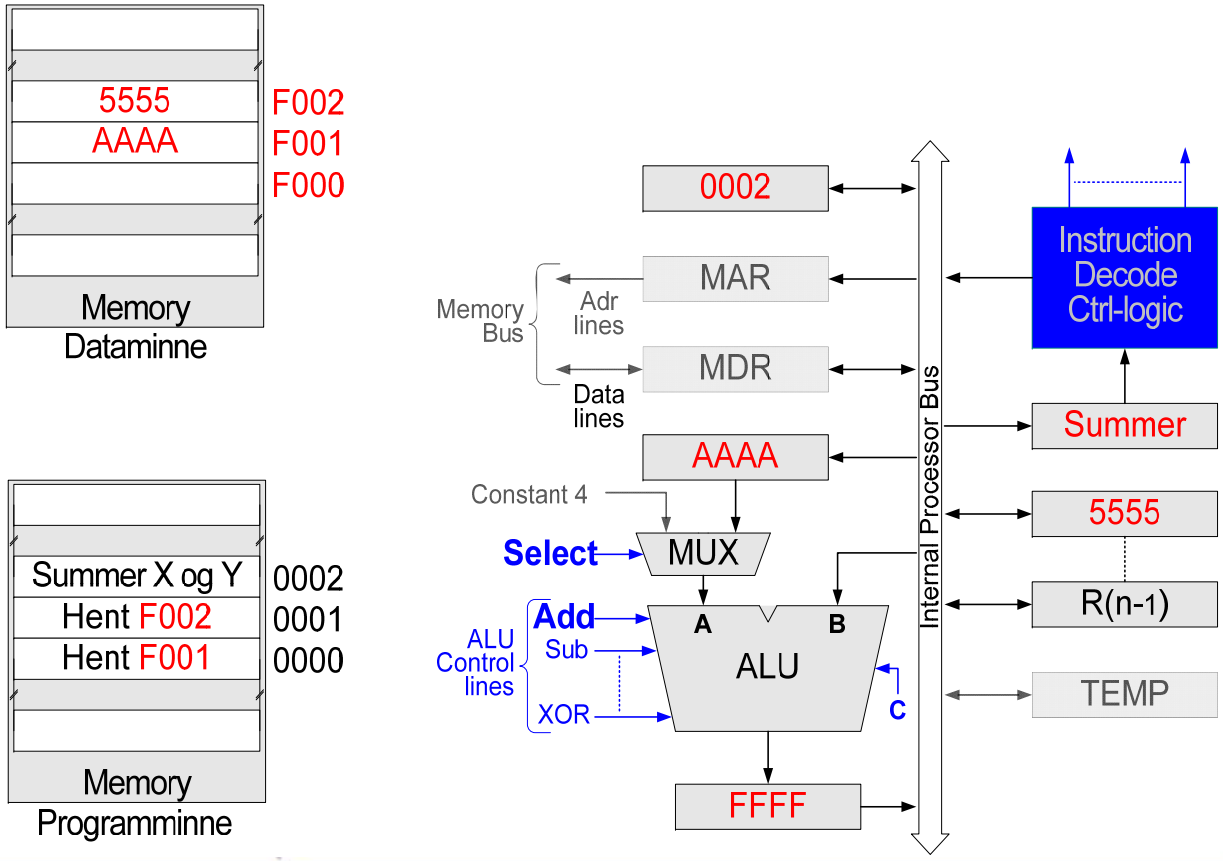
TDT4160
Datamaskiner Grunnkurs
2008

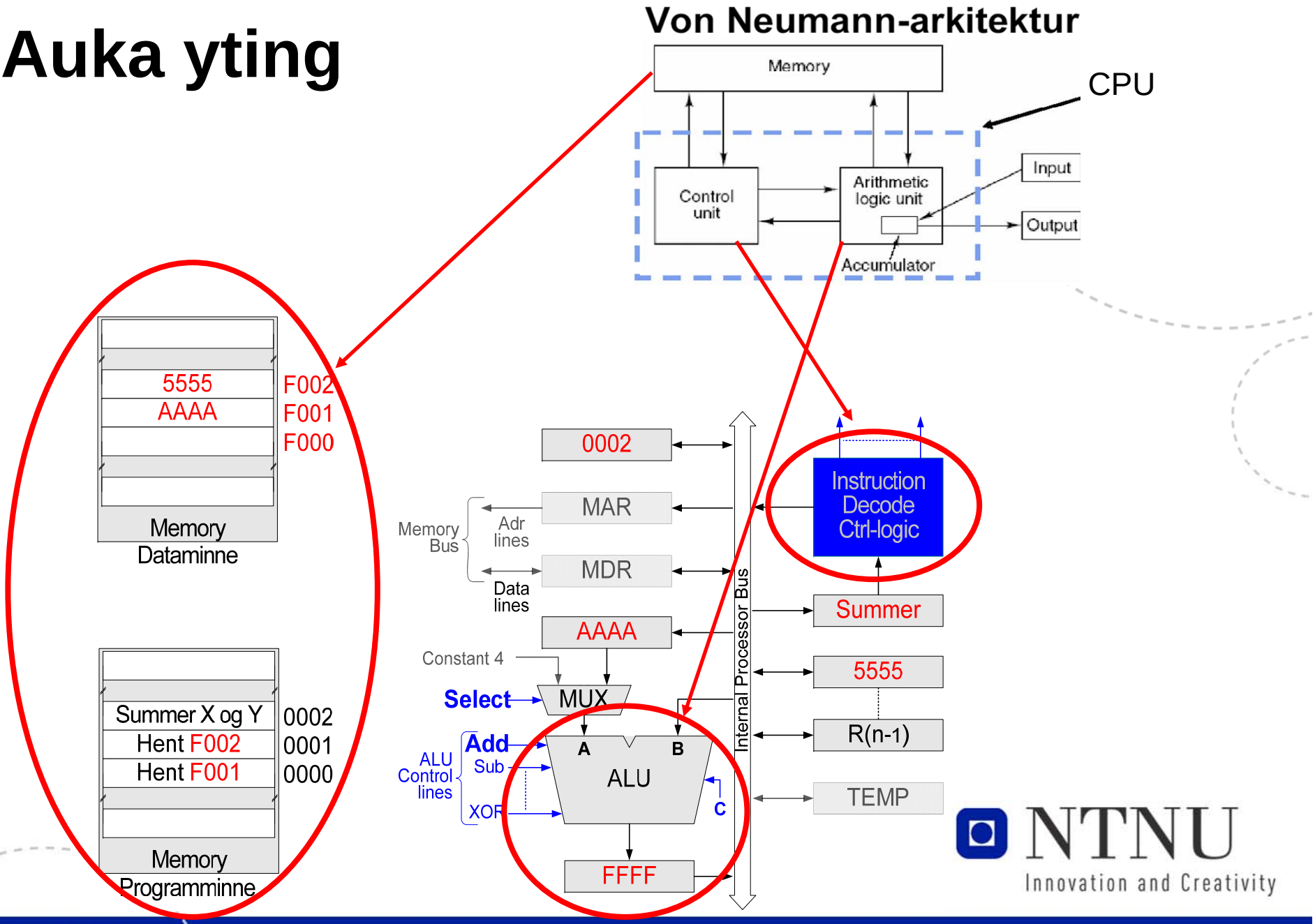
Gunnar Tufte

Auka yting



NTNU
Innovation and Creativity

Auka yting

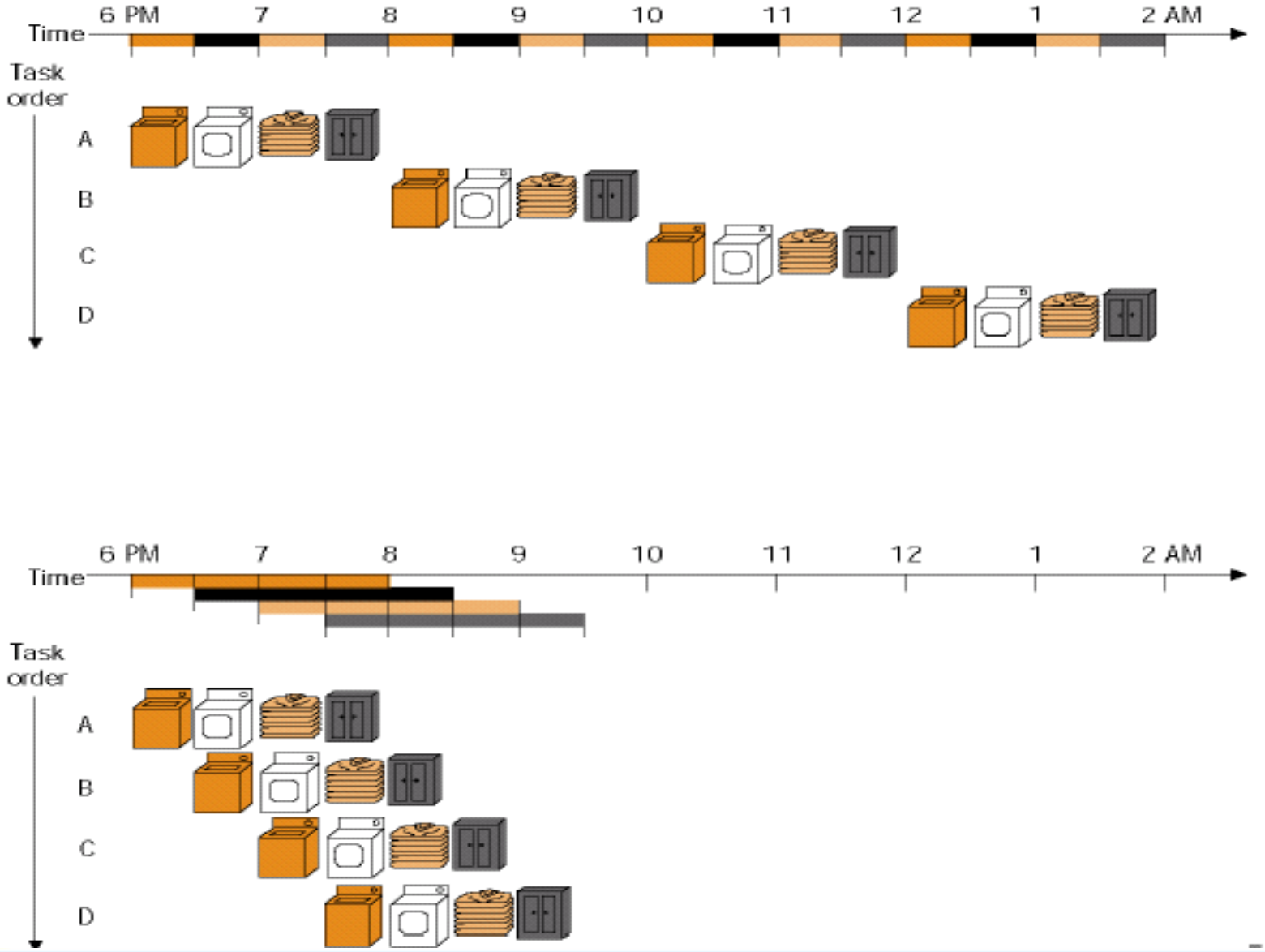


NTNU
Innovation and Creativity

Parallellitet

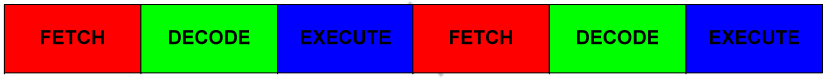
- Essensielt for å øke ytelse
- To typer:
