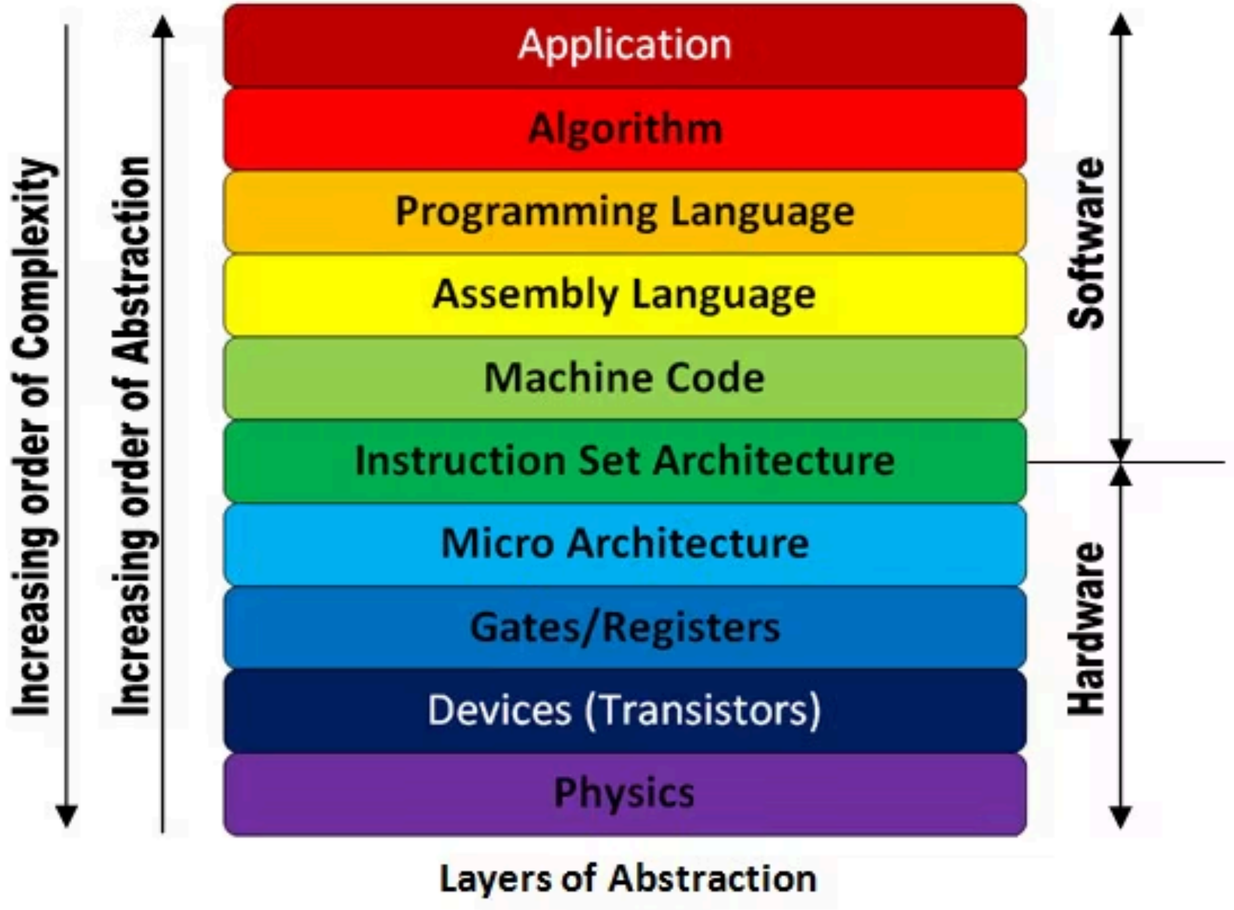
IN1020 - Hastighetsøkning

Omid Mirmotahari og Yngvar Berg

Menti

0

Menti

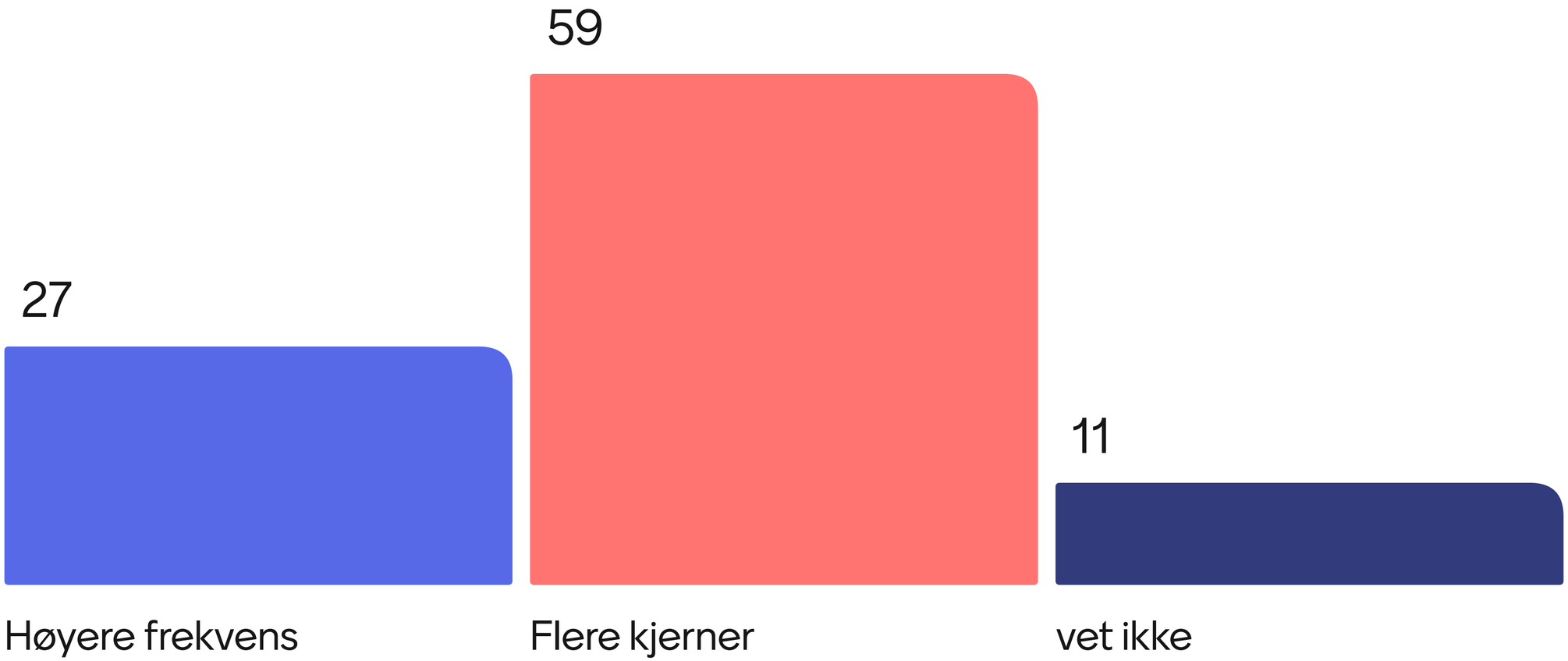


Abstraksjonsnivåer for datamaskin

0

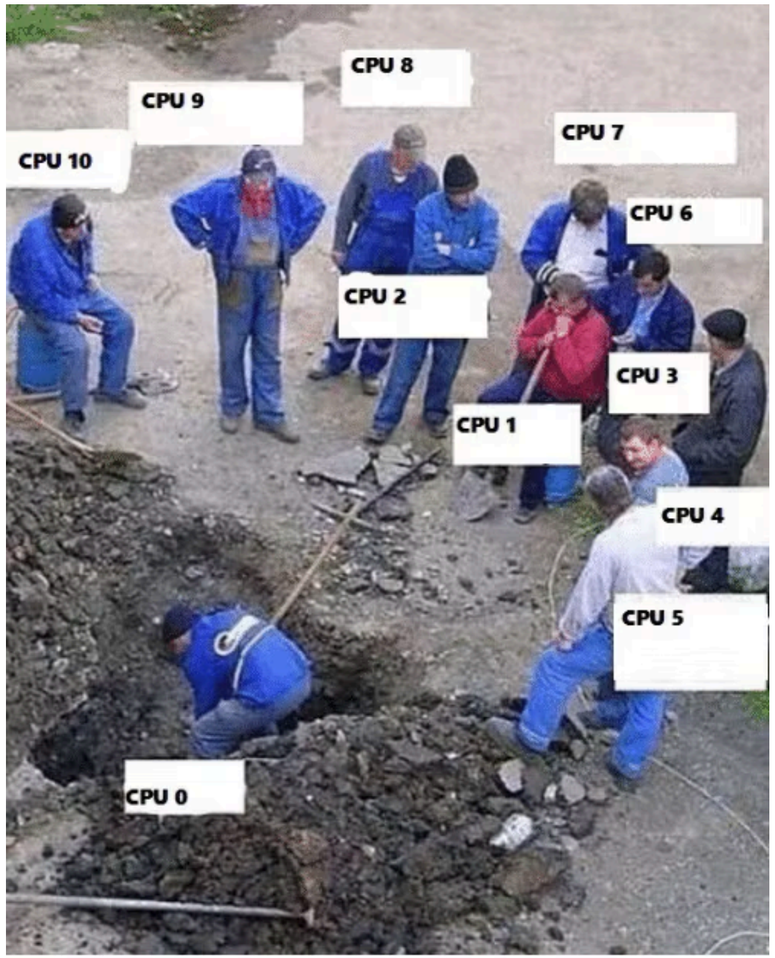
Menti

Hva foretrekker du?



