



Norwegian University of Science and Technology  
Faculty of Information Technology, Mathematics and Electrical Engineering  
The Department of Computer and Information Science

# TDT4160

## DATAMASKINER GRUNNKURS

### EKSAMEN

10. DESEMBER, 2008, 09:00–13:00

**Kontakt under eksamen:**

Gunnar Tufte 73590356

**Tillatte hjelpemiddel:**

D.

Ingen trykte eller håndskrevne hjelpemidler tillatt.

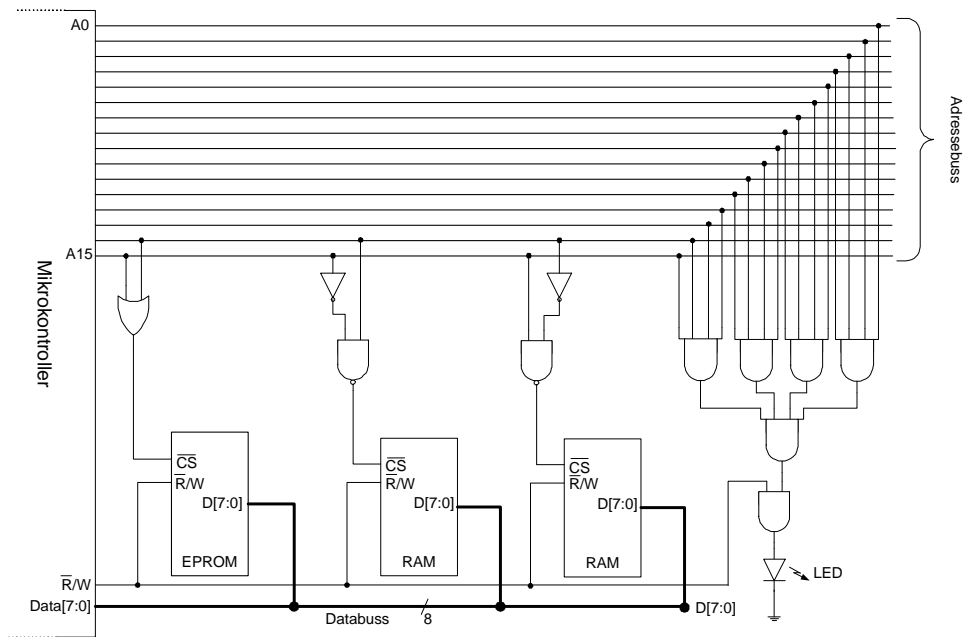
Bestemt, enkel kalkulator tillatt.

**Målform:**

Bokmål

## OPPGAVE 1: DIGITALT LOGISK NIVÅ (20% (10% PÅ A, 5% PÅ B OG C))

- a. I Figur 1 er EPROM, RAM og en lysdiode (LED) koblet til en felles buss. Finn adresseområdet for enhetene. EPROM og RAM har aktivt lavt (logisk "0") CS-signal (Chip Select).



Figur 1: Adressedekoding.

- b. Er det mulig å utvide systemet med 16kB (16384 bytes) RAM? Be-grunn svaret.
- c. Når lyser lysdioden (LED)?

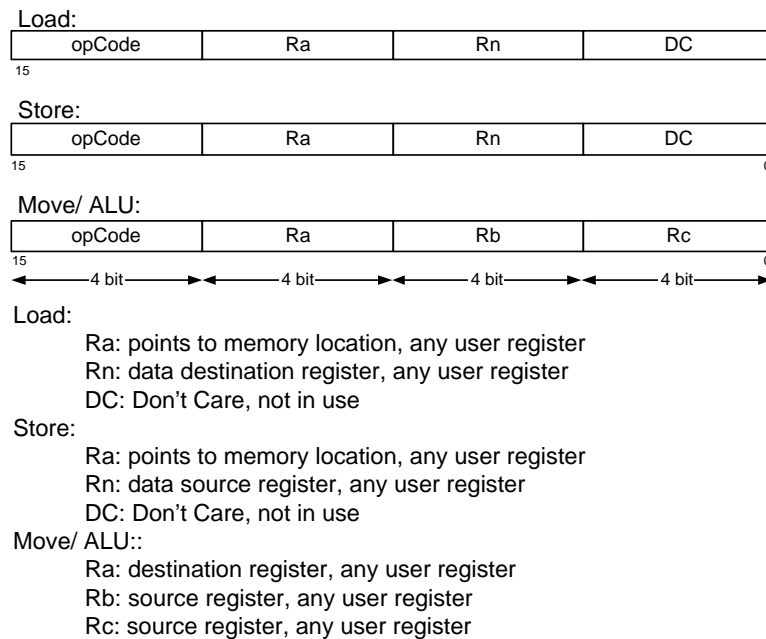
## OPPGAVE 2: MIKROARKITEKTUR OG MIKROINSTRUKSJONER (20% (5% PÅ A OG B; 10% PÅ C))