  - 1) Instruksjonsnivåparallellitet
    - Fleire instruksjonar utføres samtidig (1 prosessor)
    - Samleband
      - Eks: Unødvendig at ALU er ledig mens CPU leser instruksjon
    - Superskalaritet: Duplisering av CPU-komponenter
  - 2) Prosessornivåparallellitet
    - Ein prosessor er bra, fleire er betre?

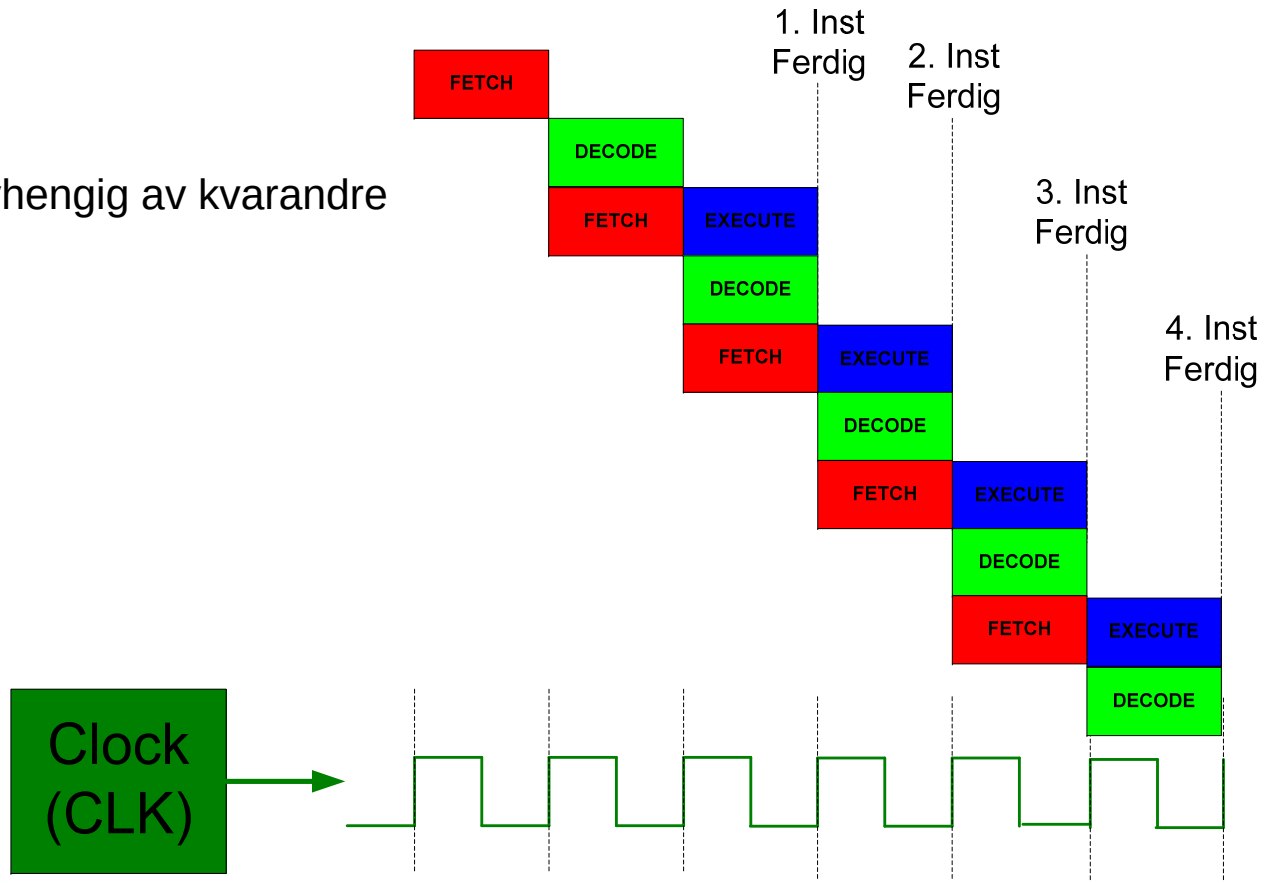
Samleband (Pipelining)



