



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

AUGUST, 2008, 09:00–13:00

Kontakt under eksamen:

Morten Hartmann 73593680

Tillatte hjelpemidler:

D.

Ingen trykte eller håndskrevne hjelpemiddel er tillat.

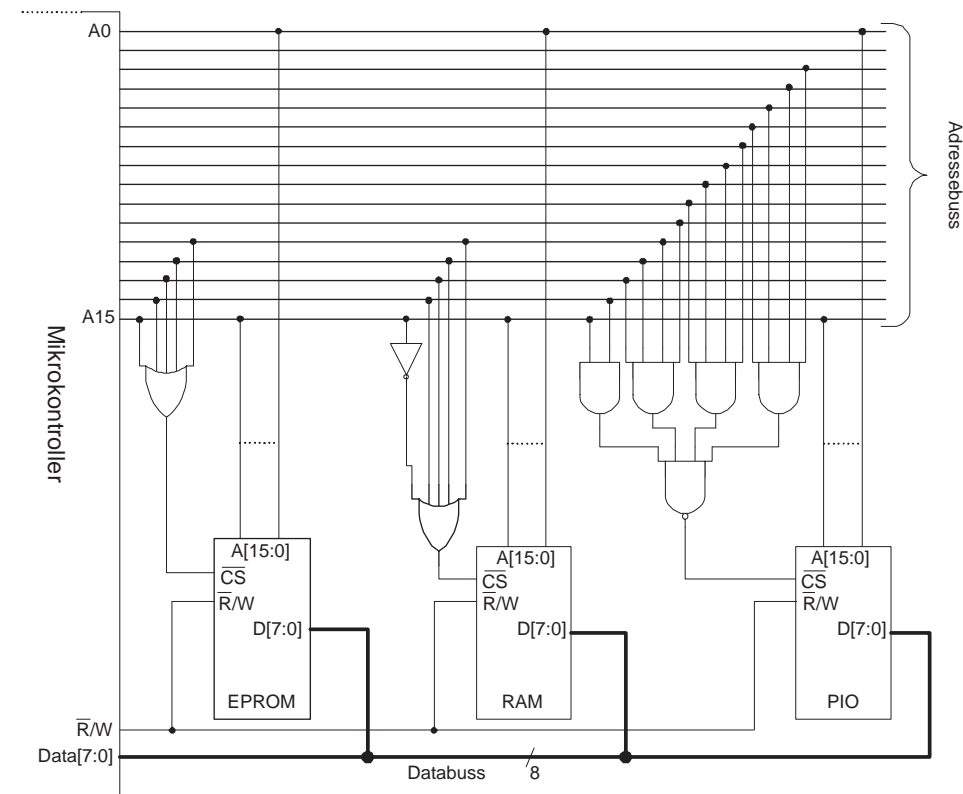
Enkel godkjent kalkulator er tillat.

Målform:

Bokmål

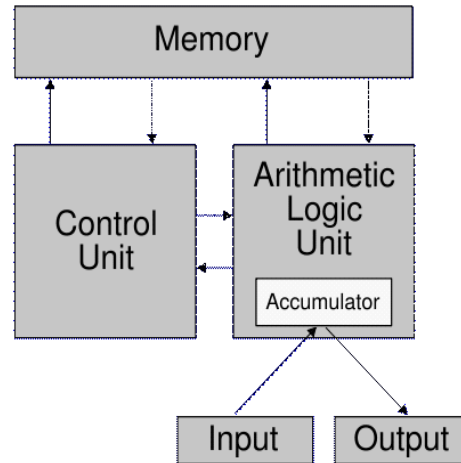
OPPGAVE 1: DATAMASKINER (30% (A OG B 2.5%, C OG D 10% OG E 5%))

Figur 1 viser en skisse av et system bestående av en mikrokontroller, EPROM, RAM og PIO. Kontroll signalet CS (Chip Select) er aktiv lav (logisk "0"). Data leses fra enhetene ved aktiv lav Read/Write (R/W) og skrives til enhetene ved aktivt høyt R/W-signal.



Figur 1: Adressedekoding.

- Hva vil typisk være funksjonen til EPROM, RAM og PIO i det skisserte mikrokontrollersystemet?
- Bruker det skisserte systemet dedikert adressebuss og databuss eller en multiplekset adresse/data buss? Begrunn svaret kort.
- Finne adresseområde for EPROM, RAM og PIO.
- Det er ønskelig å doble antall eksterne PIO-enheter. Er dette mulig uten å endre på adresseringen til EPROM og RAM? Viss mulig, hvordan kan en modifisere systemet for å realisere en slik dobling av antall PIO-enheter?



Figur 2: Von Neumann arkitektur.