Bruk vedlagte diagram i figur 5, figur 6, figur 7 og figur 8 for IJVM til å løse oppgavene.

- a. Forklar funksjonen til registeret "MPC".
- b. Lag mikroinstruksjon(er) for følgende IJVM-operasjon: last register "OPC" med innholdet i register "H".  
Se vekk fra Addr- og J-felta i mikroinstruksjonsformatet. Angi korrekte bit for ALU, C, Mem og B gitt i Figur 6.
- c. Lag mikroinstruksjon(er) for følgende IJVM-operasjon:  
 $TOS = LV + (OPC + 1)$ .  
Se vekk fra Addr- og J-feltene i mikroinstruksjonsformatet. Angi korrekte bit for ALU, C, Mem og B gitt i Figur 6.

### OPPGAVE 3: INSTRUKSJONSSETT ARKITEKTUR (ISA)(20%)

I en tenkt 16-bit arkitektur har en bare tre typer instruksjoner, last (load), lagre (store) og flytt/ALU (Move/ALU). Figur 2 viser de tre instruksjonstypene.



Figur 2: Mulige instruksjonstyper.

- a. Ut fra tilgjengelig informasjon:
  - i) Hva er det maksimale antall instruksjoner denne maskinen kan ha? Begrunn svaret.
  - ii) Hva er det maksimale antall brukerregistre (user registers) denne maskinen kan ha? Begrunn svaret.
- b. Hvilken type arkitektur er dette instruksjonsformatet for?
- c. Hvis man endrer formatet for load/store til å bruke instruksjonsfeltet merket DC til å inneholde en indeks og bruker *Ra* som baseadresse, har man endret adresseringsmodi. Har dette noen innvirkning på ISA-definisjonen for denne tenkte maskinen? Begrunn svaret.
- d. Denne maskinen er mest sannsynlig en RISC-maskin. Hvorfor?

#### OPPGAVE 4: DATAMASKINER (20% (8% PÅ A OG C; 4% PÅ B))

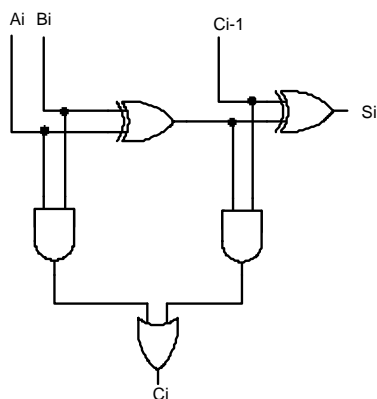
- a. Figur 9 og figur 10 i vedlegget viser forskjellige versjoner av IJVM-mikroarkitekturer. Hvilke grep er gjort og hvordan bidrar de til å øke ytelsen sammenliknet med den opprinnelige mikroarkitekturen i figur 5?
- b. Hvilken innvirkning har endringene i mikroarkitekturen gitt i spørsmål "a" på ISA-nivå?
- c. Bruk figur 5, figur 8 og figur 10: Ca. hvor mye kan klokkeperioden reduseres ved å endre mikroarkitekturen fra figur 5 til figur 10?

#### OPPGAVE 5: DIVERSE (20%)

Finn riktig svaralternativ for oppgavene. Korrekte svar gir 4% uttelling, feil svar gir -2% og vet ikke (ikke svar/flere svar) gir ingen uttelling.