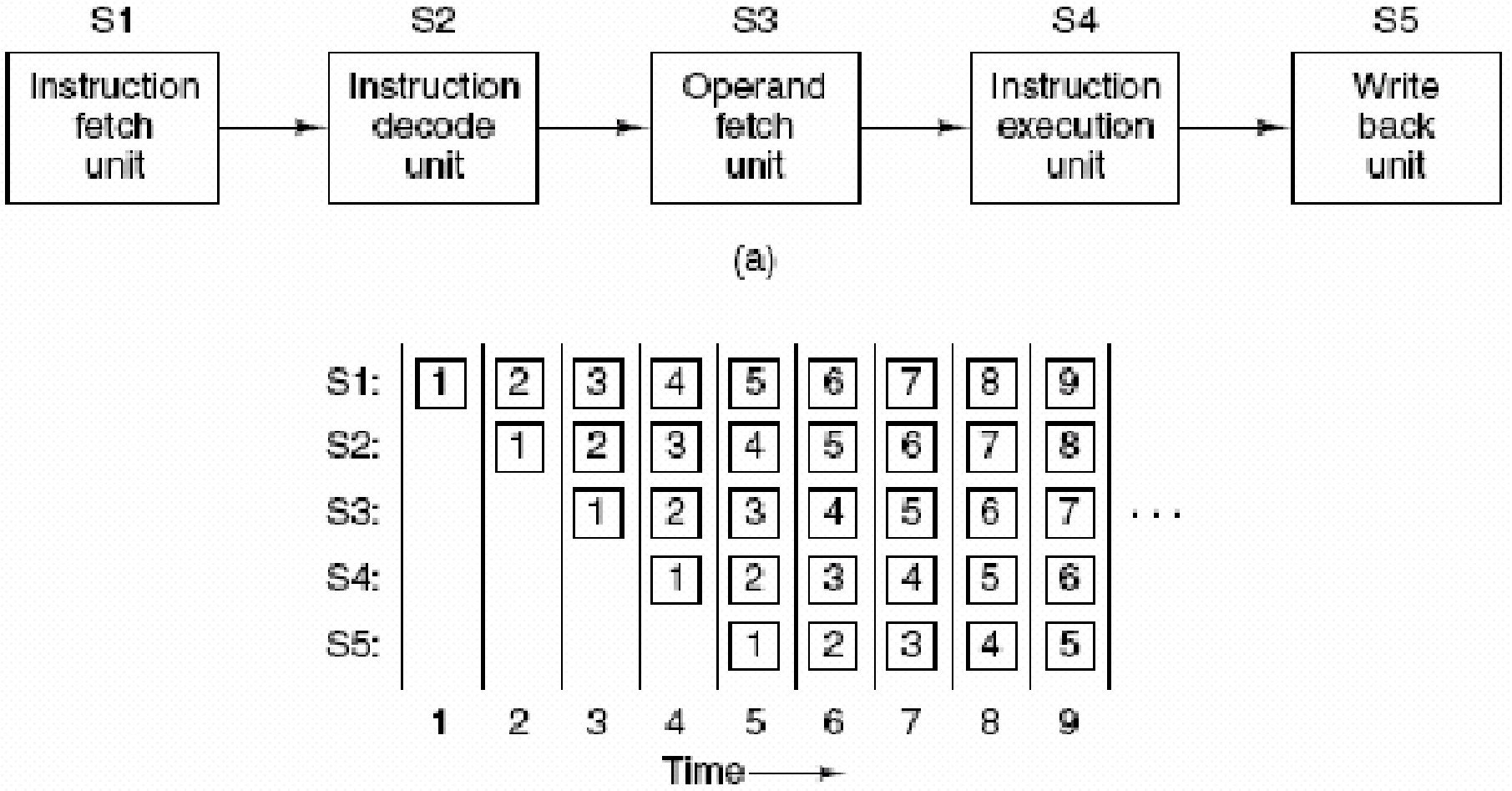
Samleband (Pipelining) 2

- **Fetch Decode Execute**
- Dei tre prinsipielle stega i instruksjonutføring
- Lagar oss ein tretrinns styreeining
  - 1: Fetch
  - 2: Decode
  - 3: Execute
  - Kan startast uavhengig av kvarandre
- Raskare
  - Enkle trinn
  - Kortare tid





Samleband (Pipelining) 3

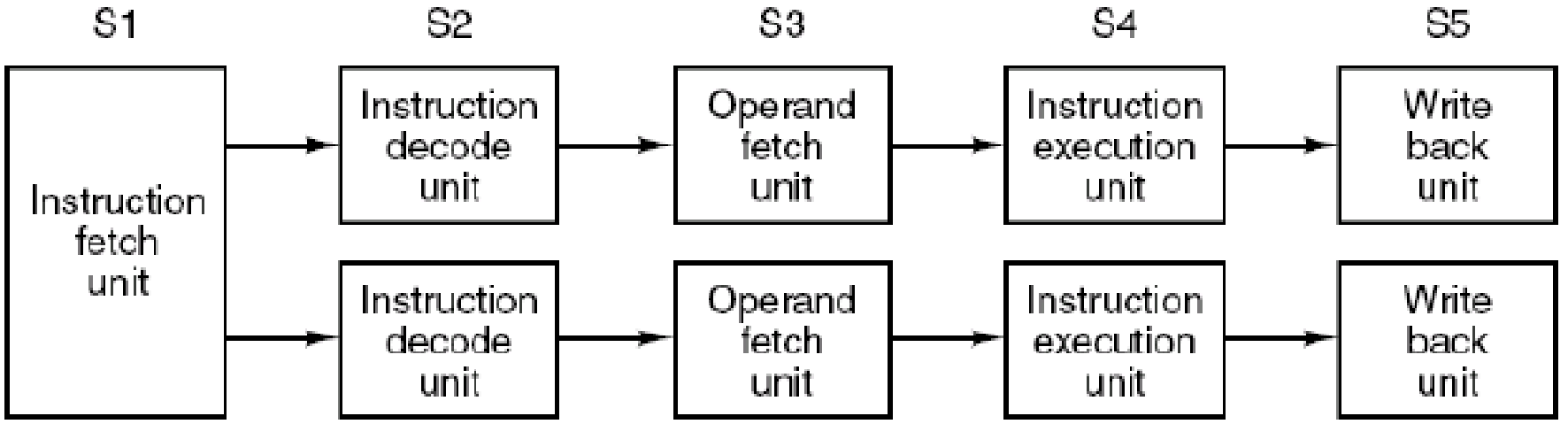


Samleband (Pipelining) 4

- Gitt 5 steg, kvart steg tar en klokkesyklus
  - Instruksjon 1: 5 klokkesyklusar
  - Deretter: 1 instruksjon ferdig per klokkesyklus
- Utan samleband
  - 1 instruksjon ferdig per klokkesyklus (RISC)
- Kvifor er samleband raskare då?
  - Klokkesyklusen kan gjerast mye kortare!
  - 1/5 av arbeidet skjer kvar klokkesyklus
  - Lengde på klokkesyklus bestemt av tregaste steg

  - Utan samleband: alt må gjerast innan ein klokkeperiode

Superskalaritet



- Hentar to instruksjonar om gangen
- Dupliserer steg 2-5
- Har dermed opptil 10 instruksjonar under utføring
- Figuren tilsvarer ca. den første Pentium prosessoren

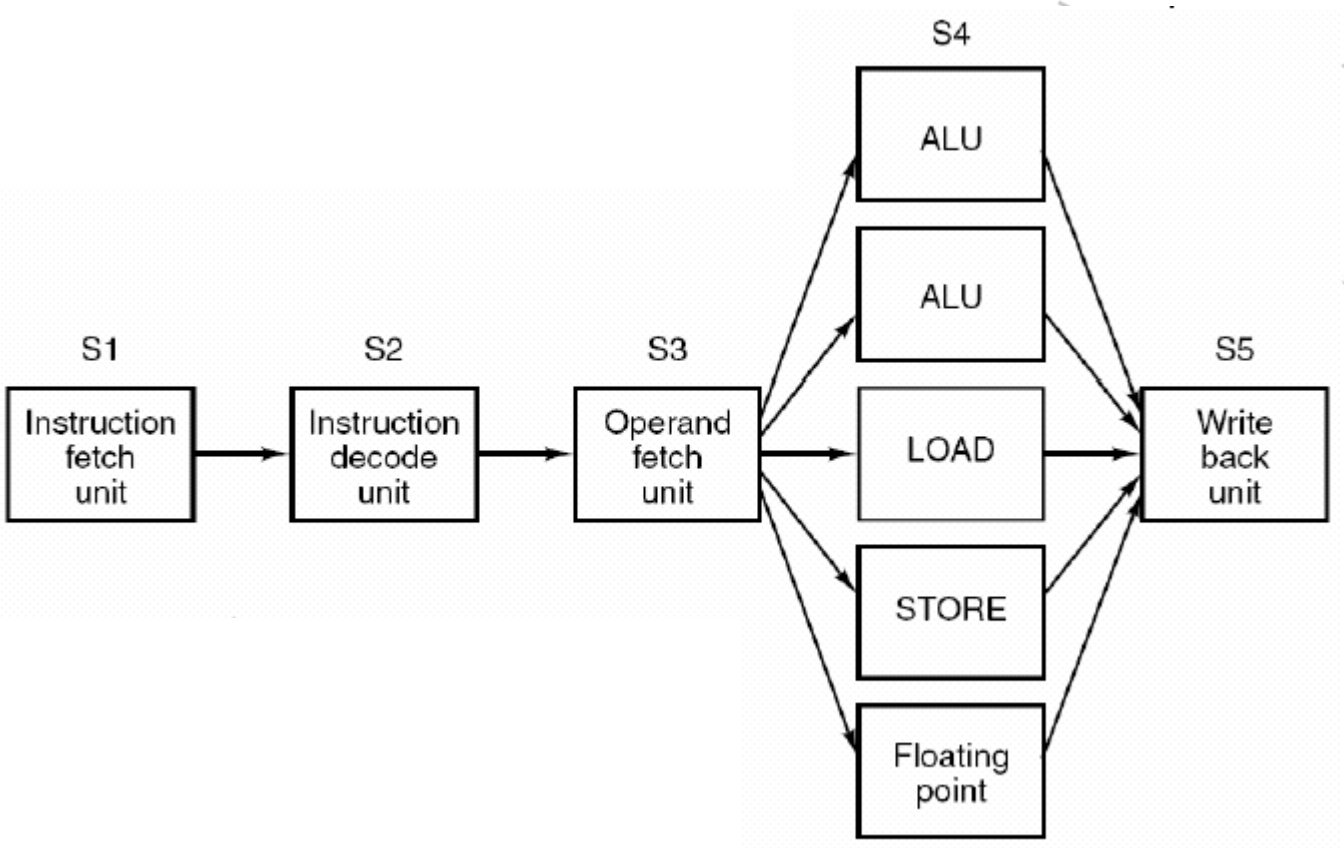
- Prosessornivåparallellitet
- Minne
- 8051, Pentium 4 og UltraSPARC III