97/97

Menti



Multikjerne

0

Dagens forelesning

- HUSK! Wanna-be læreboken er lagt ut
- Hvordan virker en pipeline
- Lengden av en klokkesykel
- Hva er en BUS?
- Ulike typer av minne
- Utlesing av minne

Menti
0

Menti

Individuell oppgave: Hva gjør instruksjonen 199 i LMC her?

```
901
399
901
199
902
000
```



69% adderer



17% leser inn en verdi



7% hopper til en angitt adresse



7% vet ikke

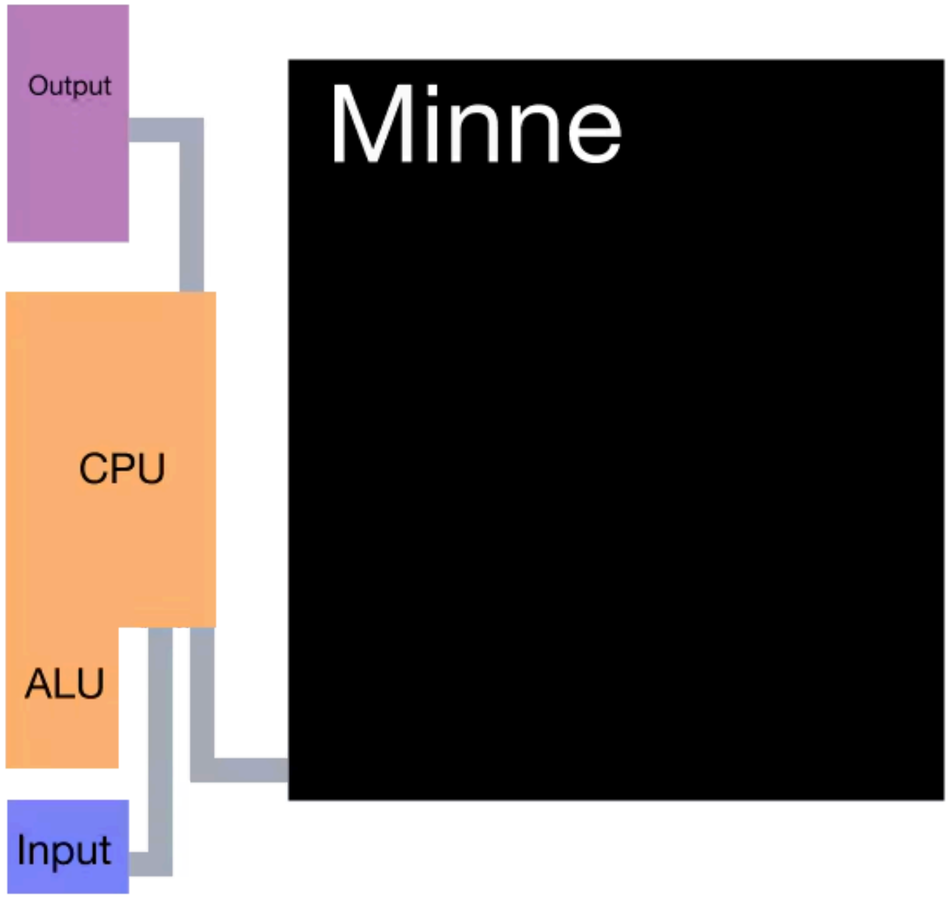
102/102

Utførelse av addisjon (instruksjon 199)

- Hent instruksjonen og øk programtelleren (FETCH)
- Splitt instruksjonen og hent fra minne (DECODE)
- Utfør instruksjonen (EXECUTE)
- Lagre data (WRITE-BACK)

Menti
0

Menti



LMC - forenklet

0

Menti

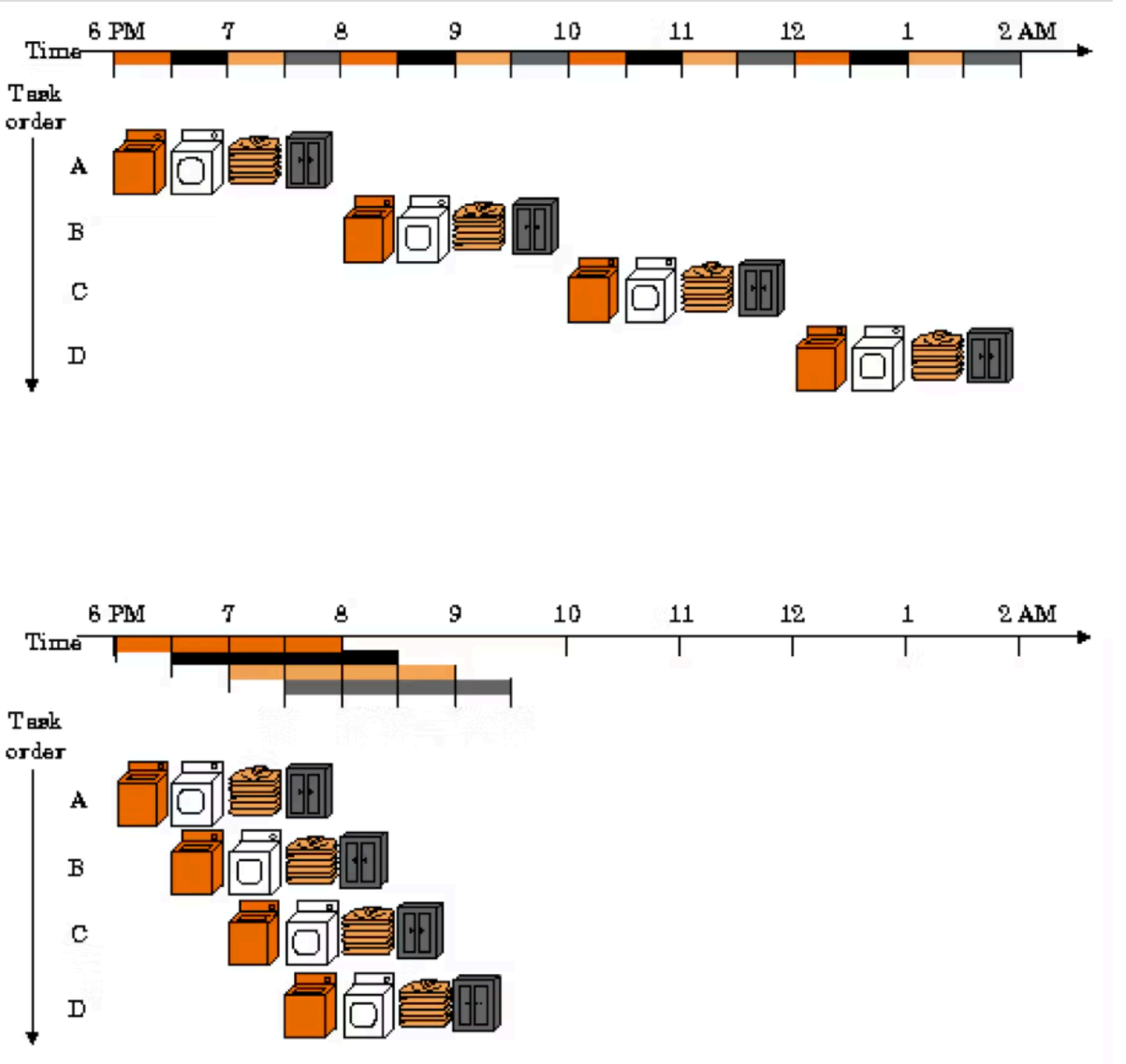
Pipeline

- Samlebåndsprinsipp
- Dele opp instruksjonen i uavhengige deler
- Neste instruksjon settes i gang før forrige instruksjon er helt ferdig
- NB!! Hver instruksjon tar like lang tid å utføre

0

Analogi: Klesvask

Menti
0


6 PM
7
8
9
10
11
12
1
2 AM
Time
Task order
A
B
C
D
6 PM
7
8
9
10
11
12
1
2 AM
Time
Task order
A
B
C
D

Maskinkode	Assemblernotasjon	
0xx	HLT	Stopp!
1xx	ADD	Addisjon og subtraksjon
2xx	SUB	
3xx	STO	Lagre til minnet Hente fra minnet
4xx	-	
5xx	LDA	
6xx	BRA	Hopping
7xx	BRZ	
8xx	BRP	
901	INP	Input/Output (IO)
902	OUT	
922	OTC	