- e. I Figur 2 er en maskin med såkalt "Von Neumann arkitektur" fra rundt 1945 skissert.

Forklar kort hvordan de forskjellige enhetene i Figur 1 forholder seg til von Neumann sin arkitektur.

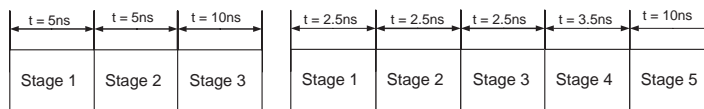
OPPGAVE 2: MIKROARKITEKTUR OG MIKROINSTRUKSJONER (30% (A OG B 2.5%, C 5% OG D OG E 10%))

Bruk vedlagte diagram og tabeller for IJVM til å løse oppgavene.

- Er det mulig å addere to operander som befinner seg i eksternt minne i en IJVM-mikroinstruksjon? Begrunn svaret kort.
- Hva er funksjonen til registrene MAR og MDR?
- Lag mikroinstruksjon(er) for følgende IJVM-operasjon: last register "H" med innholdet i register MDR - 1.
Se bort fra Addr- og J-feltene i mikroinstruksjonsformatet. Angi korrekte bit for ALU, C, Mem og B (se Figur 5).
- Lag mikroinstruksjon(er) for følgende IJVM-operasjon:
 $TOS = TOS + (MDR + 1)$.
Se bort fra Addr- og J-feltene i mikroinstruksjonsformatet. Angi korrekte bit for ALU, C, Mem og B (se Figur 5).
- Kan resultatet av å utføre mikroinstruksjonene i oppgave c og d medføre at N og/eller Z flagg aktiveres? Begrunn svaret kort.

OPPGAVE 3: FORBEDRE YTELSEN (20%)

- Gi en kort oppsummering av fordelene med en-brikke multiprosesorer (CMP).
- Gi en kort oppsummering av eventuelle utfordringer med en-brikke multiprosesorer.
- Anta at en prosessor har et samlebånd med tre trinn som vist i Figur 3(a). I Figur 3(b) er et fem-trinns samlebånd vist. Hva skjer med ytelsen viss en endrer antall trinn i samlebåndet fra tre til fem? Begrunn svaret kort.



(a) Samlebånd med tre trinn. (b) Samlebånd med fem trinn.

Figur 3: Samlebånd med angitt forsinkelse.

- Superskalare prosessorer øker ytelsen ved å duplisere CPU-komponenter (økt instruksjonsnivåparallellitet). Det er en dyr løsning. Hva kan gjøres for å redusere kostnaden og samtidig bevare prinsippet bak superskalare prosessorer?

OPPGAVE 4: DIVERSE (20%)

