

**Norges teknisk-naturvitenskapelige universitet
Institutt for datateknikk og informasjonsvitenskap**



EKSAMENSOPPGAVE I FAG TDT4160 – DATAMASKINER GRUNNKURS

Faglig kontakt under eksamen: Jon Olav Hauglid

Tlf.: 93440

Eksamensdato: 15. desember 2005

Eksamenstid: 09.00-13.00

Tillatte hjelpemiddel: D: Ingen trykte eller håndskrevne hjelpemiddel tillatt. Bestemt, enkel kalkulator tillatt.

Språkform: Bokmål

Oppgave 1 – Flervalgsspørsmål ("multiple choice") – 30 %

Denne oppgaven skal besvares på eget svarark sist i oppgavesettet. Dersom du finner flere alternativer som synes å passe, setter du kryss for det ene som passer best. For å unngå at gode tippere blir belønnet, vil et galt svar gi færre poeng enn om oppgaven forblir ubesvart.

- a) Hvilken av de følgende påstandene om Pentium 4 er **ikke riktig**?
1. Pentium 4 er en 32 bits prosessor.
 2. Pentium 4 har et langt samlebånd (mange steg).
 3. Pentium 4 er designet med fokus på ytelse vha. høy klokkefrekvens.
 4. Pentium 4 har instruksjoner med fast lengde og fast format.
 5. Pentium 4 oversetter instruksjoner til RISC-lignende mikrooperasjoner før utføring.
- b) Hvilken av de følgende påstandene om UltraSPARC III er **ikke riktig**?
1. UltraSPARC III er en RISC-prosessor.
 2. UltraSPARC III har relativt få registre, fokusert på hovedlager.
 3. UltraSPARC III er en 64 bits prosessor.
 4. UltraSPARC III er en superskalar prosessor.
 5. UltraSPARC III har nivå-2 hurtigbuffer utenfor prosessorbrikken.
- c) Hvilken av de følgende påstandene om 8051 er **ikke riktig**?
1. 8051 er en mikrokontroller.
 2. 8051 har ingen hurtigbuffer.
 3. 8051 har et fem-stegs samlebånd.
 4. 8051 er designet med fokus på pris, strømforbruk og størrelse.
 5. 8051 har separat minne for instruksjoner og data.
- d) Hvilken av alternativene under er **ikke** en årsak til å ha mer enn en buss i en datamaskin?
1. Med kun en buss, blir denne gjerne en flaskehals.
 2. Flere busser gir økt risiko for overflyt.
 3. Eksterne enheter har gjerne svært forskjellig datarate.
 4. Jo lengre en buss er, jo lengre tid tar det å overføre data.
 5. Noen eksterne enheter har seriell dataoverføring, mens andre har parallell.
- e) Hvilken av alternativene under er en ulempe med direkte adressering?
1. Gir gjerne et langt instruksjonsformat.
 2. Feltlengden begrenser operandstørrelsen.
 3. Det tar ekstra tid å slå opp i et register før man aksesserer hovedlager.
 4. Målinstruksjonen ved betingede hoppinstruksjoner ("conditional branches") kan være vanskelig å finne.
 5. Alle alternativer 1-4 er ulemper med direkte adressering.
- f) Hvilken av de følgende påstandene om samlebånd ("pipeline") er riktig?
1. Samlebånd reduserer tiden det tar å utføre en enkelt instruksjon.
 2. Samlebånd reduserer tiden det tar å utføre en serie med instruksjoner.
 3. Samlebånd samler først sammen en mengde instruksjoner som så blir utført samtidig.
 4. Samlebånd øker treffraten på hurtigbuffer.
 5. Alle alternativer 1-4 er riktige påstander om samlebånd.