- a. Hva ligg i *Addr*-feltet i *MIR*? Se figur 5
  - 1) Adressen til neste mikroinstruksjon i *control store*.
  - 2) Inneholder kun gyldig adresse hvis det skal gjøres et betinget hopp, aktivt J-bit.
  - 3) Til en hver tid en kopi av innholdet i *MBR*.
  - 4) *control store* start- og sluttadresse, for aktiv mikroinstruksjon.
- b. Hvilken påstand er korrekt for en en-brikke multiprosessorer (CMP)
  - 1) Er en "Array computer".
  - 2) Er type SIMD.
  - 3) Er av type homogen eller heterogen.
  - 4) Er av type MIMD og benytter alltid en "crossbar" for kommunikasjon mellom prosessorkjernene.

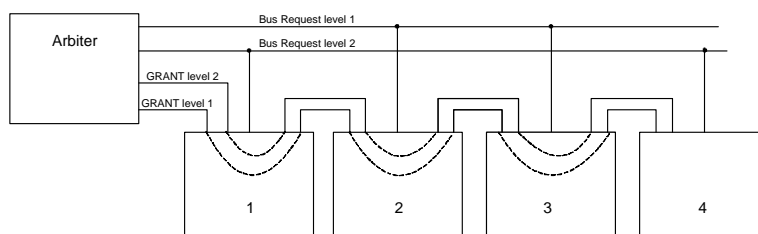
c. Hva er skissert i figur 3?



Figur 3: Mystisk dings.

- 1) Full-adder.
- 2) Halv-adder.
- 3) Statisk RAM-celle.
- 4) Multiplekser.

d. Hvordan er enhetene i figur 4 prioritert? Nivå 1 (level 1) er prioritert over nivå 2 (level 2). Rekkefølgen er gitt fra høyeste til laveste.



Figur 4: Sentralisert bussarbitrering.

- 1) 1, 2, 3, 4
- 2) 2, 3, 1, 4.
- 3) 1, 4, 2, 3.
- 4) 3, 2, 4, 1.

e. Under følger en rekke påstander om datamaskinkomponenter. Hvilken påstand er korrekt?

- 1) Asynkron bussoverføring må bruke en global klokke.
- 2) En ALU som kan utføre addisjon og invertere kan utføre subtraksjon.
- 3) EPROM og flash-minne har ekvivalent funksjonalitet.
- 4) EEPROM og flash-minne har ekvivalent funksjonalitet.