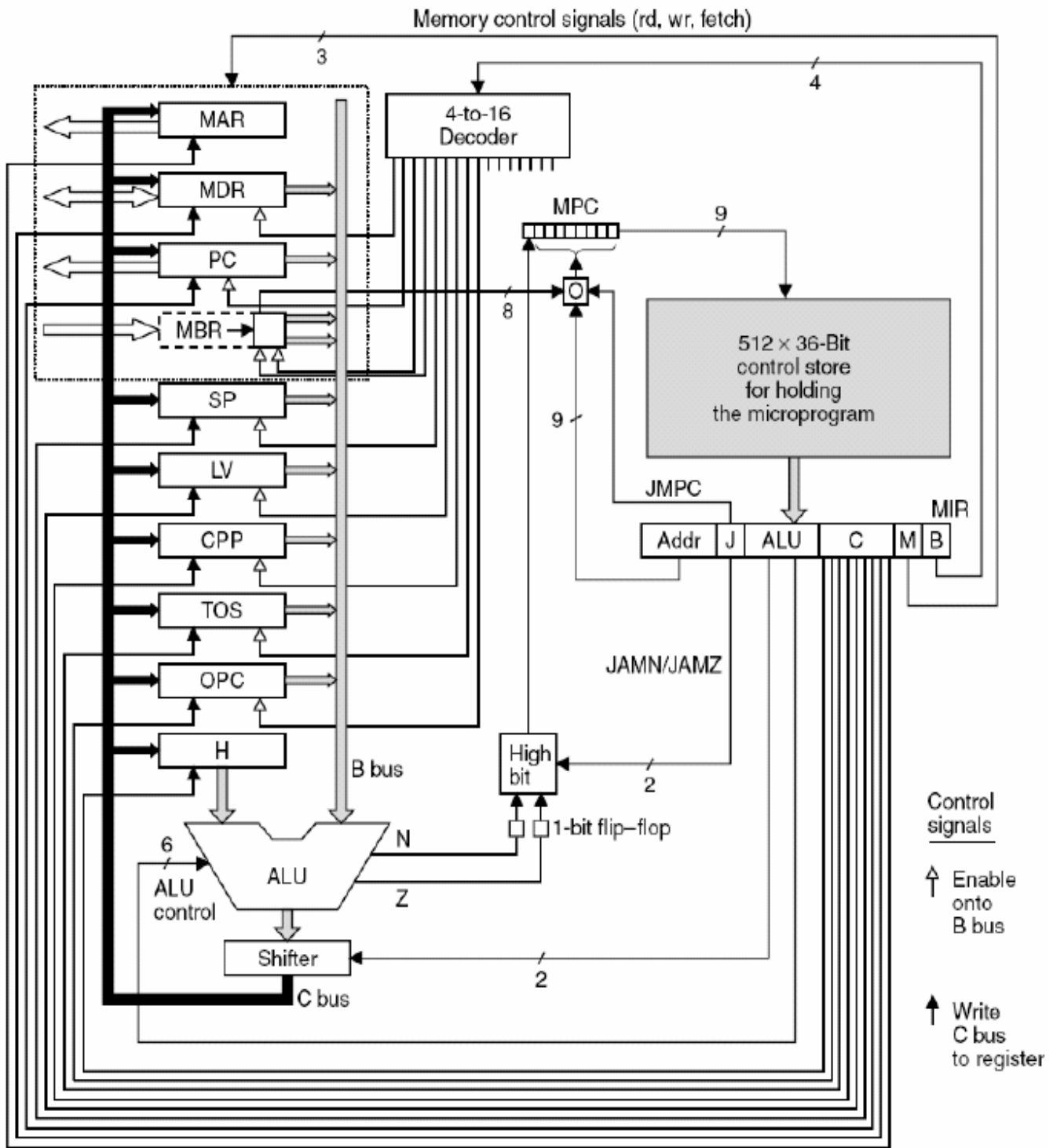
Ta stilling til påstandene. Korrekte svar gir 4% uttelling, feil svar gir -2% og vet ikke (ikke svar) gir 0%.

- a. Hvilken påstand er korrekt for en prosessor som benytter multiplekset adresse/data buss?
 - 1 Krever mindre I/O-pinner enn en løsning med dedikerte busser.
 - 2 Kan adressere mindre data enn en prosessor med dedikerte busser.
 - 3 Kan ikke benytte samlebånd siden instruksjoner og data benytter felles buss for henting av data og instruksjoner
 - 4 Antall bit til adressering er lik antall bit for instruksjoner/data.
- b. Hvilken påstand er korrekt for instruksjoner som benytter "immediate"adressering .
 - 1 Operanden kan kun angir adresse til CPU-en sit nærmeste ("immediate") lagernivå (register).
 - 2 Operanden kan kun adressere data som ligger i hurtigbufferet.
 - 3 Operanden kan kun brukes til konstanter, verdien er gitt ved kompilering.
 - 4 Immediate adressering er effektivt for globale variabler.
- c. Hvilken påstand hører **ikke** til i nivået for instruksjonsetarkitektur?
 - 1 Første nivå tilgjengelig for bruker.
 - 2 Opprinnelig det eneste nivået.
 - 3 Inkluderer støtte for flere program og virtuelt minne.
 - 4 Grense mellom maskinvare og programvare.
- d. Hvilken påstand om parallelle maskiner er **ikke** korrekt?
 - 1 Forskjellen på SIMD array-maskiner og MIMD multiprosessor-maskiner er kun gitt av hvordan minne er organisert.
 - 2 RISC prosessorer kan være superscalare.
 - 3 CISC prosessorer kan være superscalare.
 - 4 I en MIMD multiprosessormaskin med delt hovedlager er busen en flaskehals.

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

- 1 Three-state buffer har to tilstander: tilkople eller høy impedans.
- 2 Three-state buffer har tre tilstander: logisk "1", logisk "0" eller høy impedans (z).
- 3 I en EPROM må alt innhold slettes for å kunne oppdatere innholde.
- 4 PCI Express benytter punkt-til-punkt kommunikasjon.

IJVM vedlegg



Figur 4: Blokkdiagram (IJVM).

ANSWER KEY FOR THE EXAM

OPPGAVE 1: DATAMASKINER (30% (A OG B 2.5%, C OG D 10% OG E 5%))

Figur 1 viser en skisse av et system bestående av en mikrokontroller, EPROM, RAM og PIO. Kontroll signalet CS (Chip Select) er aktiv lav (logisk "0"). Data leses fra enhetene ved aktiv lav Read/Write (R/W) og skrives til enhetene ved aktivt høyt R/W-signal.

- a. Hva vil typisk være funksjonen til EPROM, RAM og PIO i det skisserte mikrokontrollersystemet?

