# Obligatorisk øving 2 i datateknikk

Dette er den andre obligatoriske øvingen i datateknikk.

## Oppgave 1 Grunnleggende virkemåte

Den hypotetiske maskinen i leksjonen Grunnleggende Virkemåte brukte 16-bits instruksjoner der 4 bits var opkode og 12 bits var operand (enten en adresse til minnet eller en IO-adresse)

1. Hvor mange opkoder kan vi skille mellom på denne maskinen? Grunngi svaret grundig

Svar:

Vi har 16 mulige opkoder, der en bit består enten av 0 eller 1, og 24 = 16. Vi tar antall mulige verdier og opphøyer med antall bits

1. Hvor mange adresser kan vi skille mellom på denne maskinen? Grunngi svaret grundig

Svar:

Samme metode som oppgaven over, får vi 212 = 4096

1. Skriv bitmønsteret nedenfor på heksadesimal form (Vis fremgangsmåten tydelig):  
   0101000011001111.

Svar:

|  |  |  |  |
| --- | --- | --- | --- |
| 0101 | 0000 | 1100 | 1111 |
| 5 | 0 | C | F |

50CF

1. Hvis bitmønsteret i oppgave c) er en instruksjon til den hypotetiske maskinen, hvilken instruksjon er det da? (Hvilken opkode og hvilken adresse)? Grunngi svaret grundig

Svar:

Opkoden 01012 er en ADD funksjon som adder tallet fra minnelokasjon 0CF med tallet fra AC og legger resultatet inn i AC.

1. Hvis bitmønsteret i oppgave c) er data til den hypotetiske maskinen (altså heltall med fortegn som står beskrevet i leksjonen), hvilken verdi representerer bitmønsteret? (Svaret skal oppgis desimalt, og svaret skal grunngis grundig)

Svar:

|  |
| --- |
| 0 101.0000.1100.1111 |
| + 214+212+27+26+23+22+21+20 = +20687 |

Tallet er positiv fordi den 15 bit-verdien er 0

## Oppgave 2 CPU, minne og buss

I leksjonen ble parallelle synkrone busser beskrevet. På moderne PCer er det en slik buss som brukes som minnebuss. Synkron buss betyr at en ny handling bare kan starte når det kommer en klokkepuls. I leksjonen så vi at det ved lesing fra minnet brukes en klokkepuls for å overføre adressen, så ventet vi en klokkepuls på at minnet skulle finne frem til rett lokasjon. Til slutt brukes en klokkepuls til å overføre data fra minnet til CPU. Se kap 1.3.4 i leksjonen om busser.

Men hva skjer hvis minnet er så tregt at det ikke rekker å hente data fra rett lokasjon i løpet av en klokkesyklus. Jo, da må vi vente flere klokkesykluser – og alltid et helt antall klokkesykluser. I denne oppgaven skal vi regne på hvor mange pulser vi må vente.

1. Anta at vi har en (synkron og parallell) buss med frekvens 800 MHz (800 millioner klokkepulser hvert sekund). Hvor lang tid går det mellom hver puls på en slik buss?

Svar:

800 Mhz = 1/sek -> sek = 1 / 800 Mhz = 1.25E-9 sek = 1.25 ns

1. Anta at DRAM-minnet som denne bussen er tilkoblet har en aksesstid på 35 ns. Hvor mange klokkepulser må vi vente?

Svar:

35 ns/1.25 ns = 28. Vi må vente 28 klokke pulser mellom hver aksesstid.

1. Anta at prosessoren som bruker bussen greier å utføre en milliard instruksjoner pr sekund. Hvor lang tid tar da hver instruksjon?

Svar:

1/1.000.000 sek = 1ns + 35 ns aksesstid -> 36 ns

*Kommentar: Hvis du synes det er noe som ikke rimer så kan du ha regnet helt rett likevel. Moderne prosessorer er mye kjappere enn DRAM-minnet, og denne oppgaven er ment som en motivasjon for å jobbe med neste modul i emnet; den som omhandler cache og systemarkitektur.*

## Oppgave 3 Disker, IO og avbrudd

I leksjonen om Sekundærlager ble harddisken Quantum Fireball 3.8 GB omtalt. Den brukte ZBR, og på den ytterste sonen ble hvert spor delt opp i 232 sektorer. Det lagres 512 Byte på hver sektor, og rotasjonshastigheten er 7200 omdreininger pr minutt.

1. Hvor mange Bytes lagres langs hvert av de ytterste sporene?

Svar:

232\*512 = 118 784 = 116 KB

1. Hvor mange bytes pr sekund overføres ved fortløpende lesing langs et spor ytterst på platen?

Svar:

7200/60\*118784=14 280 KB/s = 13.59MB/s

1. Hvor lang tid tar det mellom hver gang det ankommer en ny byte fra disken?

Svar:

1/13.59MB/s = 7.02E-8s/B

1B ankommer hvert 70.2 ns

1. Anta prosessoren fra oppgave 3. Hvor mange instruksjoner utfører denne prosessoren mellom hver gang det kommer en ny byte fra disken ved fortløpende lesing?

Svar:

70.2ns/B/36ns=1.95 ~ 2 Instruksjoner per byte

## Oppgave 4 Aksesstider

Data lagres på mange ulike lager-teknologier i en datamaskin. Lagerteknologiene spenner over en enorm variasjonsbredde av aksesstider: fra CPU-register (som har umiddelbar aksess) og til bånd (der aksesstiden måles i sekunder). Det er viktig å ha en ide om omtrentlig aksesstid for de viktigste teknologiene.

Hva er omtrentlig aksesstid for følgende teknologier (*Kommentar: Med omtrentlig aksesstid menes størrelsesorden, altså om aksesstiden måles i ns, ms osv.)*:

1. SRAM

Svar:

1-10 ns

1. DRAM

Svar:

35-70 ns

1. SSD

Svar:

0,1ms (100 us)

1. Harddisk

Svar:

4-12 ms

1. CD/DVD

Svar:

100-300 ms