ps nodig.

- b) Wat is de maximale klokfrequentie nu? Wat is nu de throughput ?
c) Noem twee oorzaken waarom de winst in throughput zoals zojuist berekend in de praktijk niet gehaald zal worden (afhankelijk van hoe de opvolging van de instructies is).

vraag 2: NOPS in de pipeline

In een 5-stage pipeline (zonder forwarding) wordt onderstaand programma geëxecuteerd:

ADD \$2, \$3, \$1
AND \$12, \$2, \$5
OR \$13, \$6, \$2
SUB \$7, \$6, \$1
SW \$8, 100, \$12
LW \$9, 100, \$13

- a) Voeg in bovenstaand programma NOPs toe zodat het programma correct werkt.
b) Herschrijf het programma door instructies te verwisselen. Minimaliseer hierbij het aantal NOPs.

Student:

Collegekaartnummer:

vraag 3: Direct mapped cache

De Harvard processor (Hoofdstuk 8 in Van 0 en 1...) is voorzien van een data-geheugen van 4096 (16-bits) woorden.

a) Hoeveel bits zijn nodig om dit geheugen te adresseren?

Voor snellere toegang tot dit geheugen is een direct mapped cache toegevoegd met een capaciteit van 256 woorden. Elke cache-line bevat 8 woorden.

b) Geef aan welke bits van het geheugenadres bepalend zijn voor:

- welke set geselecteerd wordt (set index);
- welk woord uit de set geselecteerd wordt (block offset);
- de tag.

c) Uit hoeveel bits bestaat elke cache line?

Er wordt een woord uit het geheugen opgehaald (LW-instructie) van adres 0xA59. Dit geheugen adres bevat de waarde 0x1234, alle overige geheugenlocaties staan op 0xFFFF.

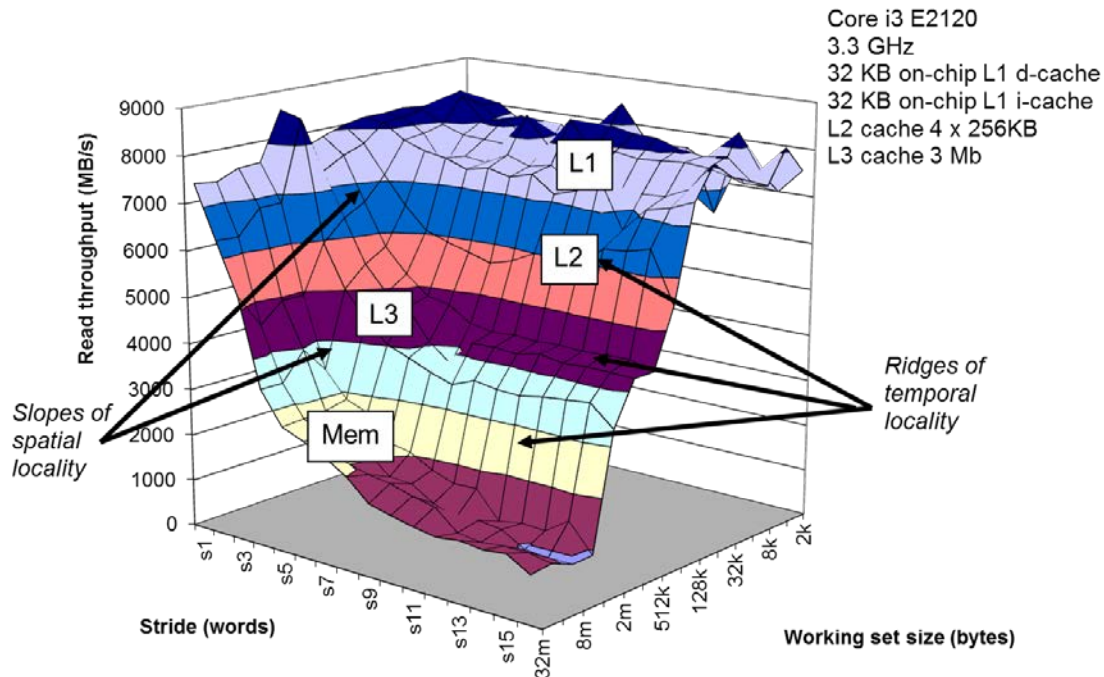
d) In welke cache set komt deze 0x1234 terecht? (de hoeveelste cache-line van de cache)

Hoe ziet deze cache-line er precies uit: geef de gehele inhoud van deze cache-line weer.

vraag 4

U werkt in practicumzaal G0.18. De computers in deze zaal hebben Core i3 processor van de tweede generatie (*Ivy / Sandy Bridge microarchitecture*).