Parallellitet

- Essensielt for å auke ytinga
- To typar:
  - 1) Instruksjonsnivåparallellitet
    - Fleire instruksjonar utføres samtidig (1 prosessor)
    - Samleband
      - Eks:
        - Unødvendig at ALU er ledig mens CPU leser instruksjon
        - Kan bruke ALU mens ein kopierar data mellom register
    - Superskalaritet: Duplisering av CPU-komponentar
  - 2) Prosessornivåparallellitet
    - En prosessor er bra, fleire er betre?

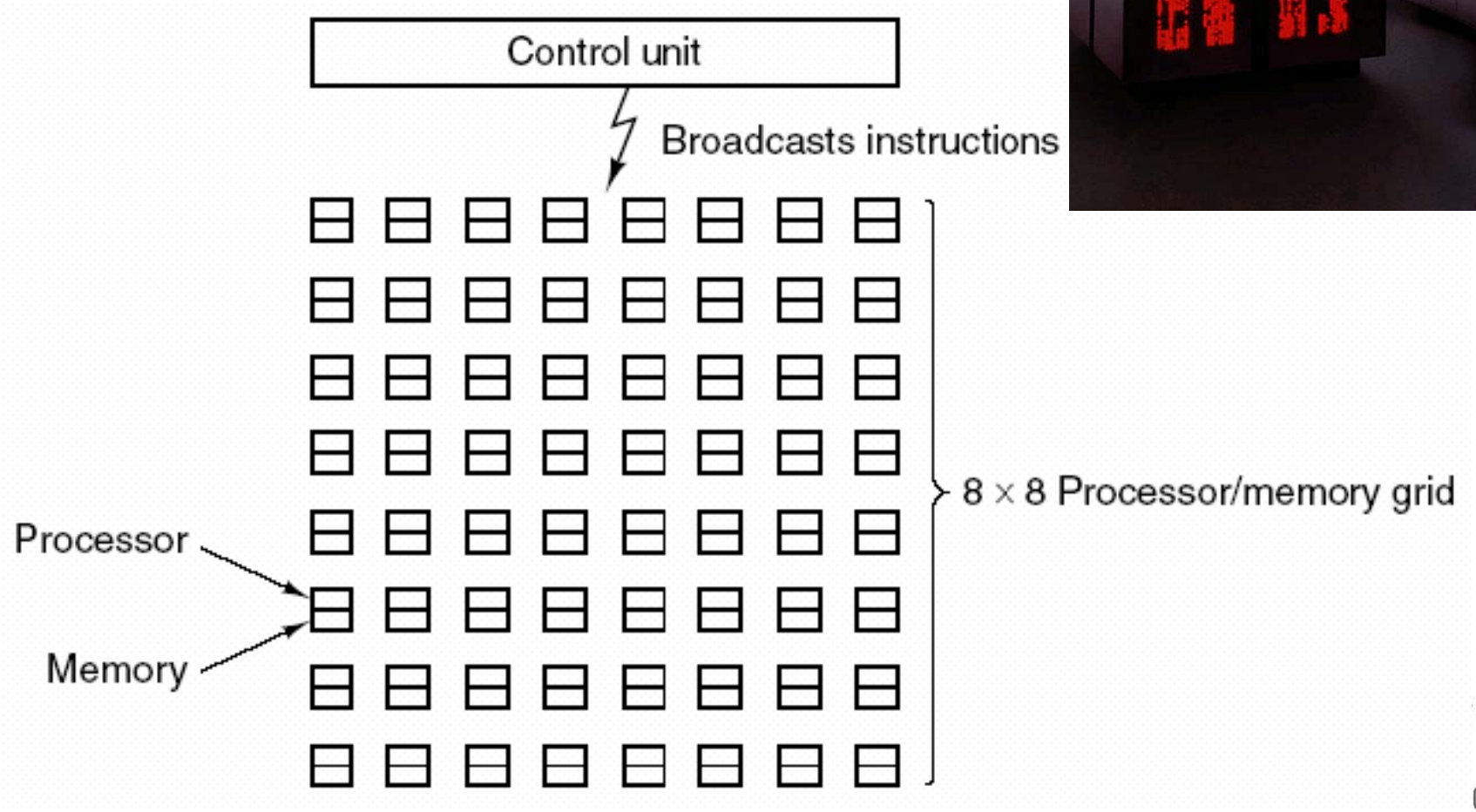
Superskalaritet 2

- Dyrt å duplisere alt 4x (Areal (mengd transistorar))
- Utføring tar mest tid
- Figur tilsvarer Pentium II
- Dupliserar mest tidkrevjande steg



Meir parallellitet 1

Array computer (SIMD)



Creativity

Thinking Machines Corporation

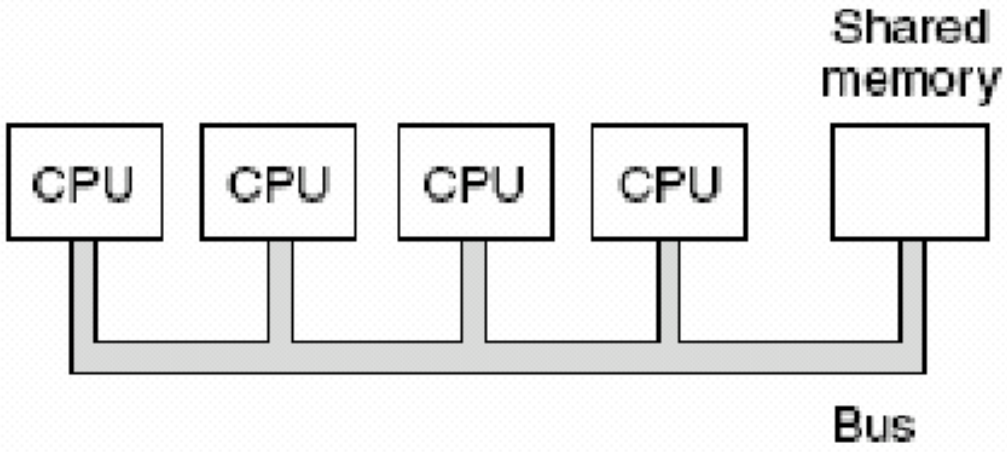
- Connection Machine models CM-1 og CM-2, 65 536 serie prosessorar
- Spesial kompilator overset kode for parallell SIMD program
- Store rundt 1990
  - DARPA redd for å gå glipp av noko
  - Ein av dei mest lovande firma
  - Maskiner stort sett kunn brukande innan forsking
- Konkurs 1994

Connection Machine's cameo in Jurassic Park

NTNU
Innovation and Creativity

Meir parallellitet 2

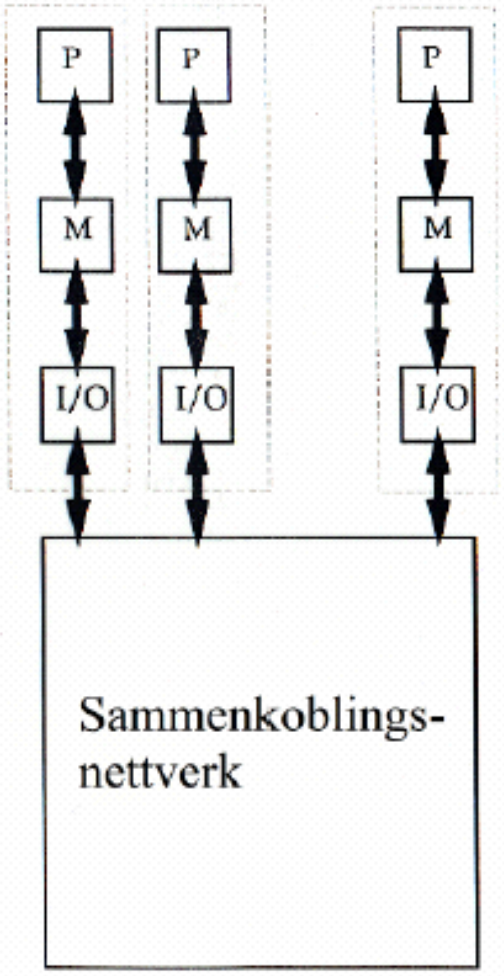
Multiprosessor (MIMD)