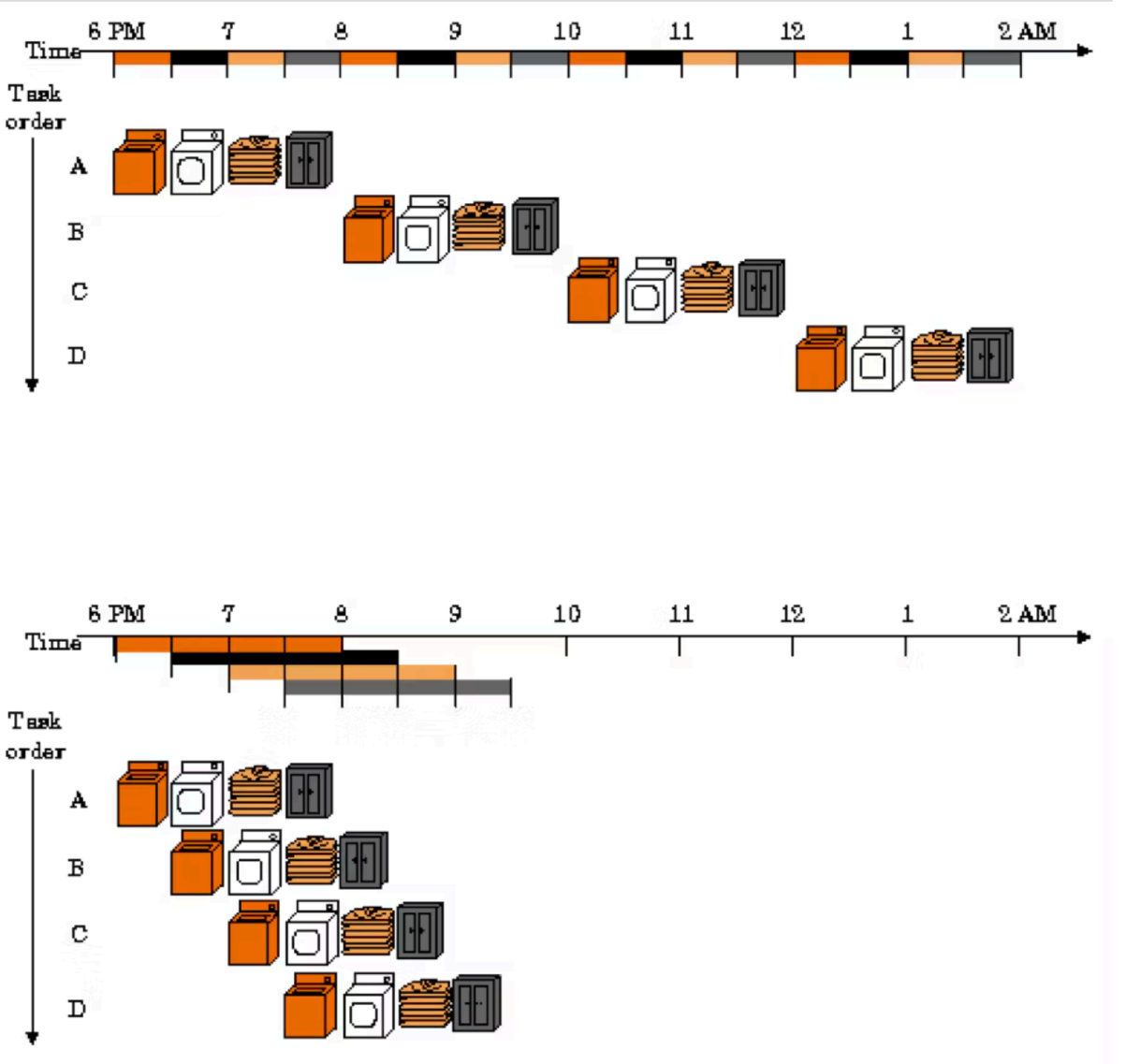
Hva skjer når maskinen kjører assembler/maskinkode?

Menti
0

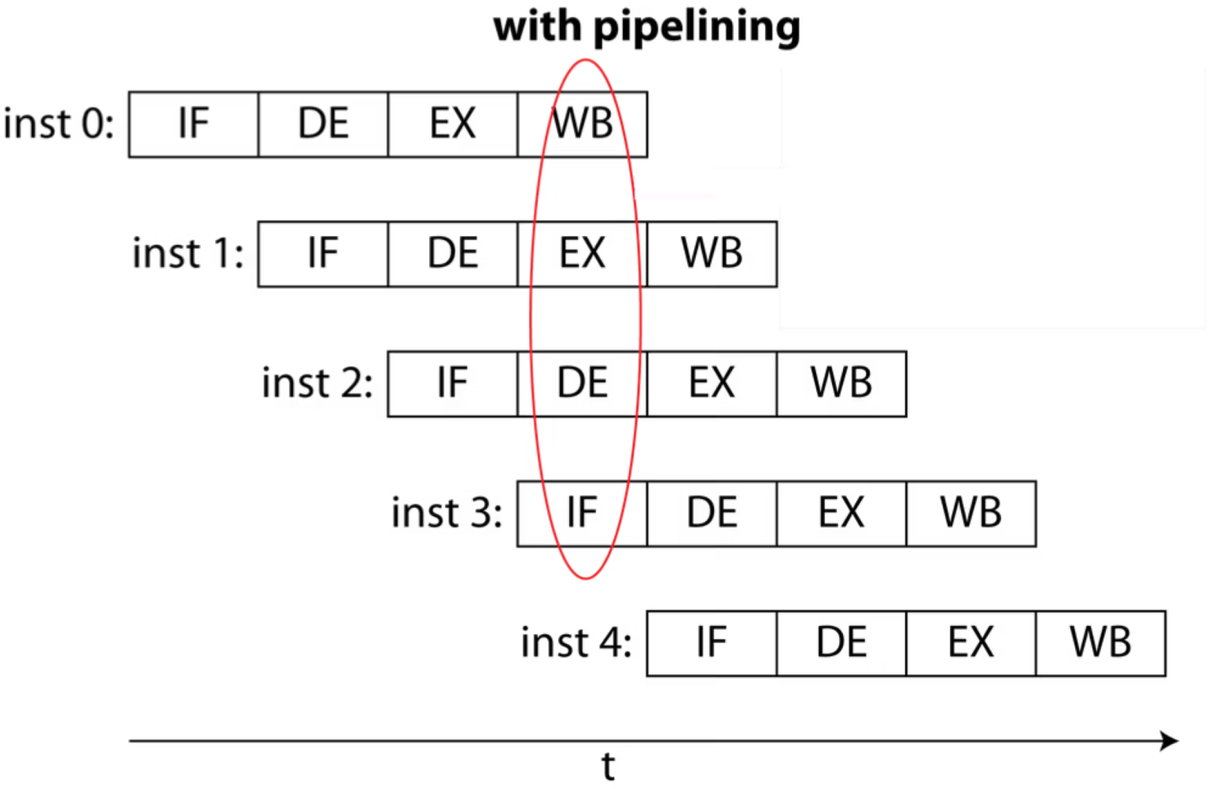
Pipeline med subinstruksjoner

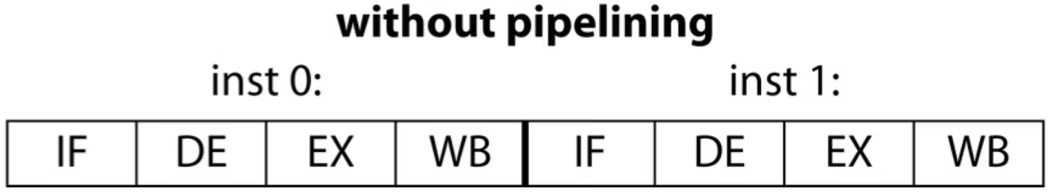
- IF - Instruction fetch
- DE - Decode instruction
- EX - Execute
- WB - Write Back