Answer: EPROM: lagre programmet (programminne).

RAM: lagre data (dataminne).

PIO: I/O til eksterne parallelle enheter.

- b. Bruker det skisserte systemet dedikert adressebuss og databuss eller en multiplekset adresse/data buss? Begrunn svaret kort.

Answer: Dedikerte busser, det er ingen adresse/data latch i figuren.

- c. Finn adresseområde for EPROM, RAM og PIO.

Answer: EPROM: hex(0000) - hex(07FF)

RAM: hex(8000) - hex (87FF)

PIO: hex(FFFC) - hex(FFFF)

- d. Det er ønskelig å doble antall eksterne PIO-enheter. Er dette mulig uten å endre på adresseringen til EPROM og RAM? Viss mulig, hvordan kan en modifisere systemet for å realisere en slik dobling av antall PIO-enheter?

Answer: Det er mulig. Det er ubrukt adresserom fra adresse hex(8800) til hex(FFFB).

Enkleste løsning er å fjerne linjen A2 fra PIO adressedekoding. Utvider da PIO-adresserom til hex(FFF8) - hex(FFFF). Øker da fra 4 til 8 PIO-enheter.

- e. I Figur 2 er en maskin med såkalt "Von Neumann arkitektur" fra rundt 1945 skissert.

Forklar kort hvordan de forskjellige enhetene i Figur 1 forholder seg til von Neumann sin arkitektur.

Answer: Memory: EPROM og RAM.

Input/output: PIO.

CTRL og ALU: mikrokontroller.

OPPGAVE 2: MIKROARKITEKTUR OG MIKROINSTRUKSJONER (30% (A OG B 2.5%, C 5% OG D OG E 10%))

Bruk vedlagte diagram og tabeller for IJVM til å løse oppgavene.

- a. Er det mulig å addere to operander som befinner seg i eksternt minne i en IJVM-mikroinstruksjon? Begrunn svaret kort.

Answer: Nei. En av operandene må over i H registeret.

- b. Hva er funksjonen til registrene MAR og MDR?

Answer: MAR: inneholder adresse til eksternt minne.

MDR: inneholder data til/fra eksternt minne.

- c. Lag mikroinstruksjon(er) for følgende IJVM-operasjon: last register "H" med innholdet i register MDR - 1.

Se bort fra Addr- og J-feltene i mikroinstruksjonsformatet. Angi korrekte bit for ALU, C, Mem og B (se Figur 5).

Answer: ALU: 110110 (B - 1) C: 10000000 (H) Mem: 000 (ingen mem opprasjon) B: 0000 (0 MDR)

- d. Lag mikroinstruksjon(er) for følgende IJVM-operasjon:

$TOS = TOS + (MDR + 1)$.

Se bort fra Addr- og J-feltene i mikroinstruksjonsformatet. Angi korrekte bit for ALU, C, Mem og B (se Figur 5).

Answer: 1: Laste TOS inn i H ALU: 010100 (B) C: 10000000 (H) Mem: 000 (ingen mem opprasjon) B: 0111 (7 TOS)

2: addere H + MDR skriv til TOS ALU: 111101 (A+B+1) C: 00100000 (TOS) Mem: 000 (ingen mem opprasjon) B: 1000 (0 MDR)

eller MDR kan lastast i H i fyrste opperasjon og $TOS + H$ i andre operasjon.

- e. Kan resultatet av å utføre mikroinstruksjonene i oppgave c og d medføre at N og/eller Z flagg aktiveres? Begrunn svaret kort.

Answer: Z flagg aktivert viss resultat er 0.

N flagg aktivert viss resultat er negativt

OPPGAVE 3: FORBEDRE YTELSEN (20%)

- a. Gi en kort oppsummering av fordelene med en-brikke multiprosessorer (CMP).

- Answer:**
- 1 Tilgjengeleg ILP (Instruction Level Parallelism) i instruksjonsstraumen er avgrensa
 - 2 Ein-kjerneprosessorar har ofte høgt effektforbruk.
 - 3 Ein moderne ein-kjerneprosessor er svært kompleks og derfor dyr å utvikle.

b. Gi en kort oppsummering av eventuelle utfordringer med en-brikke multiprosessorer.

Answer: Krever støtte i programmvare som kan utnytte flere prosessor kjerner.

Se forelesings notater for utfyllande informasjon.

c. Anta at en prosessor har et samlebånd med tre trinn som vist i Figur 3(a). I Figur 3(b) er et fem-trinns samlebånd vist. Hva skjer med ytelsen viss en endrer antall trinn i samlebåndet fra tre til fem? Begrunn svaret kort.