- Uavhengige prosessorer (forskjell fra array computer)
- Delt hovedlager
- Buss blir flaskehals etter hvert
  - Kan ha litt lokalt lager til hver prosessor

Meir parallellitet 3

Multidatamaskin (MIMD)



- Distribuert hovedlager
- "Maskiner" med nesten 10.000 prosessorer laget
- Lettere å lage enn multiprosessor
- Men vanskeligere å programmere
  - Ikke delt lager

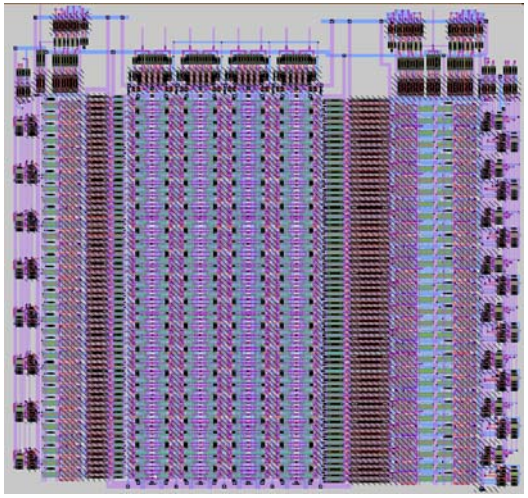
Lager 2.1 2.2

Hard disc

Tape storage

RAM Module

Optical disc

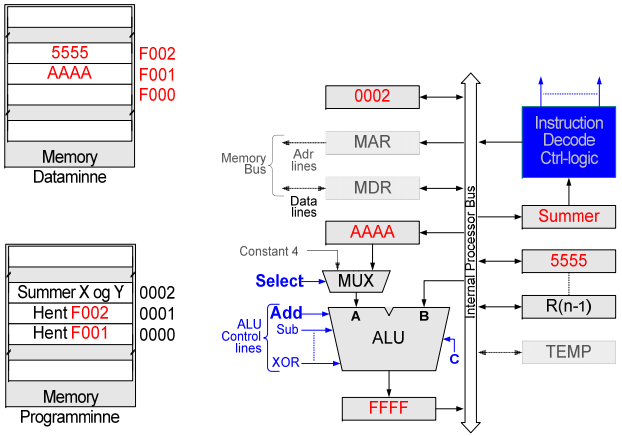


Register bank

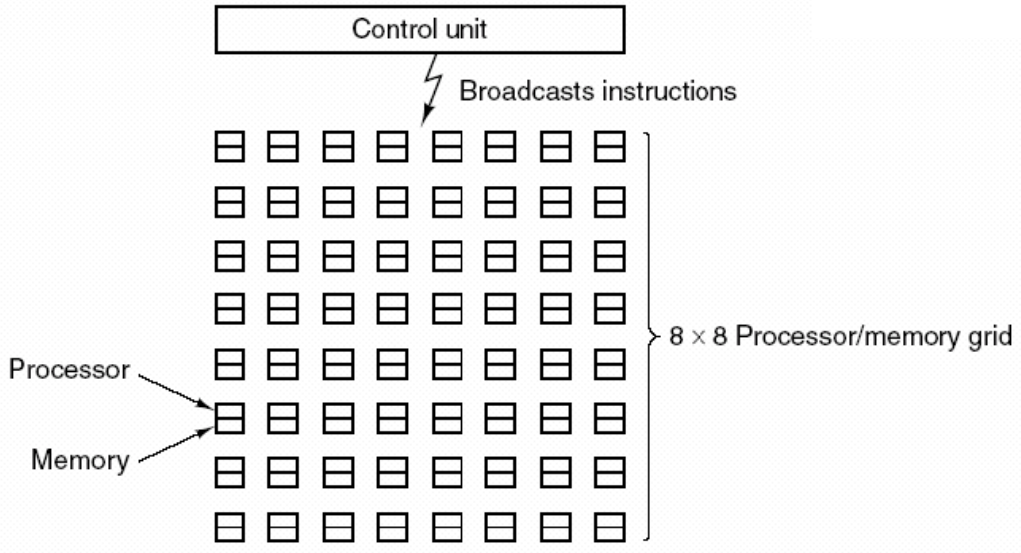
Core memory

NTNU
Innovation and Creativity

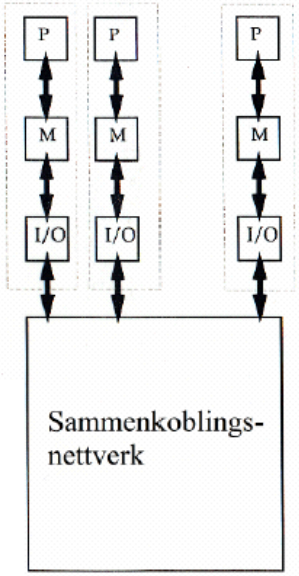
Ein-prosessor maskin



Array computer (SIMD)

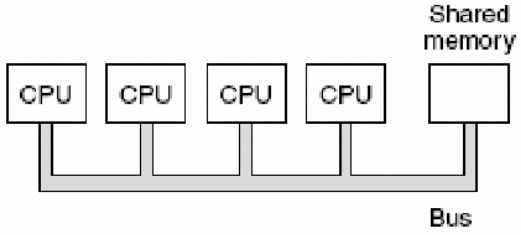


Multidatamaskin (MIMD)



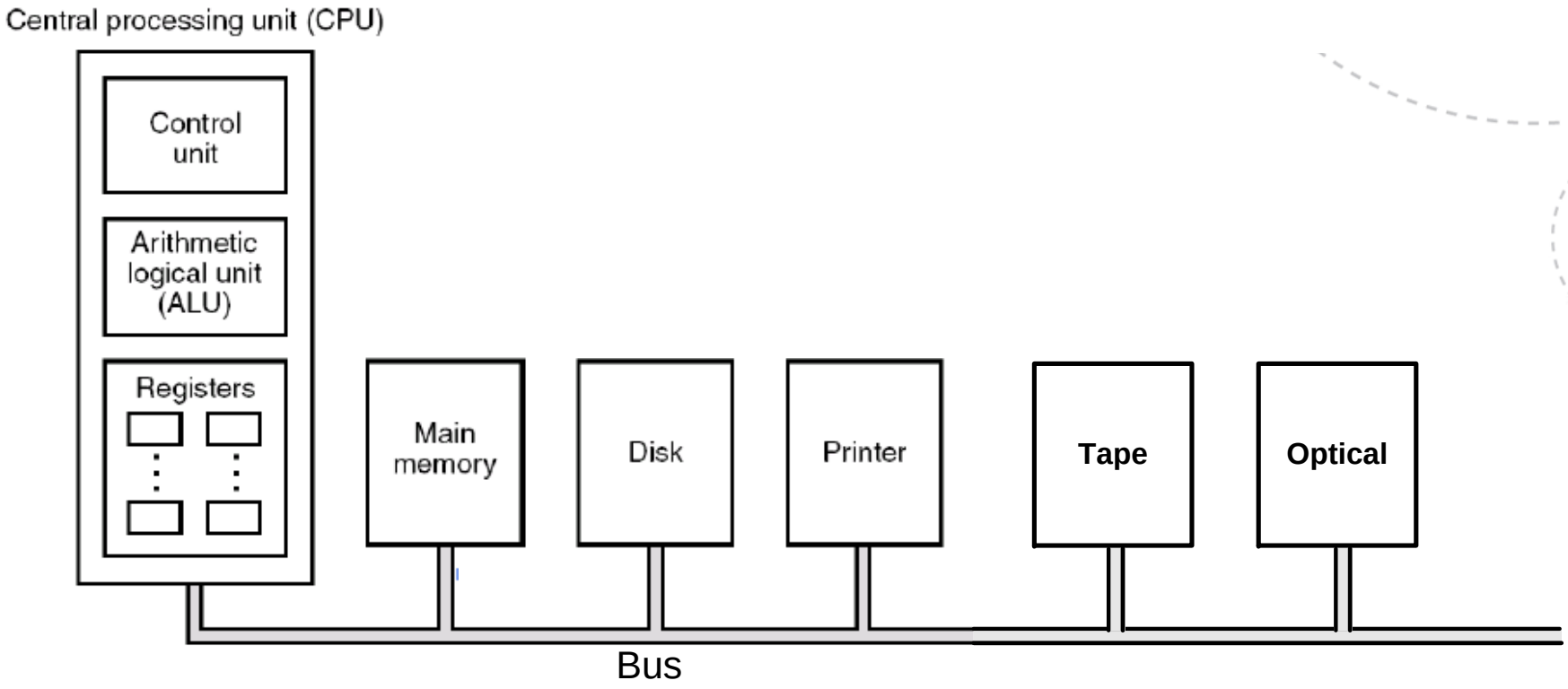
- Distribuert hovedlager
- "Maskiner" med nesten 10.000 prosessorer laget
- Lettere å lage enn multiprosessor
- Men vanskeligere å programmere
  - Ikke delt lager