# IJVM vedlegg



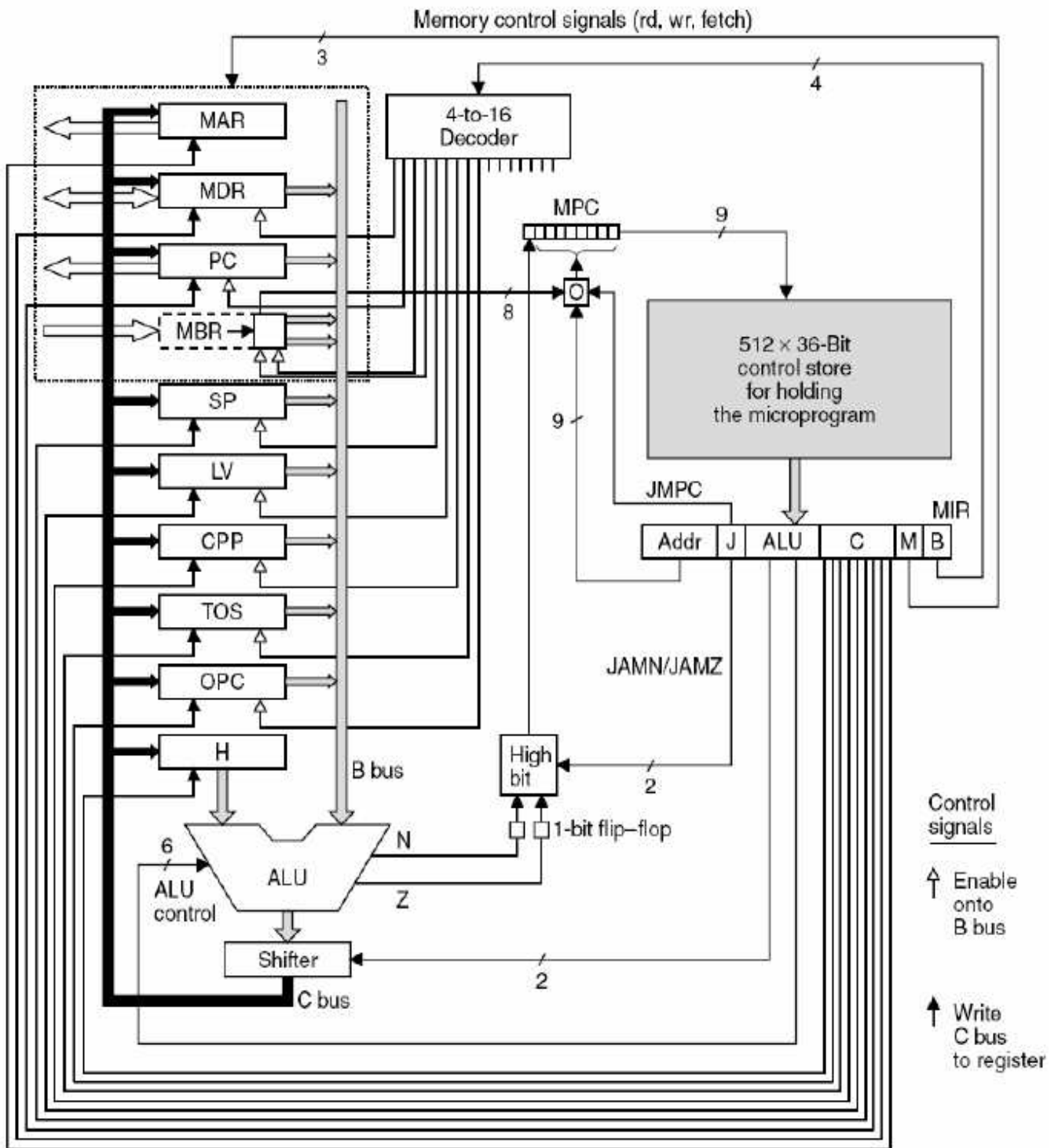
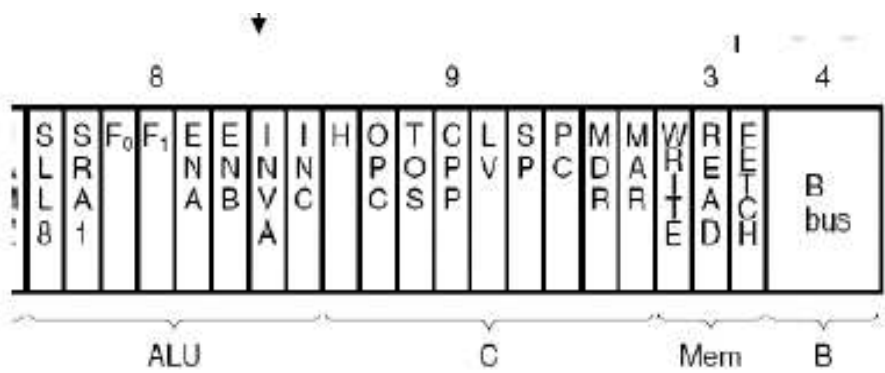


Figure 5: Blokkdiagram (IJVM).