Menti
0

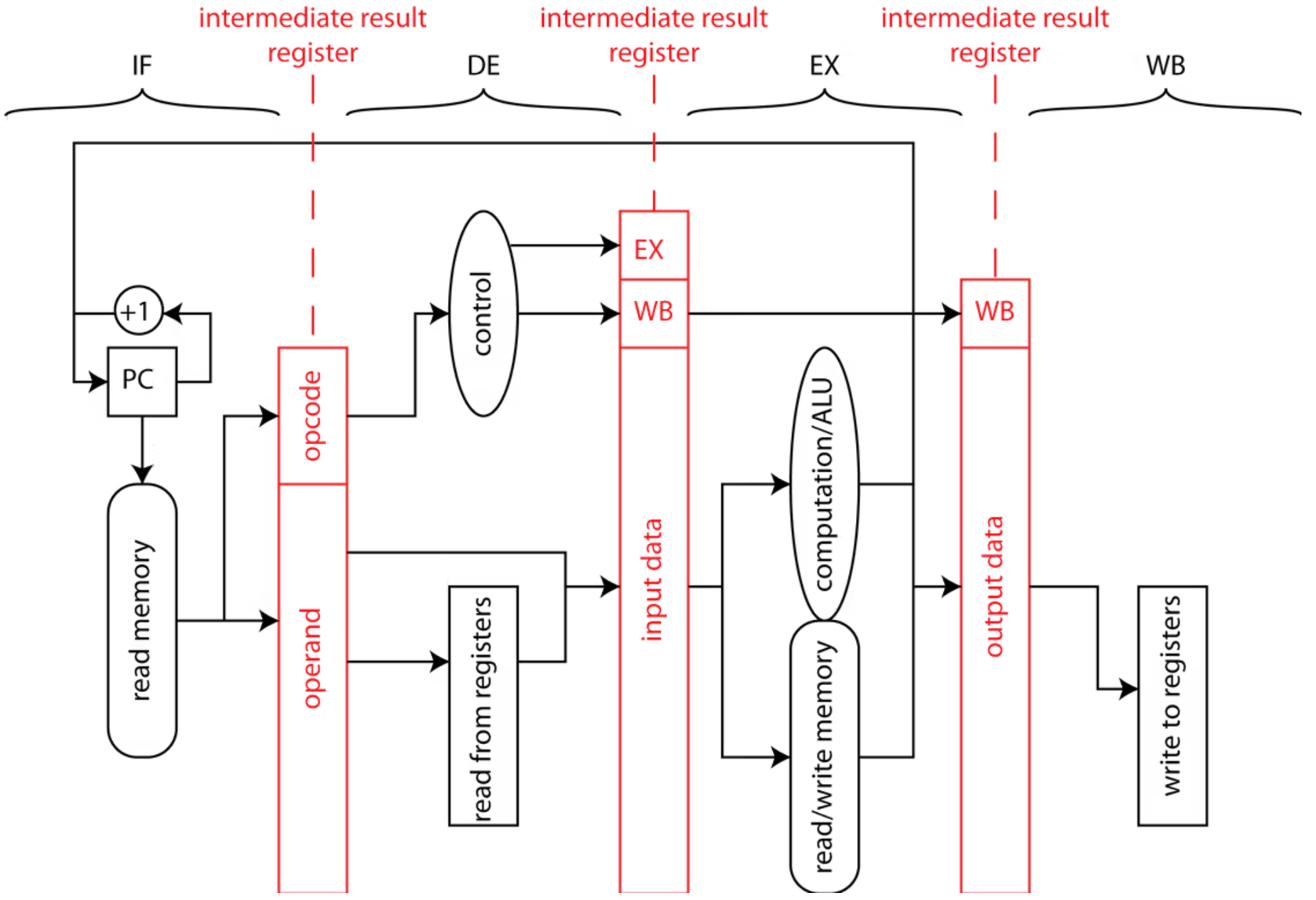

6 PM
7
8
9
10
11
12
1
2 AM
Time
Task order
A
B
C
D
6 PM
7
8
9
10
11
12
1
2 AM
Time
Task order
A
B
C
D

Menti


with pipelining
inst 0:
IF
DE
EX
WB
inst 1:
IF
DE
EX
WB
inst 2:
IF
DE
EX
WB
inst 3:
IF
DE
EX
WB
inst 4:
IF
DE
EX
WB
t


without pipelining
inst 0:
inst 1:
IF
DE
EX
WB
IF
DE
EX
WB

0


intermediate result register
intermediate result register
intermediate result register
IF
DE
EX
WB
+1
PC
read memory
opcode
operand
control
read from registers
EX
WB
input data
computation/ALU
read/write memory
WB
output data
write to registers

Menti

Pipeline med flere trinn

- Denne 4-trinns pipeline er den korteste som finnes for en CPU
- Moderne CPU har typisk vesentlig flere trinn

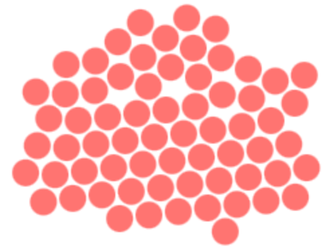
0

Menti

Gitt en pipeline med 4 trinn, vil vi kunne oppnå 4-ganger så høy hastighet enn 1 trinns?



4 Ja, selvfølgelig



66 Ja, med forbehold



16 Nei, absolutt ikke



15 Nei, men jeg vet ikke hvorfor

101/101

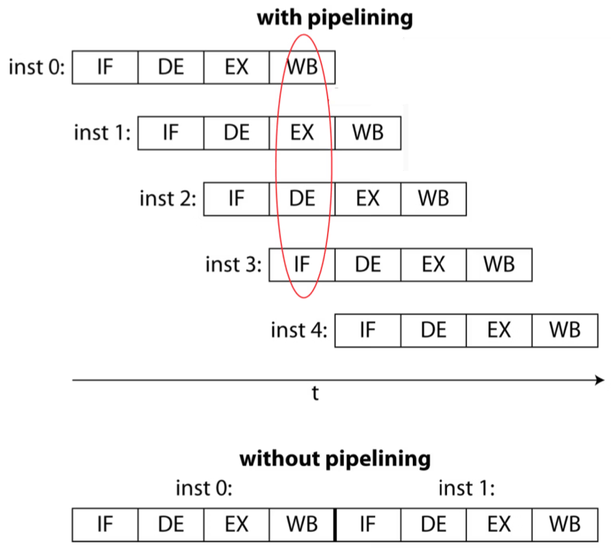
Speed-up

Det å ha en 4 trinns pipeline betyr ikke at man får 4 ganger raskere prosessering. Det går alltid noe tid bort til administrering av instruksjoner og komplikasjoner!

Menti
0

Menti

Hvilke komplikasjoner kan oppstå ved pipeline?



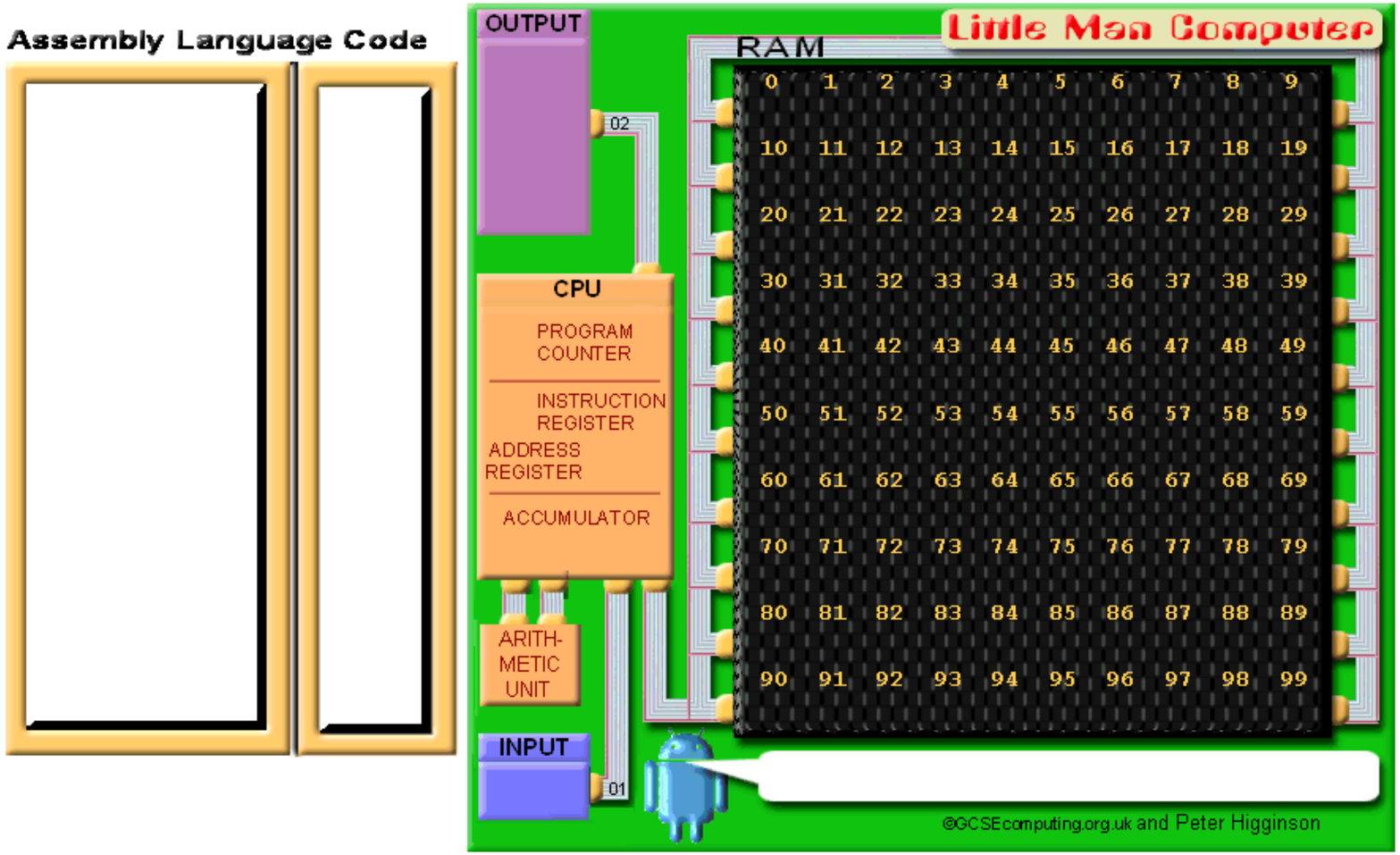
datahasard	overkjøringer	lekasje	Hassards
følgefeil	halt	røret kan bli tett	problemer med minne