Answer: Klokka bestemmes av tregeste trinn (10ns) det er ikkje mulig å auke ytelsen ved å auke klokke frekvensen. Fem trinn vill bruke lengre tid på å tomme og fylles.

d. Superskalare prosessorer øker ytelsen ved å duplisere CPU-komponenter (økt instruksjonsnivåparallelitet). Det er en dyr løsning. Hva kan gøres for å redusere kostnaden og samtidig bevare prinsippet bak superskalare prosessorer?

Answer: Duplisere kunn mest tidkrevende trinn (feks utførende som i Pentium II).

OPPGAVE 4: DIVERSE (20%)

Ta stilling til påstandene. Korrekte svar gir 4% uttelling, feil svar gir -2% og vet ikke (ikke svar) gir 0%.

a. Hvilken påstand er korrekt for en prosessor som benytter multiplexset adresse/data buss?

- 1 Krever mindre I/O-pinner enn en løsning med dedikerte busser.
- 2 Kan adressere mindre data enn en prosessor med dedikerte busser.
- 3 Kan ikke benytte samleband siden instruksjoner og data benytter felles buss for henting av data og instruksjoner
- 4 Antall bit til adressering er lik antall bit for instruksjoner/data.

Answer: 1

b. Hvilken påstand er korrekt for instruksjoner som benytter "immediate" adressering .

- 1 Operanden kan kun angir adresse til CPU-en sit nærmeste ("immediate") lagernivå (register).
- 2 Operanden kan kun adressere data som ligger i hurtigbufferet.
- 3 Operanden kan kun brukes til konstanter, verdien er gitt ved kompilering.
- 4 Immediate adressering er effektivt for globale variabler.

Answer: 3

c. Hvilken påstand hører **ikke** til i nivået for instruksjonsetarkitektur?

- 1 Første nivå tilgjengelig for bruker.
- 2 Opprinnelig det eneste nivået.
- 3 Inkluderer støtte for flere program og virtuelt minne.
- 4 Grense mellom maskinvare og programvare.

Answer: 3

d. Hvilken påstand om parallelle maskiner er **ikke** korrekt?

- 1 Forskjellen på SIMD array-maskiner og MIMD multiprosessor-maskiner er kun gitt av hvordan minne er organisert.
- 2 RISC prosessorer kan være superscalare.

- 3 CISC prosessorer kan være superscalare.
- 4 I en MIMD multiprosessormaskin med delt hovedlager er bus-
sen en flaskehals.

Answer: 1

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

- 1 Three-state buffer har to tilstander: tilkople eller høy impedans.
- 2 Three-state buffer har tre tilstander: logisk "1", logisk "0" eller høy impedans (z).
- 3 I en EPROM må alt innhold slettes for å kunne oppdatere innholde.
- 4 PCI Express benytter punkt-til-punkt kommunikasjon.