Multiprosessor (MIMD)



- Uavhengige prosessorer (forskjell fra array computer)
- Delt hovedlager
- Buss blir flaskehals etter hvert
  - Kan ha litt lokalt lager til hver prosessor

Lager og prosessor overordna



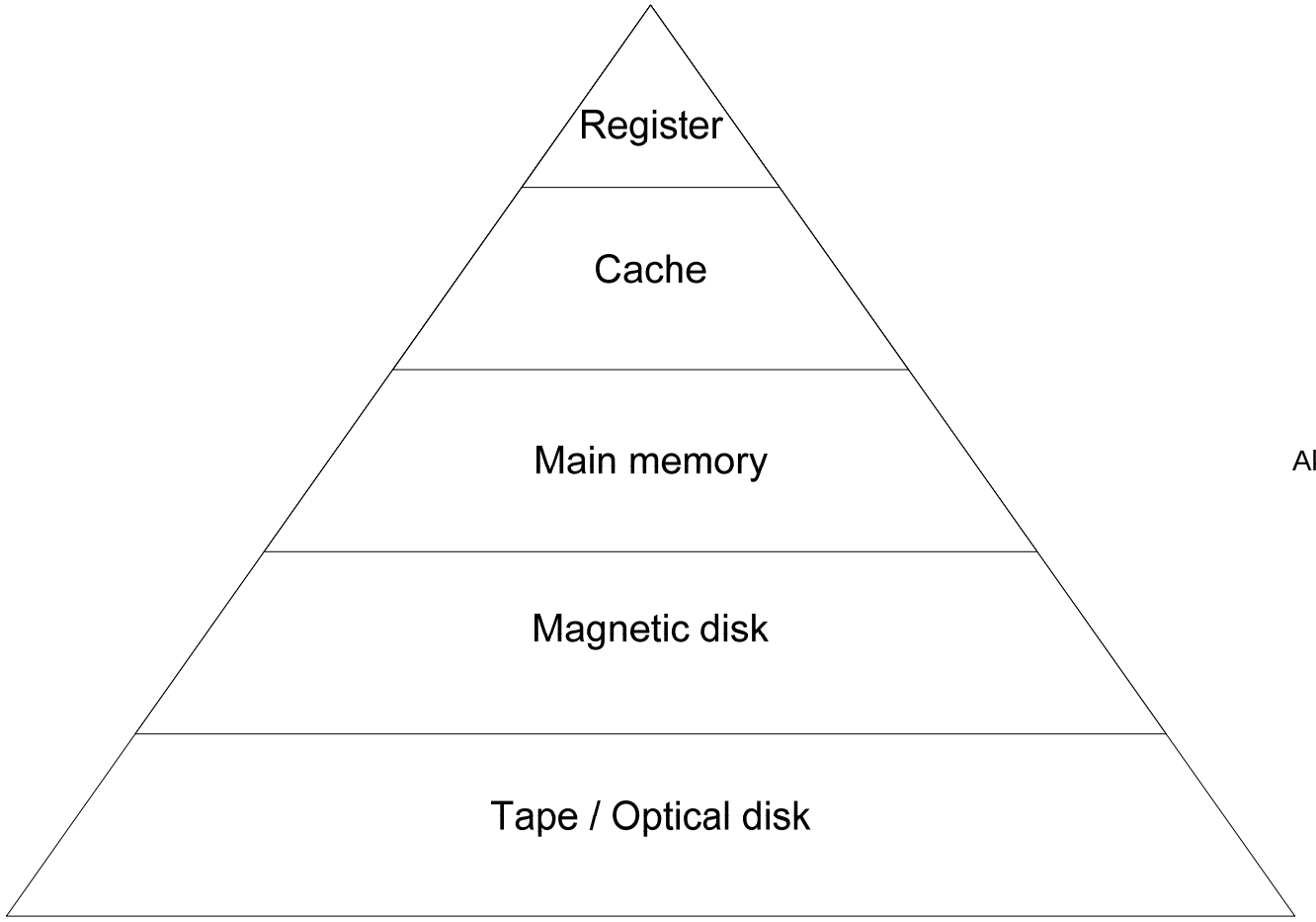
Lagerhierarki

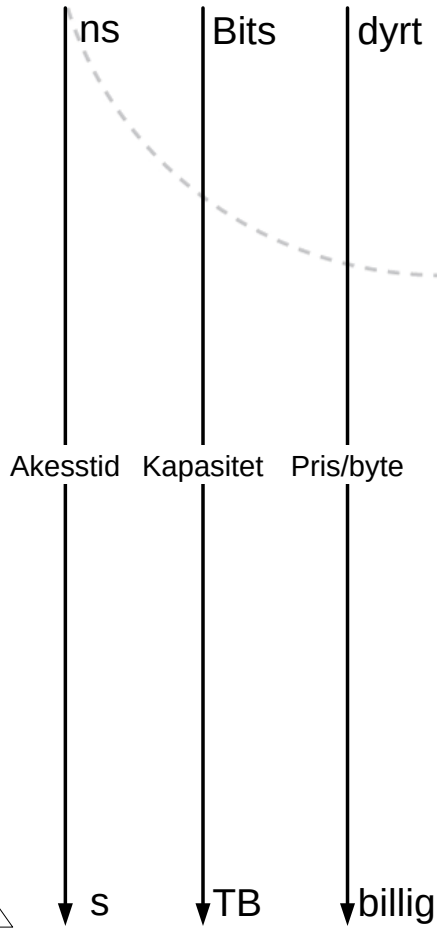
- Ofte eit hierarki av minne
  - Yting
  - Fysiske avgrensingar
  - Økonomi
  - Fleksibilitet