60/60

BUS

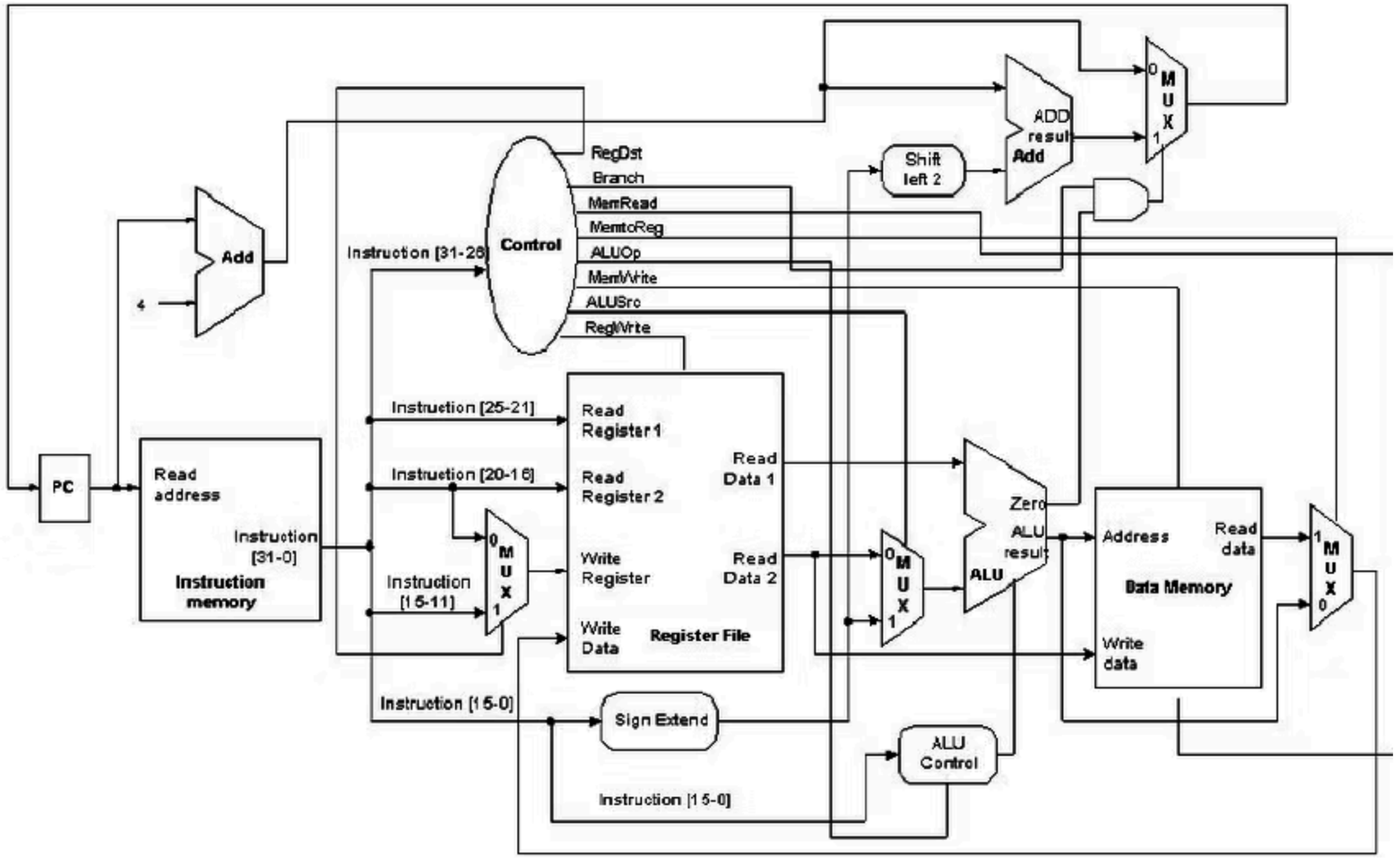
- Kommunikasjonskanal (mellom registere, ALU, minne og I/O enheter)
- kan deles mellom flere enheter, men kun en som kan sende av gangen
- Internt i en CPU er den en eller flere bus(er) som overfører data mellom interne registere.

Menti
0



Hvor er BUS i LMC?

Menti
0



Enkel datapath (32-bits)

Menti
0

Menti

Én klokkesykels CPU

- Klokkeperioden må være like lang som den lengste forsinkelsen for enhver instruksjon gjennom data-path´en
- Resultat: Raskere instruksjoner tvinges til å bruke mer tid enn nødvendig
- Mulig løsning: Klokke med variabel periode
- Vurdering: Vanskelig å implementere

0

In1020 - Minnehierarki

Omid Mirmotahari og Yngvar Berg

Menti

0


I DON'T ALWAYS RUN OUT OF MEMORY,
BUT WHEN I DO I DOWNLOAD MORE RAM.

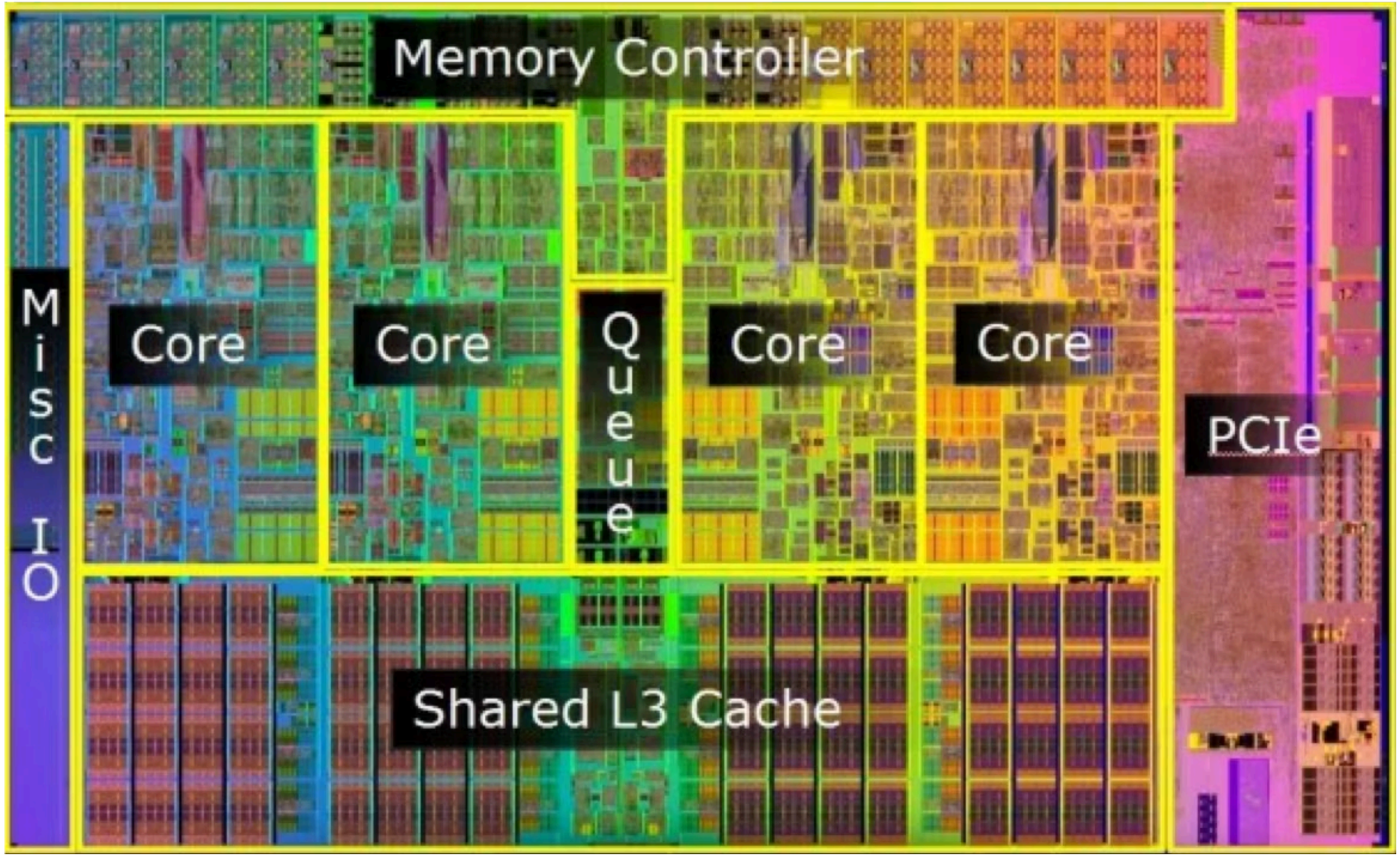
Menti

Teknikker for hastighetsøkning

- Pipeline - øke antall instruksjoner som utføres pr. sekund
- Flere CPU´er i en maskin
- Raskere disker
- Hurtigere og større RAM