**B bus registers**

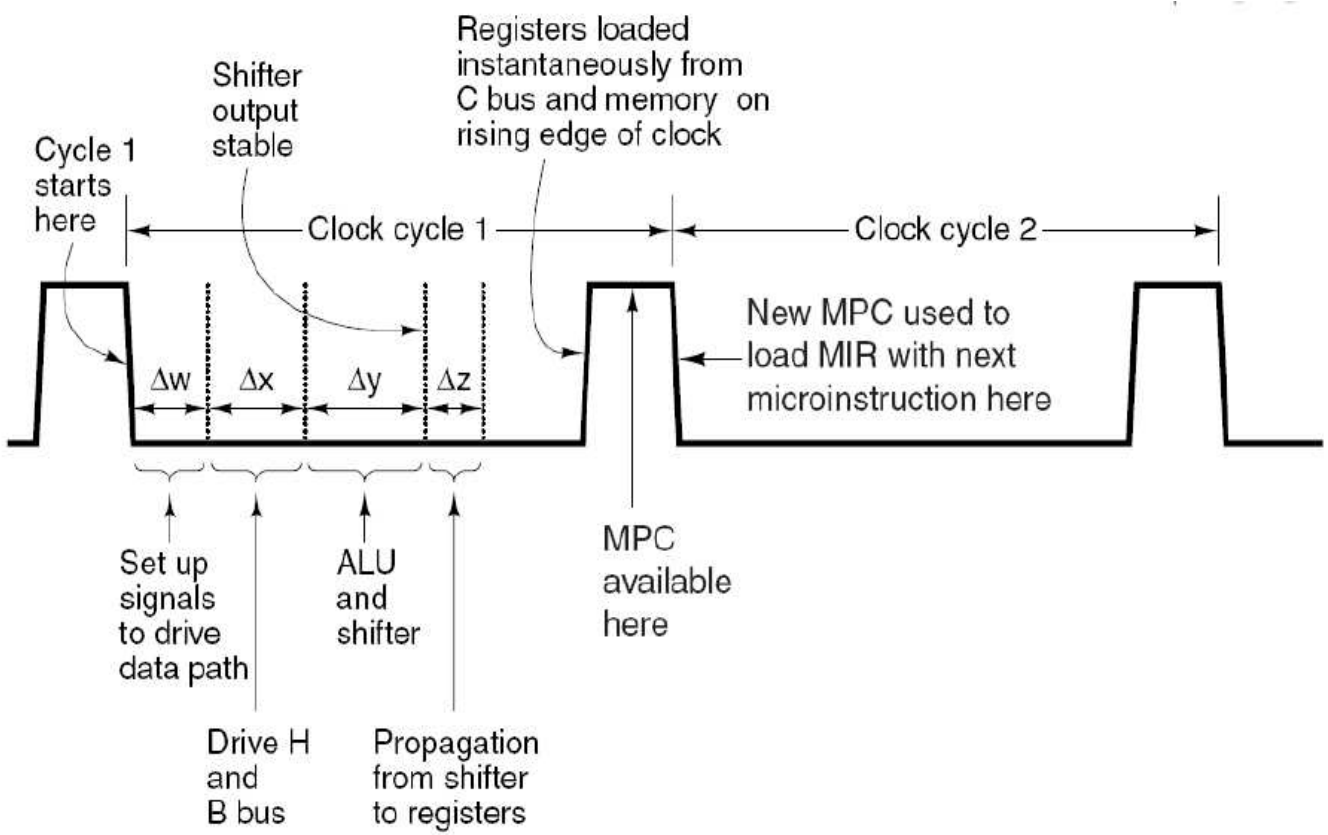
- 0 = MDR      5 = LV
- 1 = PC        6 = CPP
- 2 = MBR      7 = TOS
- 3 = MBRU    8 = OPC
- 4 = SP        9-15 none

Figur 6: Mikroinstruksjonsformat (IJVM).