  - Maksimal ytels

- Hierarki bygd opp
  - Fysisk plassering
  - Akesstid
  - Størrelse

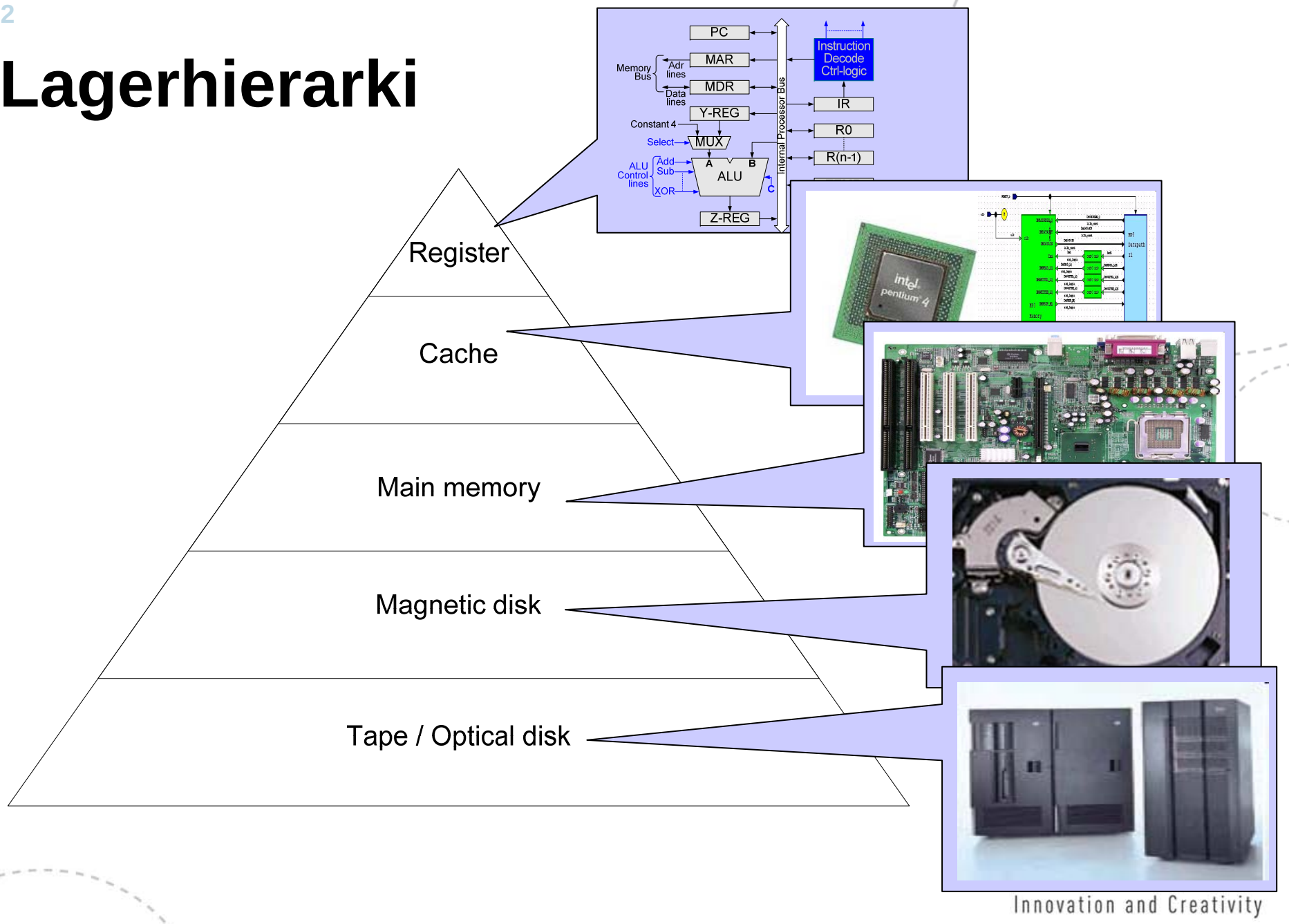
Lagerhierarki



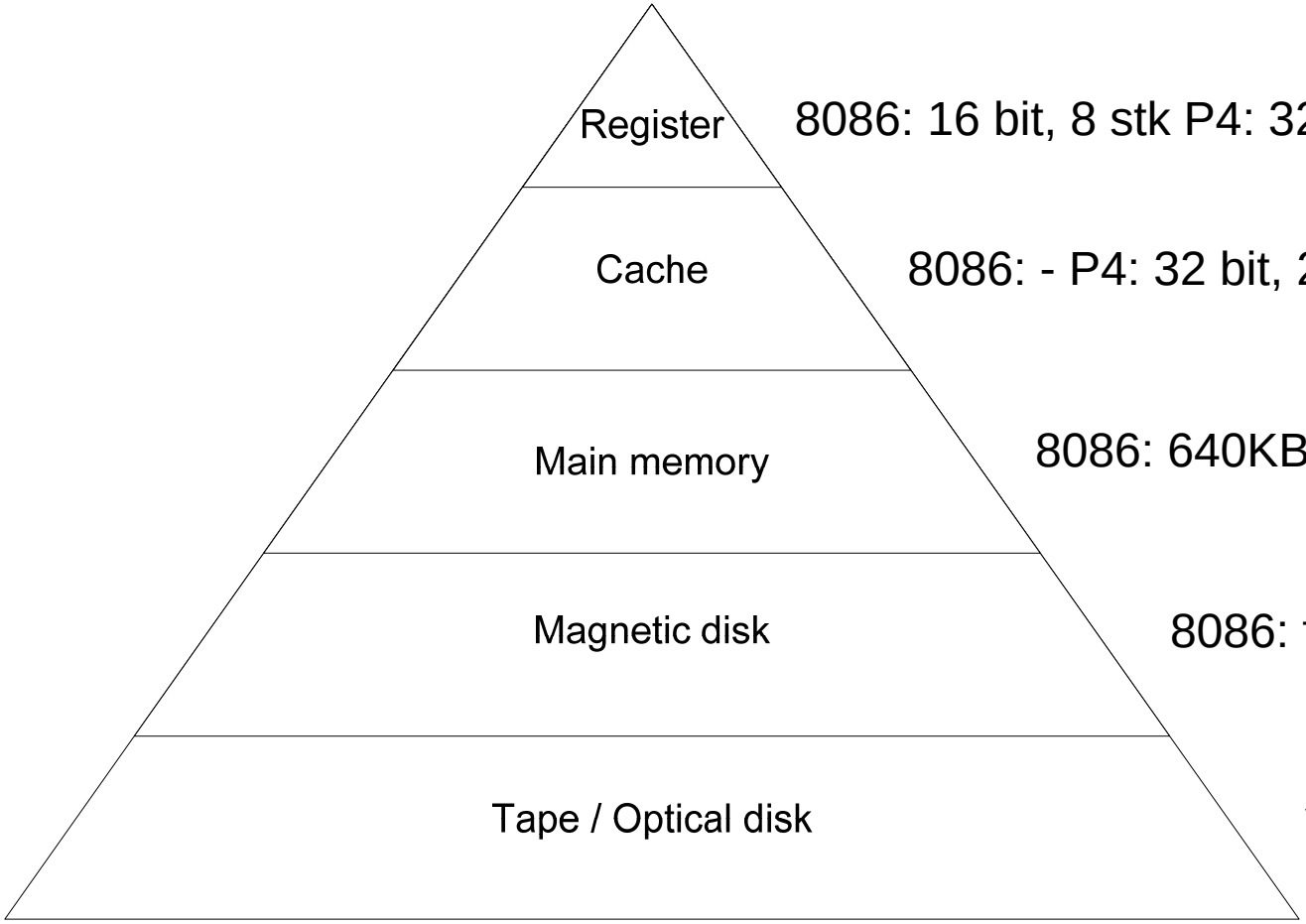


NTNU
Innovation and Creativity

Lagerhierarki



Lagerhierarki



8086: 16 bit, 8 stk P4: 32 bit, 128 stk

8086: - P4: 32 bit, 28kB L1 + 1024 kB L2

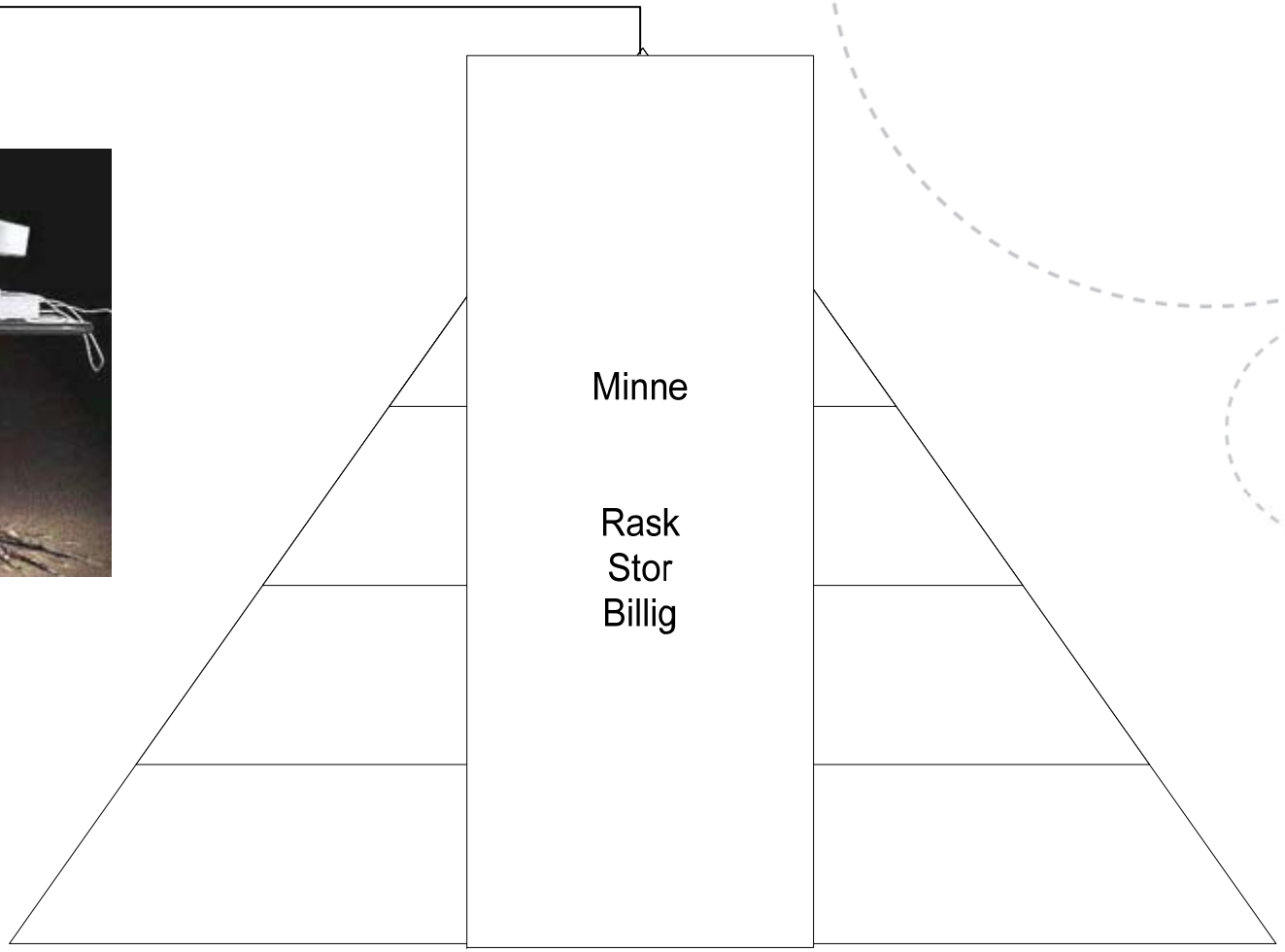
8086: 640KB P4: 4GB

8086: floppy 128kB P4: >100GB

Vekst > dobling kvart år