- g) Hvilken av de følgende påstandene om RISC er riktig?
1. RISC-prosessorer har instruksjoner som er kompliserte å dekode.
 2. De fleste instruksjoner i RISC-prosessorer kan aksessere hovedlager.
 3. En RISC-prosessor utfører instruksjoner vha. mikroprogram.
 4. RISC-prosessorer har små og enkle instruksjonssett.
 5. RISC-prosessorer har mange adresseringsmodi.
- h) Hvilken av de følgende påstandene om multibatamaskin og multiprosessor er riktig?
1. Prosessorer i multibatamaskiner deler hovedlager.
 2. Med mange prosessorer er multiprosessor bedre enn multibatamaskin.
 3. Både multiprosessor og multibatamaskin er eksempler på MIMD (Multiple Instruction Multiple Data).
 4. En multibatamaskin er det samme som en superskalar prosessor.
 5. Ingen av alternativene 1-4 er riktige.
- i) Hvilken av de følgende påstandene om PCI og PCI Express er riktig?
1. PCI er ikke egentlig en buss, men punkt-til-punkt-forbindelse.
 2. PCI er parallell mens PCI Express er seriell.
 3. Med PCI kan man overføre adresse og data samtidig.
 4. Både PCI og PCI Express er asynkrone busser.
 5. PCI Express er en egen overføringsmodus som PCI-bussen bruker når bussmaster har spesielt dårlig tid.
- j) Registeromdøping ("register renaming") er en teknikk som ...
1. ... gjør det vanskeligere å skrive assemblykode.
 2. ... er unødvendig for CISC-prosessorer.
 3. ... reduserer antall sanne dataavhengigheter.
 4. ... reduserer antall avhengigheter i superskalare samleband.
 5. ... ikke kan brukes av prosessorer som støtter registeradressering.
- k) Prefetch er en teknikk som ...
1. ... går ut på at betingede hopp ("conditional branch") utføres vha. to instruksjoner (eks: Compare/Branch).
 2. ... IA-64 bruker for å bestemme hvilken utførende enhet en instruksjon skal utføres av før instruksjonen blir hentet fra hovedlager.
 3. ... går ut på å hente data og/eller instruksjoner fra hovedlager før de trengs.
 4. ... går ut på at flyttall normaliseres før addisjon og subtraksjon utføres.
 5. ... brukes for å redusere tiden fra et register blir lest til ALU kan utføre operasjon på registerverdien.
- l) Hva vil det si at en prosessor er superskalar?
1. Den er spesiallaget for heltallsoperasjoner.
 2. Den bruker 1 klokkesyklus på å utføre en instruksjon.
 3. Den har et stort og komplisert instruksjonssett.
 4. Den har både nivå-1 og nivå-2 av hurtigbuffer på prosessorbrikken.
 5. Den har flere utføringsenheter ("functional units").

- m) Hva vil det si at en prosessor har støtte for 3-adresseinstruksjoner ("3-address instruction")?
1. Den pakker 3 instruksjoner sammen i en pakke med et felt som indikerer begrensninger i parallellitet.
 2. Den har spesielle hoppinstruksjoner som kan medføre hopp til 3 forskjellige adresser alt etter resultat av forrige instruksjon.
 3. Noe slikt eksisterer ikke; det maksimale er 2-adresseinstruksjoner.
 4. Hver instruksjon har 3 implisitte operander.
 5. 3 eksplisitte operander kan oppgis i instruksjonsformatet.
- n) Hvor mange forskjellige tall kan man maksimalt representere vha. 32 bit?
1. 2^{31} (= 2 147 483 648)
 2. $2^{31} - 1$ (= 2 147 483 647)
 3. 2^{32} (= 4 294 967 296)
 4. $2^{32} - 1$ (= 4 294 967 295)
 5. Dette avhenger av om tallet er et heltall eller et flyttall.
- o) Hva brukes predikatregistrene i IA-64 til?
1. De brukes i forbindelse med forgreningspredikering.
 2. De brukes i forbindelse med registeromdøping.
 3. De brukes i forbindelse med registervindu.
 4. De brukes i forbindelse med betinget utføring ("conditional execution").
 5. De brukes i forbindelse med samlebånd i programkode ("software pipelining").