0

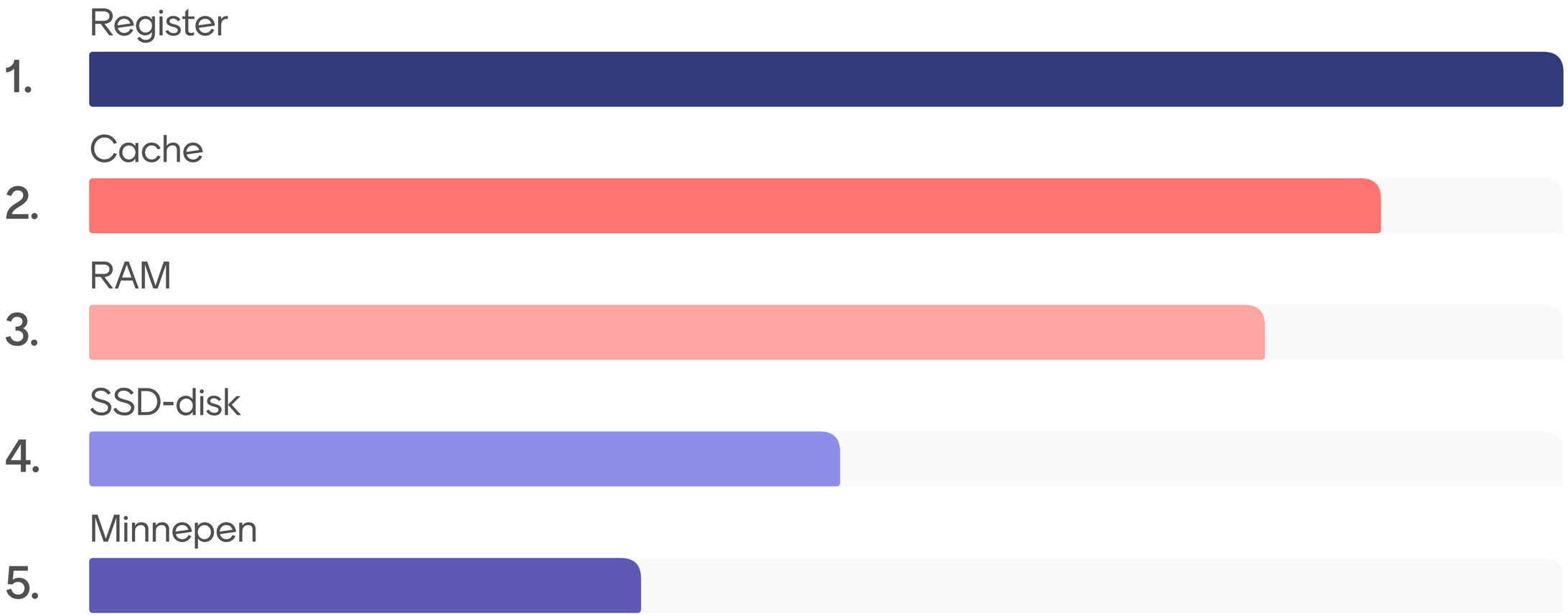
Menti


Memory Controller
Misc IO
Core
Core
Queue
Core
Core
PCIe
Shared L3 Cache

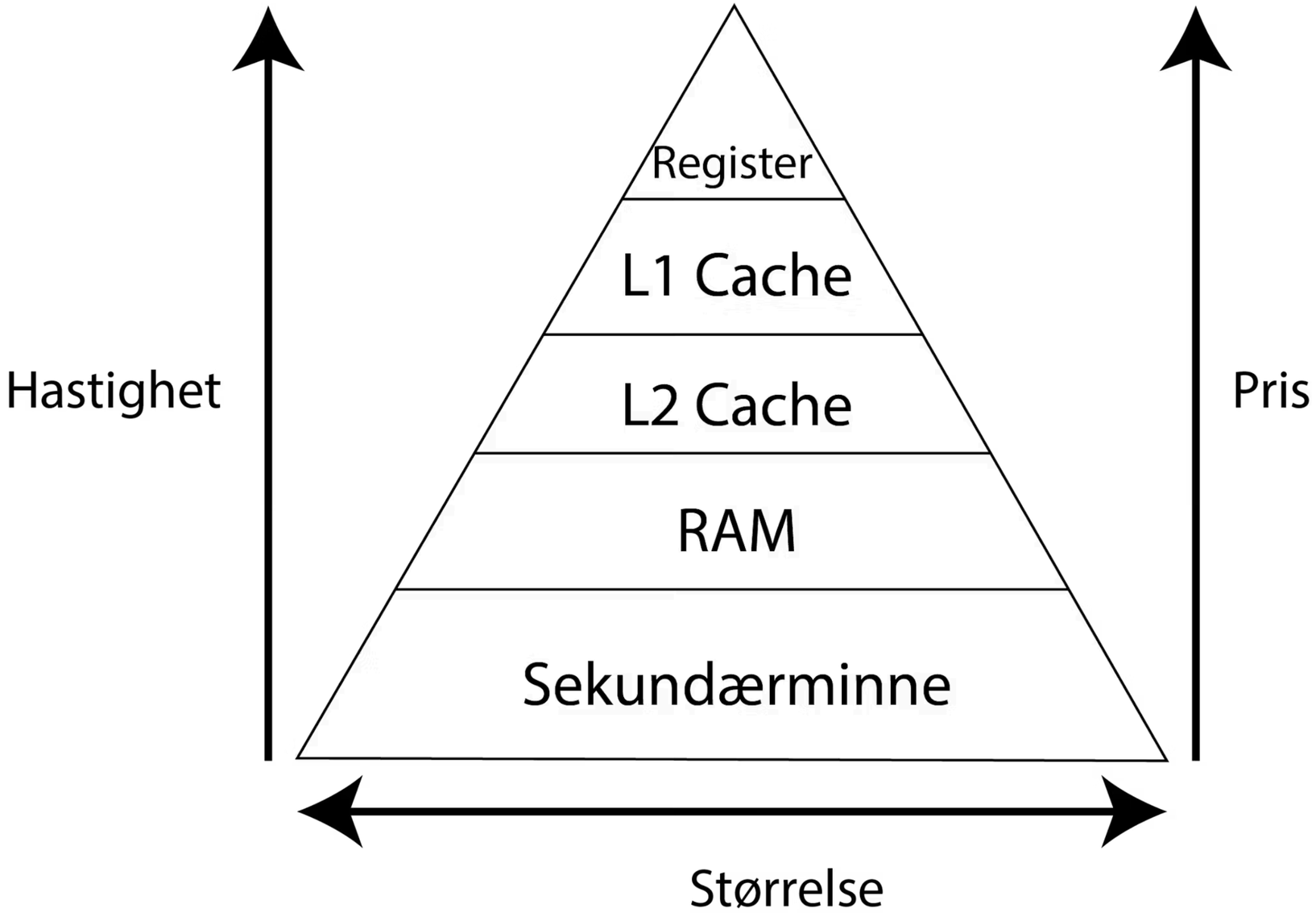
0

Menti

Ranger disse minneelementene etter hastighet.



106/106


Register
L1 Cache
L2 Cache
RAM
Sekundærminne
Hastighet
Pris
Størrelse

Minnehierarki – Bruksområder

- Register

  Intern kladdeblokk for CPU´en med rask aksess til innholdet.

- Cache

  Hurtig mellomlager for både instruksjoner og data for å jevne ut hastighetsforskjellen mellom CPU´en og hurtigminnet.

- RAM

  Buffer mellom eksternt lagringsmedium og CPU´en med rask både lese- og skrive aksess.

- SSD/Flash/Harddisk

  Høykapasitetsmedium for program/data

Menti
0

Menti

Minnehierarki – Lagringskapasitet

- Register

  Integrert på CPU´en, relativt få (32-128 stykker)

- Cache

  Mellomlager internt (L1) eller i nærheten (L2,L3) av CPU´en, typiske kapasiteter er fra 10 KiloByte (L1) til 1 MegaByte (L2) og flere MegaByte (L3).

- RAM

  Internt på hovedkortet i nærheten av CPU´en, størrelser opptil flere GigaByte.

- SSD/Flash/Harddisk

  Ekstern eller intern lagringsenhet i maskinen med kapasitet opptil flere TeraByte.