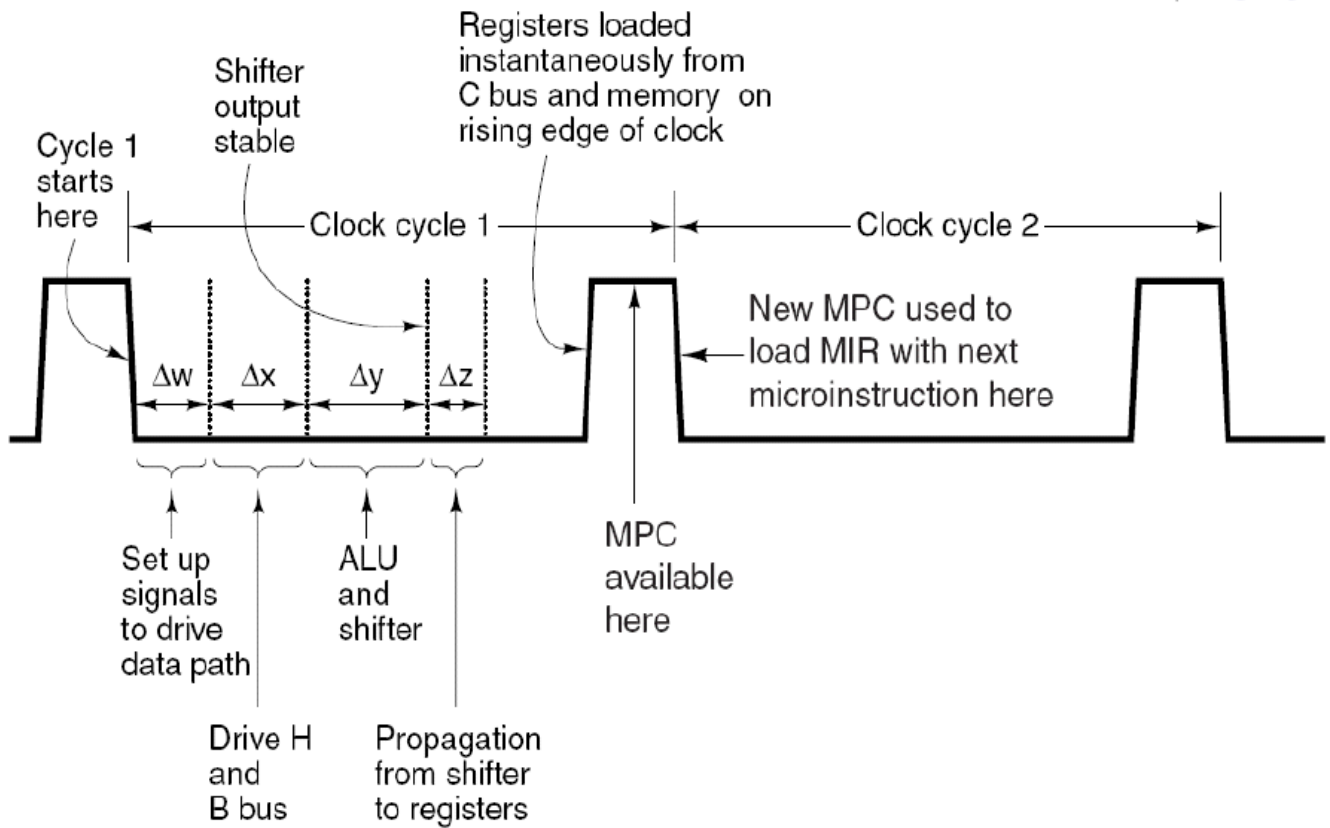
Answer: 2

IJVM vedlegg

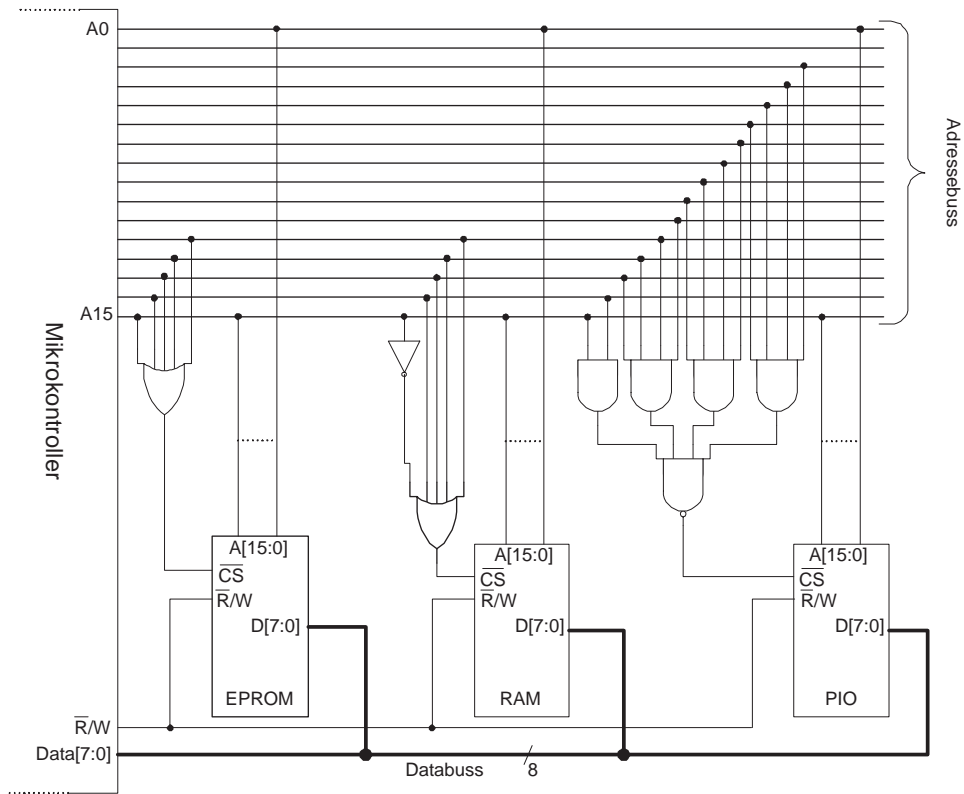
F_0	F_1	ENA	ENB	INVA	INC	Function
0	1	1	0	0	0	A
0	1	0	1	0	0	B
0	1	1	0	1	0	\bar{A}
1	0	1	1	0	0	\bar{B}
1	1	1	1	0	0	A + B
1	1	1	1	0	1	A + B + 1
1	1	1	0	0	1	A + 1
1	1	0	1	0	1	B + 1
1	1	1	1	1	1	B - A
1	1	0	1	1	0	B - 1
1	1	1	0	1	1	-A
0	0	1	1	0	0	A AND B
0	1	1	1	0	0	A OR B
0	1	0	0	0	0	0
1	1	0	0	0	1	1
1	1	0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