Oppgave 2 – Instruksjonssettarkitektur – 20 %

- a) Flyttallsoperasjoner kan gi underflyt. Hva vil dette si?
- b) Forklar vha. et kodeksempel både sanne dataavhengigheter, ut- og anti-avhengigheter.
- c) Beskriv hva som skjer når et avbrudd ("interrupt") oppstår.
- d) Hvorfor kan man ikke uten videre bytte utføringsrekkefølgen på LOAD og STORE instruksjoner?

Oppgave 3 – Hurtigbuffer ("cache") – 20 % (5 % på a og b, 10 % på c)

- En hurtigbuffer har gjerne bare plass til 1/1000 av innholdet i hovedlageret. Forklar hvorfor det likevel er mulig med en treffrate på 95 %.
- Pentium 4 bruker "write through" mellom 1. og 2. nivå's hurtigbuffer og "write back" mellom 2. nivå's hurtigbuffer og hovedlager. Diskuter fordeler og ulemper med denne løsningen.
- I denne oppgaven skal du se på hvordan innholdet i en hurtigbuffer endrer seg under utføring av et kodesegment. Dette kodesegmentet er oppgitt under.

1. MOV R0, #0	Move	<i>Legger 0 inn i R0</i>
2. MOV R5, #40	Move	<i>Legger 40 (101 000 i binær) inn i R5</i>
3. LD R1, [R5]	Load	<i>Laster fra hovedlager til R1 (R5 har adressen)</i>
4. ADD R1, R1, R9	Add	<i>Adderer R9 til R1 ($R1 = R1 + R9$).</i>
5. ST [R5], R1	Store	<i>Lagrer fra R1 til hovedlager (R5 har adressen)</i>
6. ADD R5, R5, #4	Add	<i>Adderer 4 til R5</i>
7. SUB R0, R0, #1	Subtract	<i>Subtraherer 1 fra R0</i>
8. ADD R9, R9, #1	Add	<i>Adderer 1 til R9</i>

Gitt følgende opplysninger:

- Instruksjonsformatet er på 32 bit for alle instruksjoner.
- Load-instruksjonen henter 8 bit med data.
- Store-instruksjonen skriver 8 bit med data.
- Instruksjon 1 ligger i hovedlageret på adresse 0 med resten av instruksjonene lagret fortløpende.
- Hovedlager på 1 MB – adresselengden er dermed 20 bit (mao. byteadresserbart).
- Hurtigbufferet:
 - Direkte avbildning
 - 4 hurtigbufferlinjer
 - 2 ord per linje, hvert ord er 32 bit
 - Felles for både instruksjoner og data
 - "Write allocation" – hvis prosessoren skriver til hovedlager og data ikke finnes i hurtigbuffer, hentes data inn til hurtigbuffer først.

Anta at vi starter med tom hurtigbuffer. Vis hva som ligger i de fire hurtigbufferlinjene etter at hver enkelt av instruksjonene 1-8 er utført. Ta hensyn til både henting av instruksjoner og data og husk på å forklare hvordan du har tenkt!

Oppgave 4 – Mikroarkitektur – 20 % (5 % på a og b, 10 % på c)

- Hvilke to hoveddeler deles mikroarkitekturen inn i og hvilken oppgave har hver av disse?
- Forklar hva mikroinstruksjoner og ISA-instruksjoner er og hva som er forskjellen mellom dem.

- c) I denne oppgaven skal du simulere utføring av tre IJVM-instruksjoner i Mic-1 mikroarkitekturen. De tre instruksjonene er:

	Symbolisk form	Forklaring	Hexadesimal form
1	BIPUSH 4	Legg 4 øverst på stakken	0x10 0x04
2	BIPUSH 3	Legg 3 øverst på stakken	0x10 0x03
3	SWAP	Bytt de to øverste tallene på stakken	0x5F

Mic-1 utfører IJVM-instruksjoner ved hjelp av mikroprogram. Mikroprogram for BIPUSH og SWAP, samt hovedløkka (Main1) er gitt under.