0

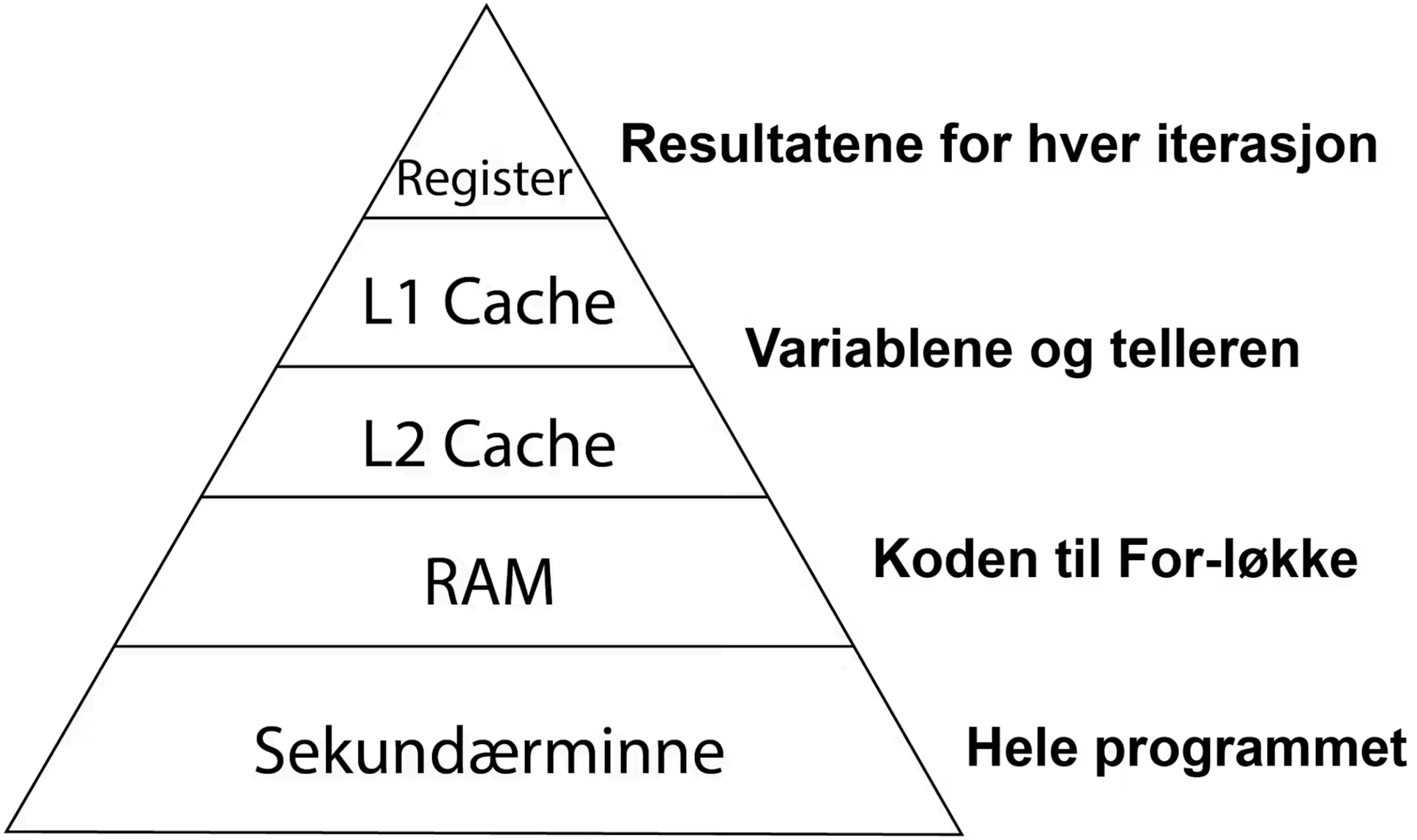
Menti

Minnehierarki – aksesshastighet

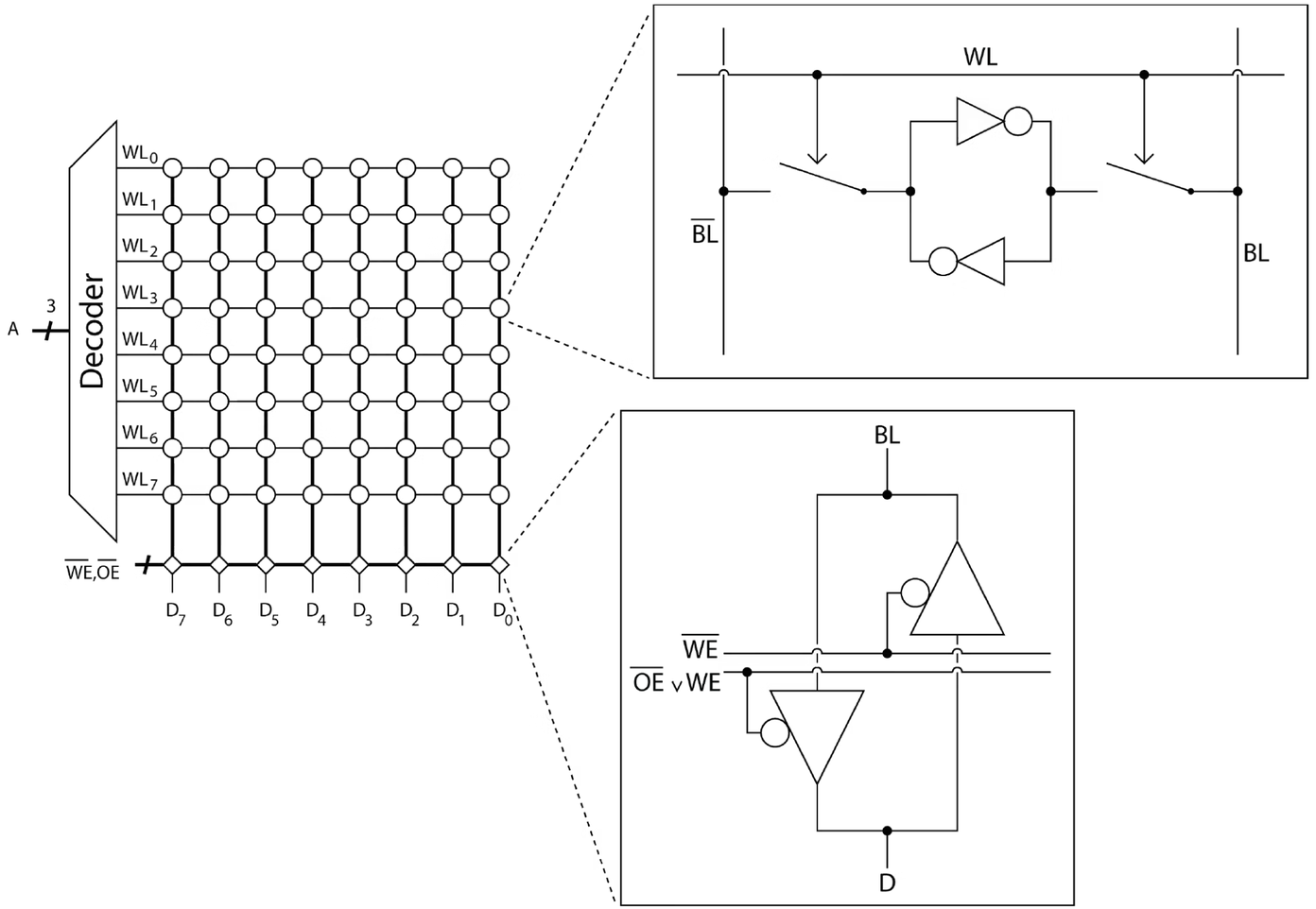
Registers	< 1ns	≈ 100 Byte
L1 (på CPU) cache	≈ 1ns	≈ 10 KB
L2,L3 (utenfor CPU) cache	2-10ns	≈ 1 MB
Hovedminne (RAM)	20-100ns	≈ 1 GB
SSD/Flash	100ns-1us	≈ 1 TB
Harddisk	1ms	≈ 1 TB

0

Menti


Register
L1 Cache
L2 Cache
RAM
Sekundærminne
Resultatene for hver iterasjon
Variablene og telleren
Koden til For-løkke
Hele programmet

0



Static Random Access Memory (SRAM)

Menti
0

Menti

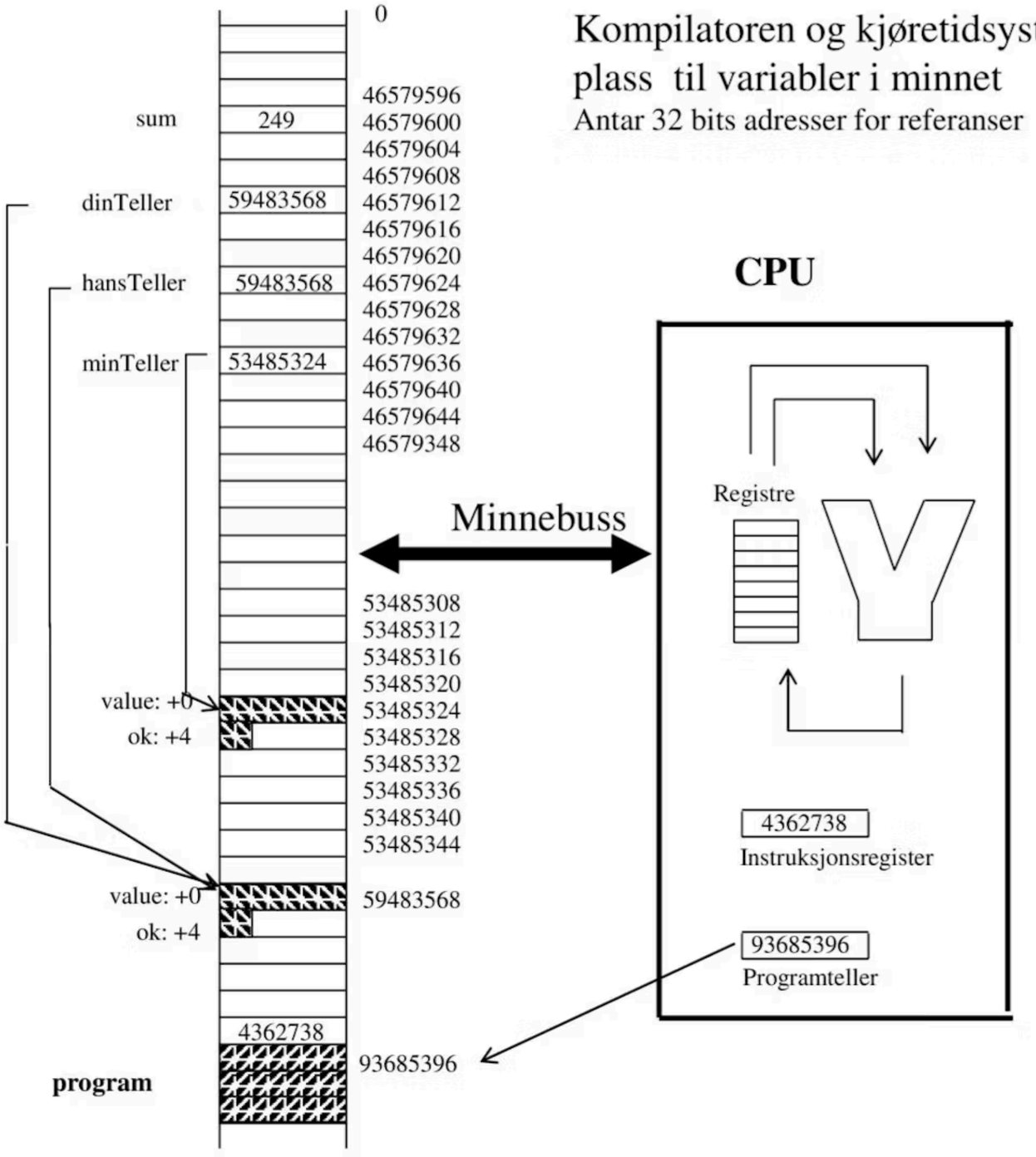
Ram Terminologi

- **Address Space** - *Størrelsen på adresserbare minne enheter, oppgis enten i word eller bites*
- **Word Length** - *Antall bit som kan bli lest/skrevet i en operasjon*
- **Memory size** - *Størrelsen er gitt av address space x word length*