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

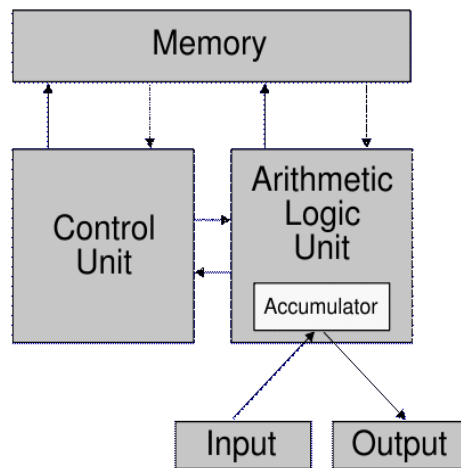
Figur 6: Funksjonstabell for ALU (IJVM).



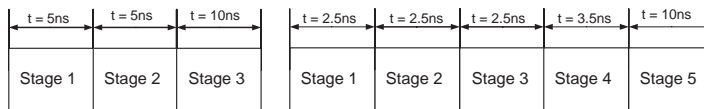
Figur 7: Timingdiagram (IJVM).



Figur 8: Adressedekoding.



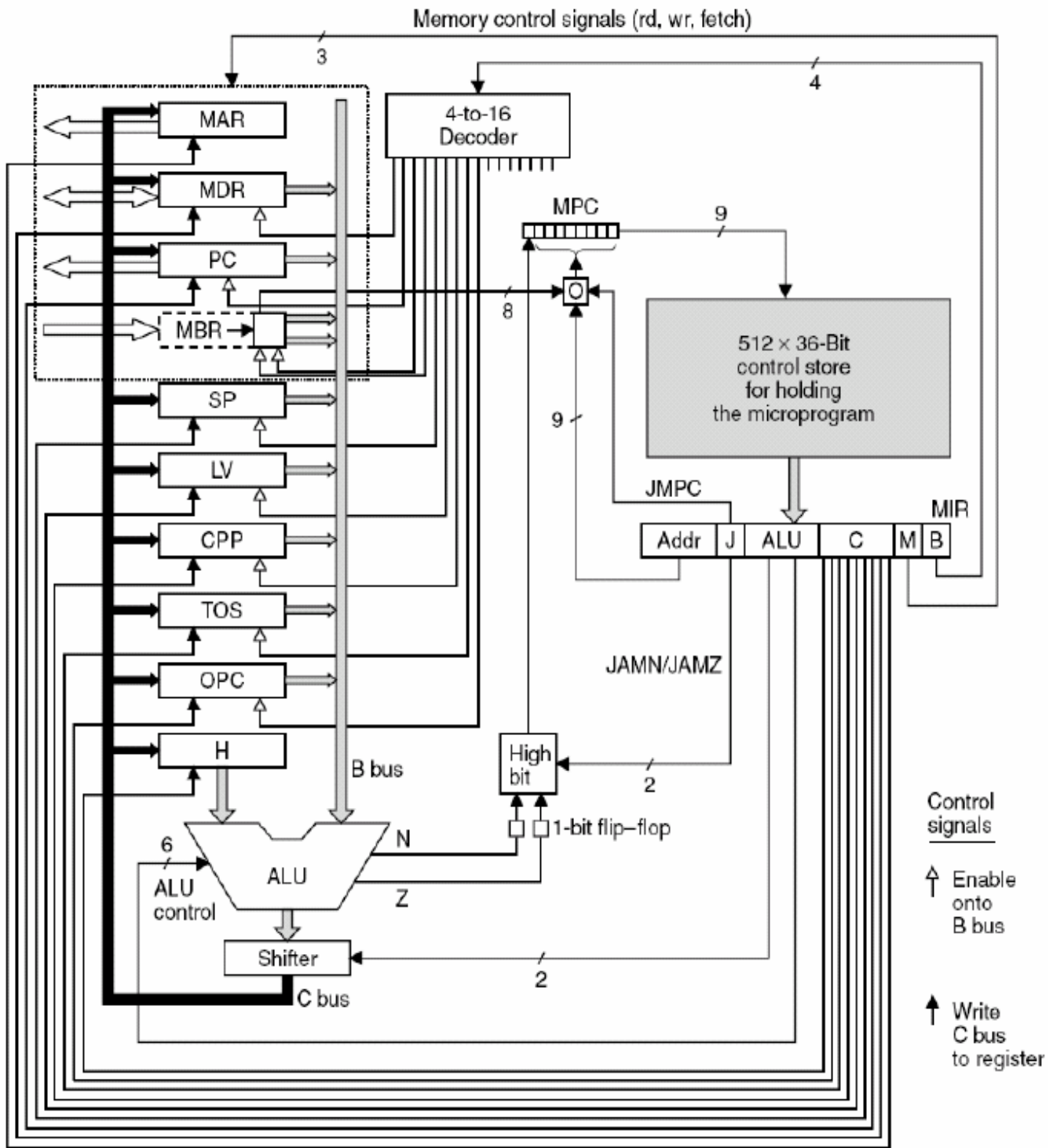
Figur 9: Von Neumann arkitektur.