Main1 PC = PC + 1; fetch; goto (MBR)

Bipush1 SP = MAR = SP + 1

Bipush2 PC = PC + 1; fetch

Bipush3 MDR = TOS = MBR; wr; goto Main1

Swap1 MAR = SP - 1; rd

Swap2 MAR = SP

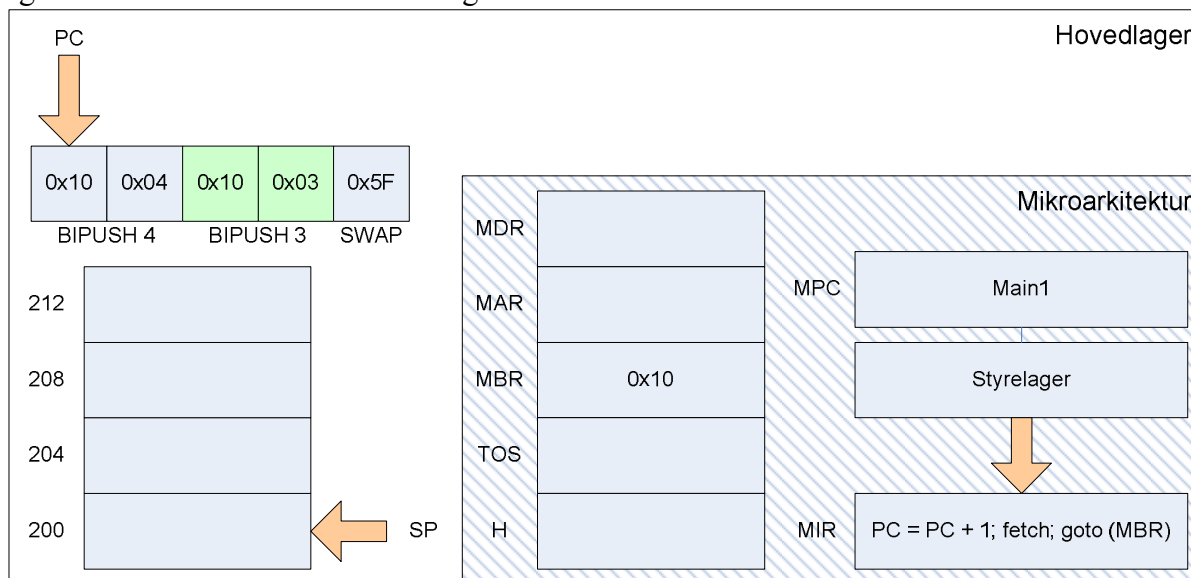
Swap3 H = MDR; wr

Swap4 MDR = TOS

Swap5 MAR = SP - 1; wr

Swap6 TOS = H; goto Main1

Starttilstanden er gitt under. Tomme ruter betyr at den nåværende verdien i registeret/hovedlageradressen er irrelevant for oppgaven. MIR ble lastet i slutten av forrige klokkesyklus og Main1 skal akkurat starte utføring.



Oppgaven går ut på å finne tilstanden etter at hver IJVM-instruksjon er ferdig (dvs. etter Bipush3, Bipush3 og Swap6). Bruk eget svarark til slutt i oppgavesettet.

Studentnr: _____ Studieprogram: _____ Arknr: _____ Antall ark: _____

Svarark for oppgave 1 – Flervalgsspørsmål ("multiple choice")

Dette arket skal brukes til å svare på oppgave 1. Arket skal rives av oppgavesettet og leveres inn sammen med resten av besvarelsen.

Dersom du finner flere alternativer som synes å passe, setter du kryss for det ene som passer best. For å unngå at gode tippere blir belønnet, vil et galt svar gi færre poeng enn om oppgaven forblir ubesvart.