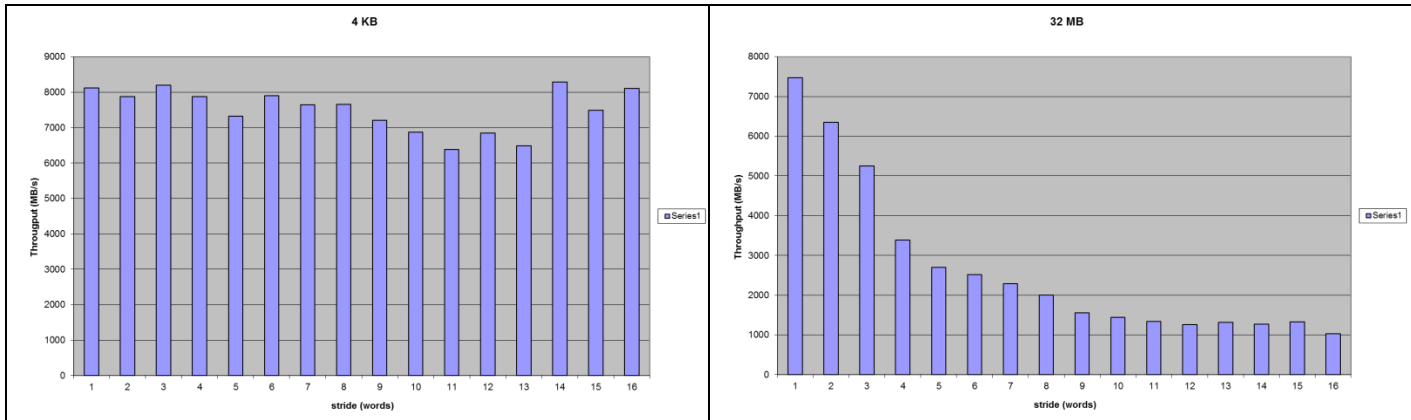
U meet het geheugen als functie van de *spatial* en *temporal locality*. U krijgt het volgende resultaat:



In de volgende figuur zijn twee doorsneden van deze figuur te zien, voor respectievelijk een kleine en grote *working set*:

Student:

Collegekaartnummer:

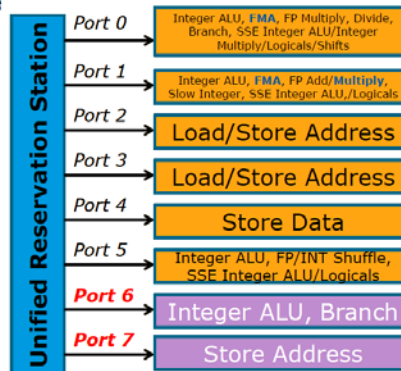


- Met welke snelheid kan data uit het RAM geheugen gelezen worden?
- Met welke snelheid kan data uit de *cache* gelezen worden?
- Reken dit voor een *4-byte word* om in nanoseconden en tikken.

De architectuur van de volgende generatie *Core* processors is al aangekondigd. Vorige maand op het Intel Developer Forum gaf Ronak Singhal *et al* een voorproefje van de nieuwe microarchitectuur (*Haswell*):

Microarchitecture Enhancements

- Energy-efficient performance**
 - Features to improve existing software
- No change in key pipelines**
- Example improvements**
 - Improved code fetch BW
 - Better branch prediction
 - Larger OOO window and corresponding structures
 - Increased throughput via 2 new dispatch ports
 - Larger L2 TLB
 - Lower virtualization latencies



Continue To Push Power-Efficient Performance Within the CPU

IDF2012
INTEL DEVELOPER FORUM

15

Courtesy Ronak Singhal *et al*, 'Intel® Next Generation Microarchitecture', Sept. 2012

In een van de slides is met rood twee nieuwe *dispatch ports* aangegeven. Deze *dispatch ports* zijn de toegang tot de *Functional Units* (zie bijvoorbeeld Fig. 5.11 in het boek).

- Zal de functie `combine6` (bladzijde 549) bij deze architectuur uitgebreid kunnen worden met een 3^e *critical path*? Wat is het effect voor datotypen van het soort *int* en *float*? Motiveer uw antwoord.

Student:

Collegekaartnummer:

vraag 5

Bekijk het volgende stukje code. Om ruimte te besparen, is in deze code geen *error checking* geïmplementeerd. Men mag er vanuit gaan dat alle functies met succes worden uitgevoerd:

```
#include "csapp.h"
pid_t pid;

void handler1(int sig) {
    printf("zip");
    fflush(stdout);           /* Flushes the printed string to stdout */
    Kill(pid, SIGUSR1);
}

void handler2(int sig) {
    printf("zap");
    exit(0);
}

int main() {
    Signal(SIGUSR1, handler1);
    if ((pid = Fork()) == 0) {
        Signal(SIGUSR1, handler2);
        Kill(getppid(), SIGUSR1);
        while(1) {};
    } else {
        pid_t p; int status;
        if ((p = Wait(&status)) > 0) {
            printf("zoom");
        }
    }
    exit(0);
}
```

Welke string print dit programma als output? Leg uit waarom!

vraag 6

Het volgende programma bevat een fout. Het is de bedoeling dat de *thread* één seconde slaapt, en dan de *string* "Hello, world!" print. Echter, als dit programma gestart wordt op één van onze machines, wordt er niets geprint.

- a) Waarom wordt er niets geprint?
- b) Je kan de bug oplossen door de `exit` functie op lijn 9 te vervangen door één van de Pthread functies. Er zijn twee Pthread functies die hier geschikt voor zijn. Welke twee? Geef ook aan hoe deze functies op lijn 9 aangeroepen moeten worden.

```
1     #include "csapp.h"
2     void *thread(void *vargp);
3
4     int main()
5     {
6         pthread_t tid;
7
8         Pthread_create(&tid, NULL, thread, NULL);
9         exit(0);
10    }
11
12    /* thread routine */
13    void *thread(void *vargp)
14    {
15        Sleep(1);
16        printf("Hello, world!\n");
17        return NULL;
18    }
```