(a) Samlebånd med tre trinn.

(b) Samlebånd med fem trinn.

Figur 10: Samlebånd med angitt forsinkelse.

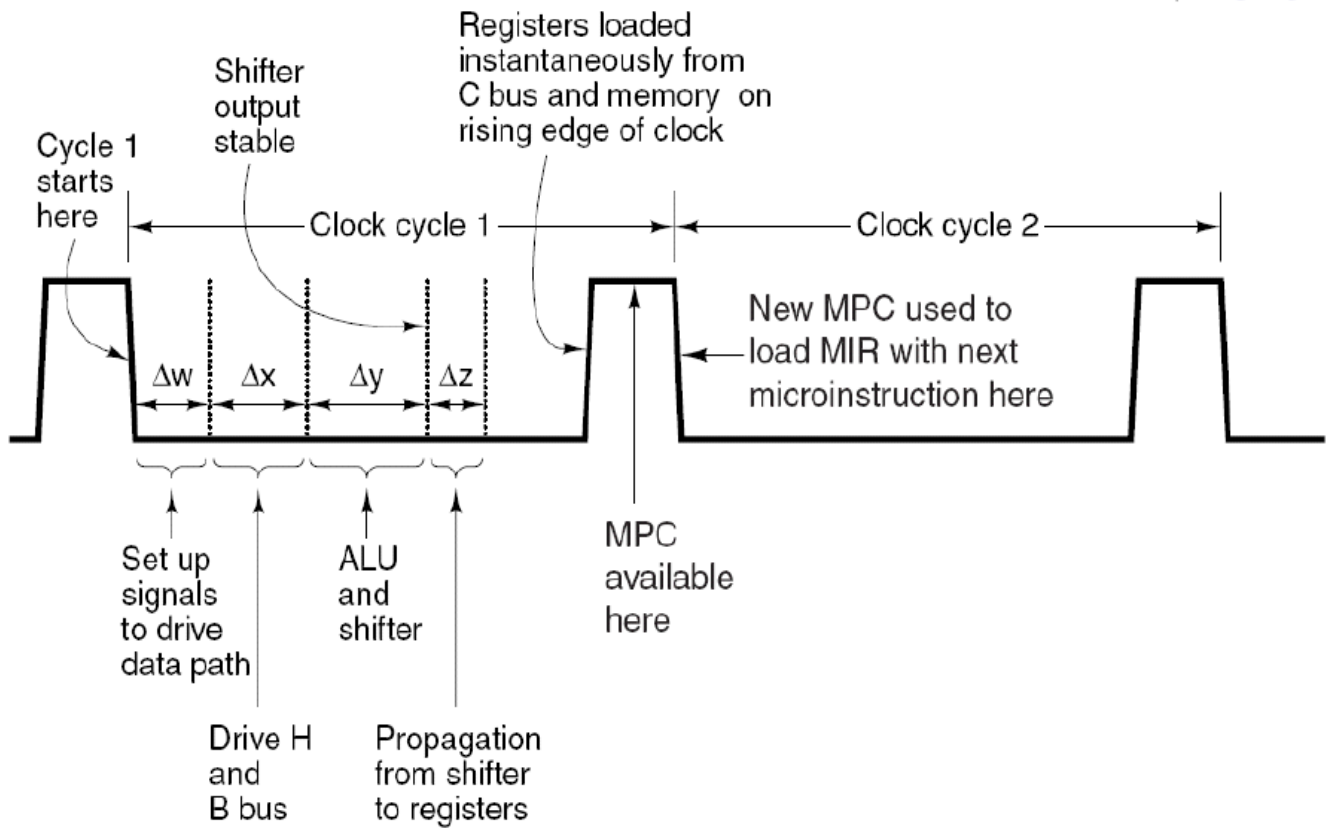


Figur 11: Blokkdiagram (IJVM).

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 13: Funksjonstabell for ALU (IJVM).



Figur 14: Timingdiagram (IJVM).