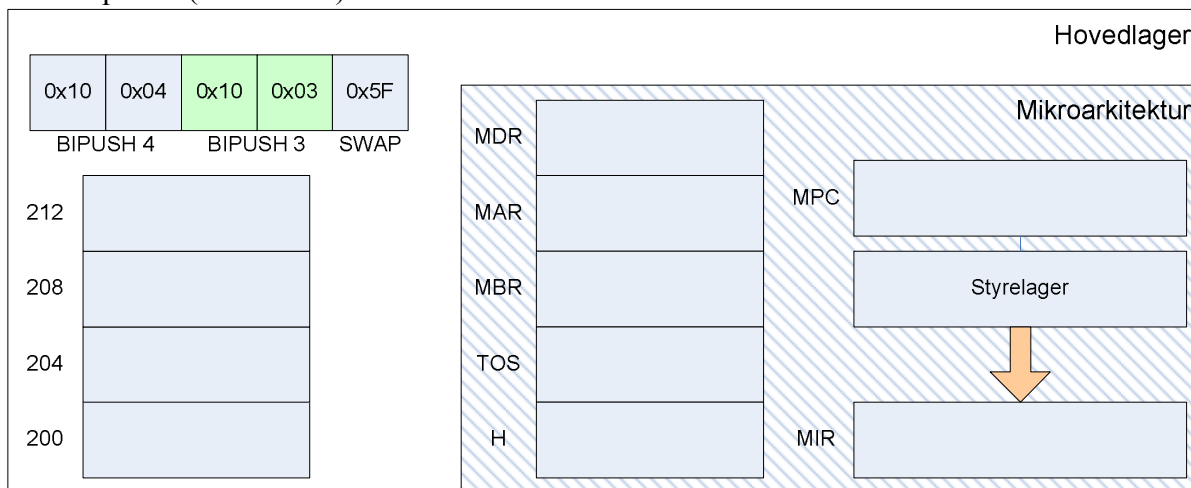
Alternativ→ Oppgave↓	1	2	3	4	5
a)					
b)					
c)					
d)					
e)					
f)					
g)					
h)					
i)					
j)					
k)					
l)					
m)					
n)					
o)					

Studentnr: _____ Studieprogram: _____ Arknr: _____ Antall ark: _____

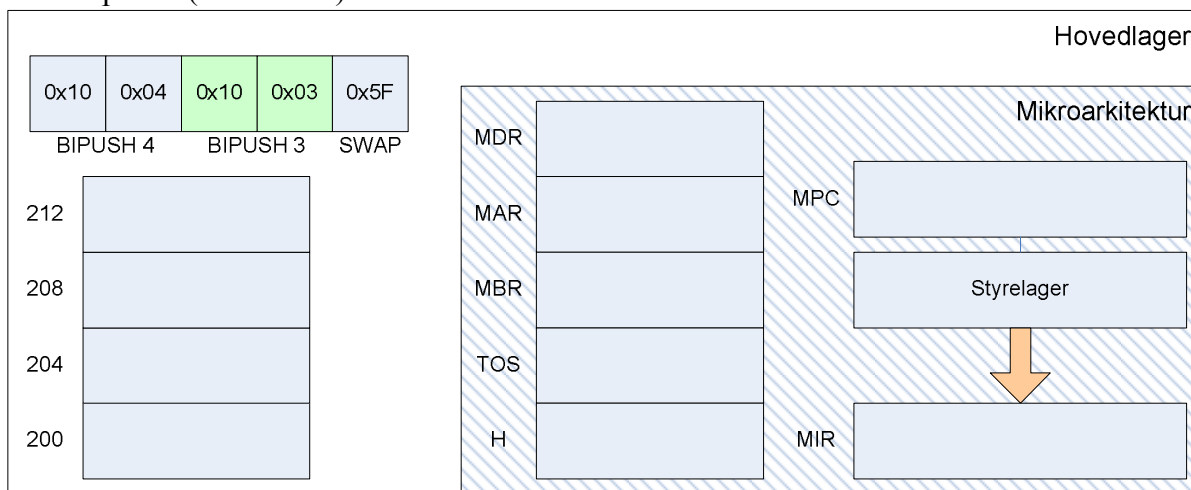
Svarark for oppgave 4c – Mikroarkitektur

Arket skal rives av oppgavesettet og leveres inn sammen med resten av besvarelsen.

Etter Bipush3 (BIPUSH 4):



Etter Bipush3 (BIPUSH 3):



Etter Swap6 (SWAP):