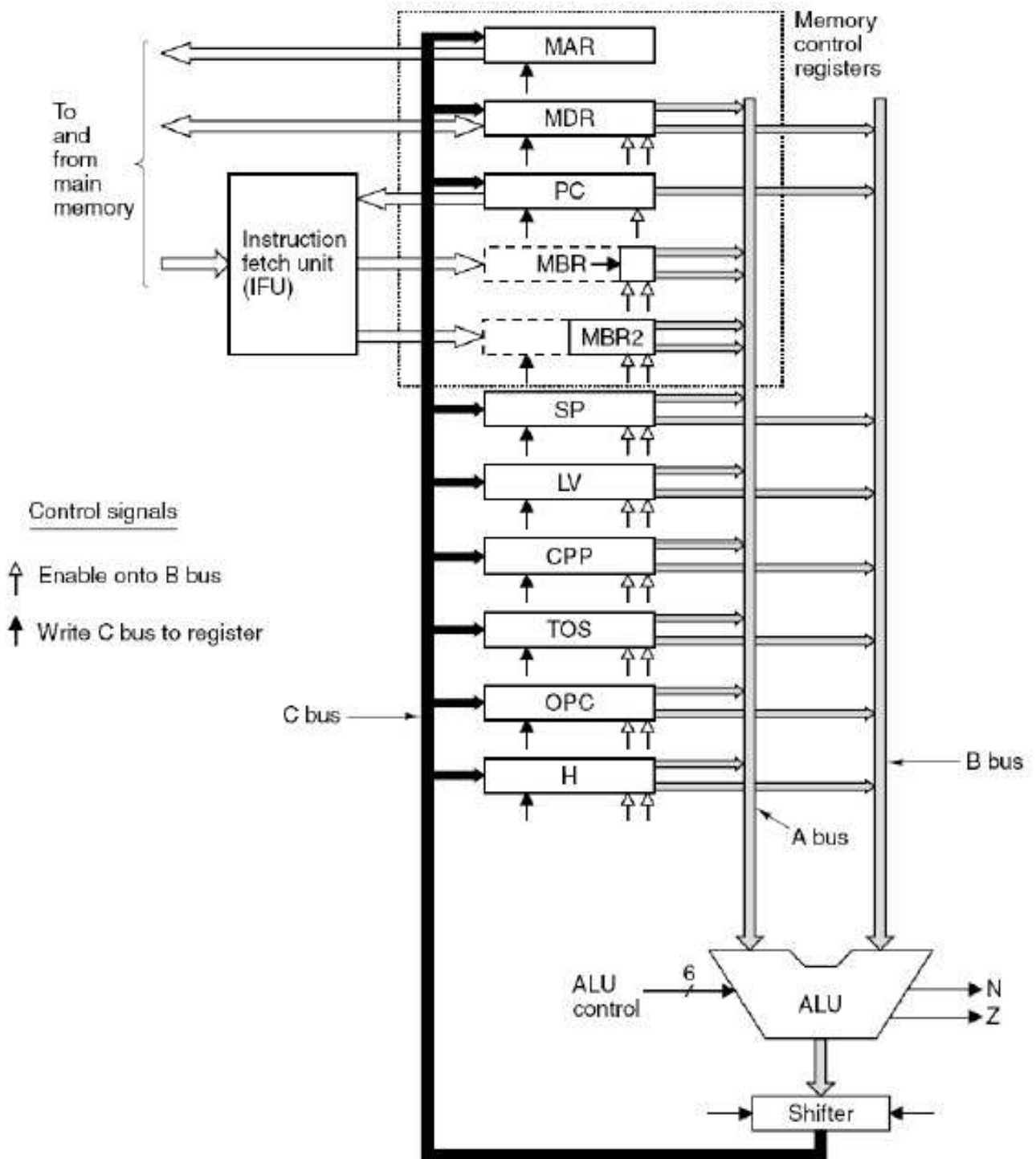
$F_0$	$F_1$	ENA	ENB	INVA	INC	Function
0	1	1	0	0	0	A
0	1	0	1	0	0	B
0	1	1	0	1	0	$\bar{A}$
1	0	1	1	0	0	$\bar{B}$
1	1	1	1	0	0	A + B
1	1	1	1	0	1	A + B + 1
1	1	1	0	0	1	A + 1
1	1	0	1	0	1	B + 1
1	1	1	1	1	1	B - A
1	1	0	1	1	0	B - 1
1	1	1	0	1	1	-A
0	0	1	1	0	0	A AND B
0	1	1	1	0	0	A OR B
0	1	0	0	0	0	0
1	1	0	0	0	1	1
1	1	0	0	1	0	-1

SLR1 SLL8 Function  
 0 0 No shift  
 0 1 Shift 8 bit left  
 1 0 Shift 1 bit right