0

Kompilatoren og kjøretidsystemet setter av plass til variabler i minnet

Antar 32 bits adresser for referanser



sum ligger minnelokasjon 46579600, 46579601, 46579602 og 46579603

dinTeller ligger minnelokasjon 46579612, 46579613, 46579614 og 46579615

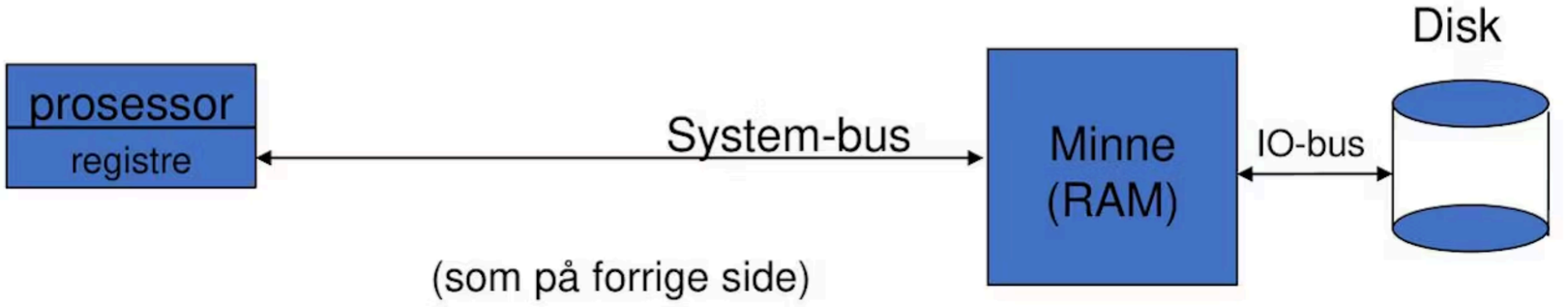
hansTeller ligger minnelokasjon 46579624, 46579625, 46579626 og 46579627

minTeller ligger minnelokasjon 46579636, 46579637, 46579638 og 46579639

value ligger i starten av objektet, mens ok ligger 4 byte ut i objektet

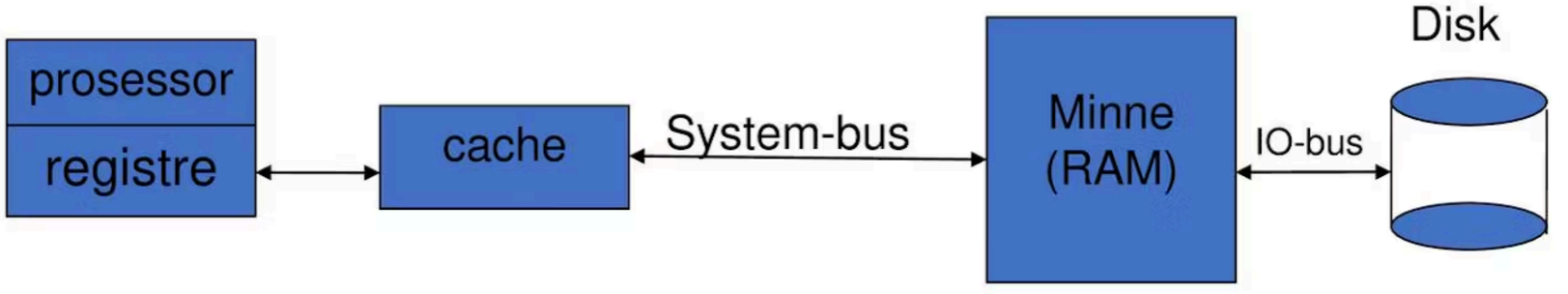
Datamaskinarkitektur (Computer Architecture)

1. I gamle dager =
(nesten) fremdeles dagens abstrakte modell:



(som på forrige side)

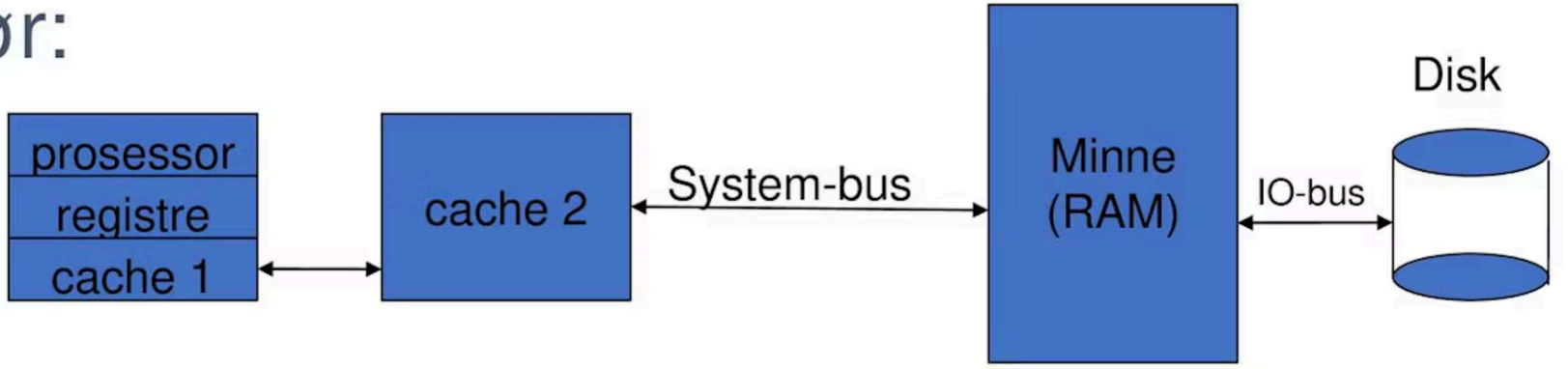
2. Før:



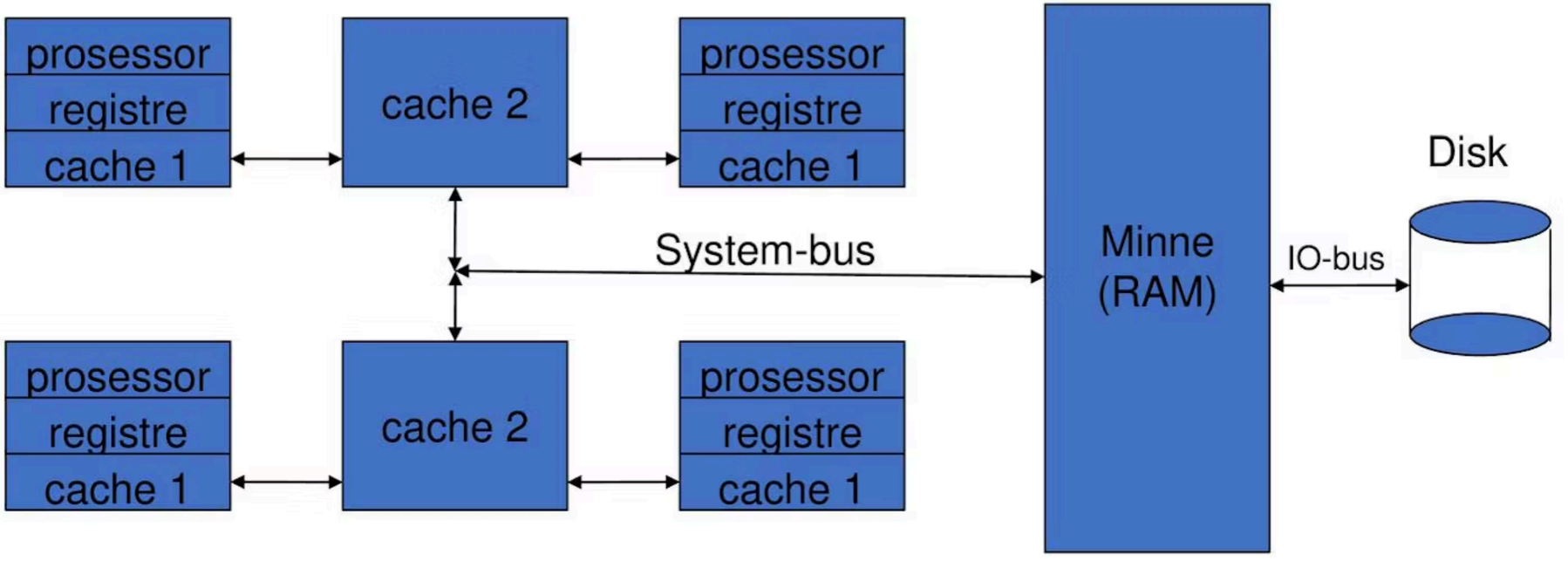
Menti
0

Datamaskinarkitektur (Computer Architecture)

3. Før:



4. Nå, f.eks:



Menti

Hit og Miss

Cache-hit og cache-miss

RAM-hit and RAM-miss

0

Koherens

Skrive til minne tar lengre tid enn å lese fra minne

Menti

0

Spørsmål

17 questions
3 upvotes

Menti

0