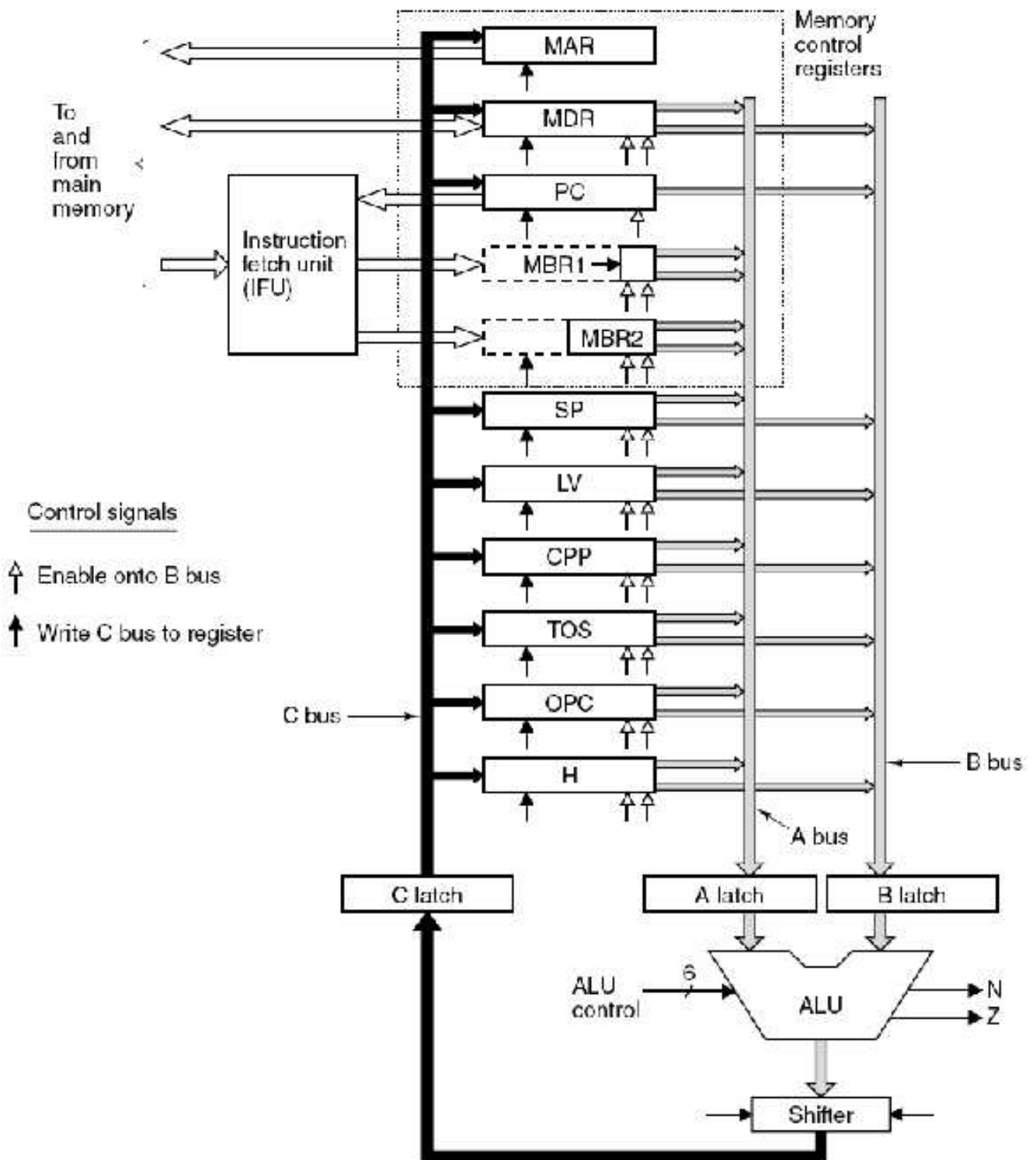
Figur 7: Funksjonstabell for ALU (IJVM).



Figur 8: Timing-diagram (IJVM).



Figur 9: Alternativ mikroarkitektur I.



Figur 10: Alternativ mikroarkitektur II.