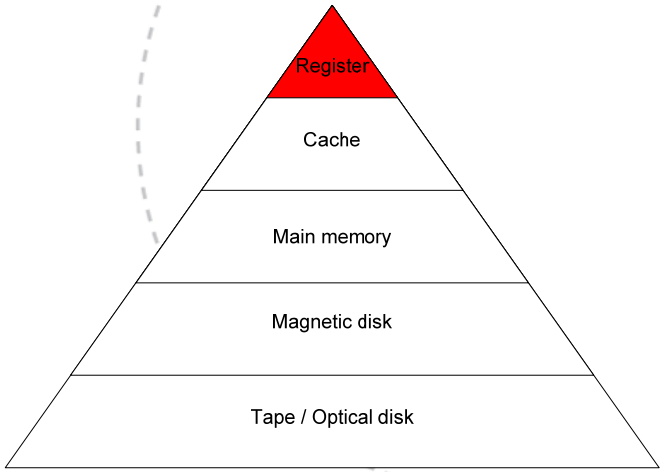
NTNU
Innovation and Creativity

Lagerhierarki

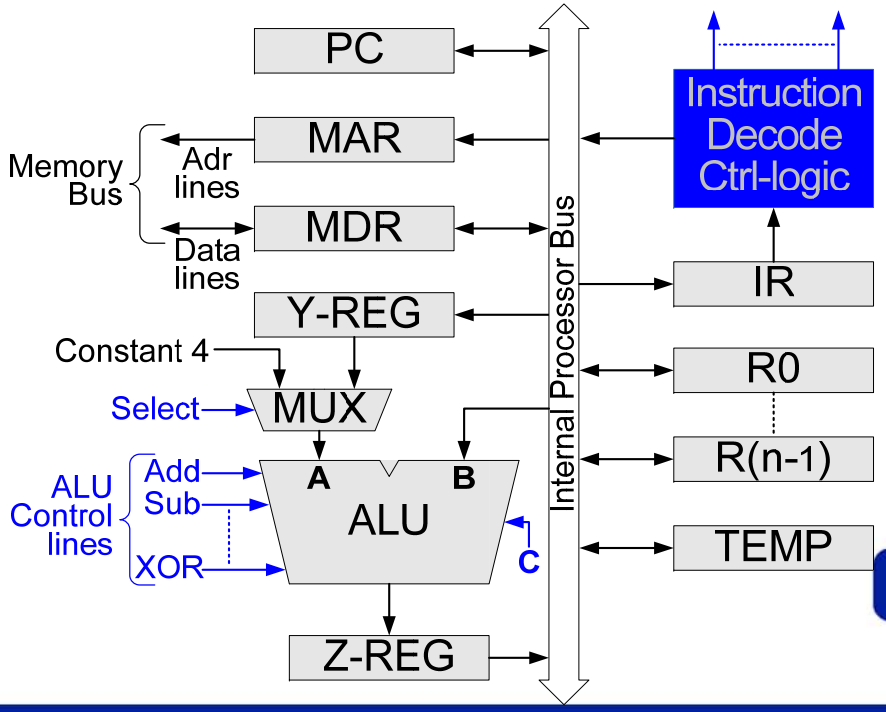


NTNU
Innovation and Creativity

Register

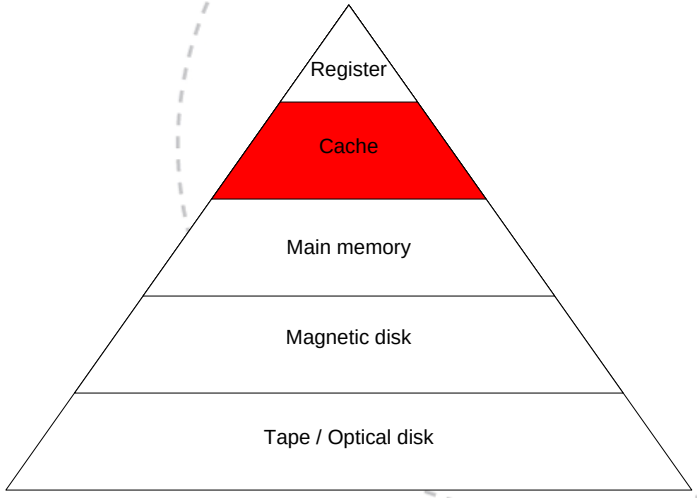


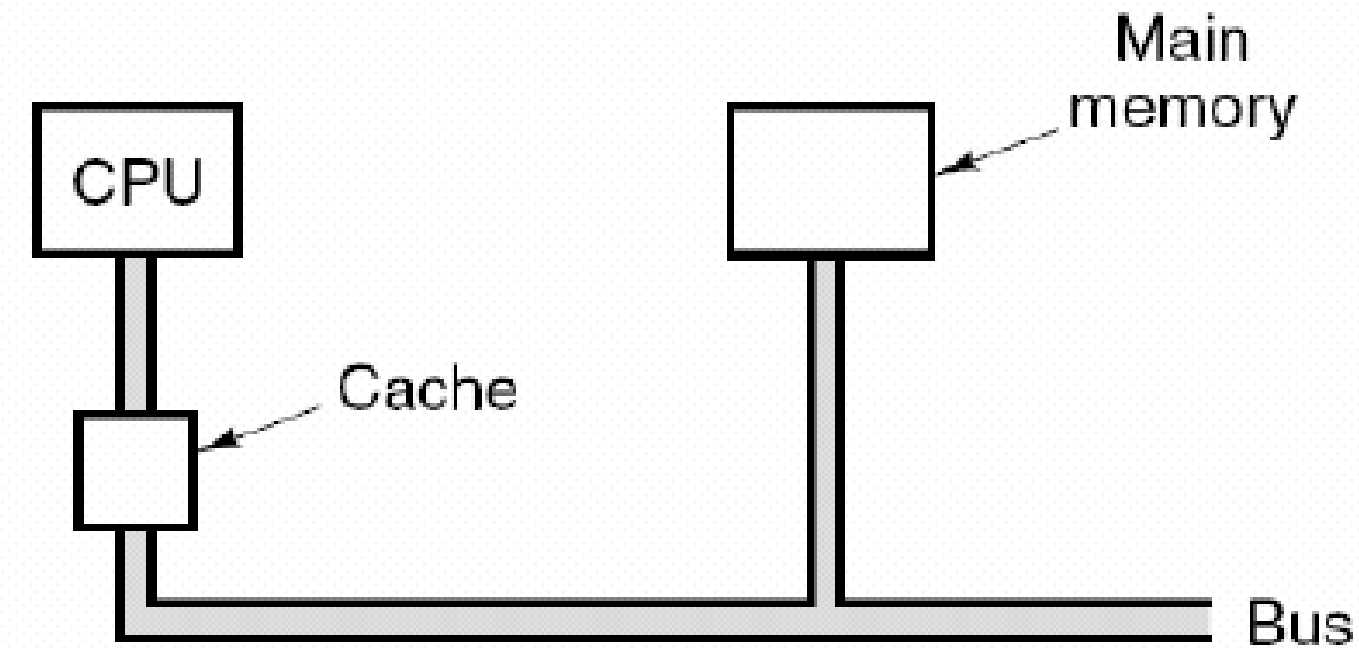
- Register:
  - Register i prosessor (PC, IR osv)
  - Registerbank
  - Data som prosessoren manipulerar
  - All data må inn i register
  - Raskast 0.25 ns



NTNU
Innovation and Creativity

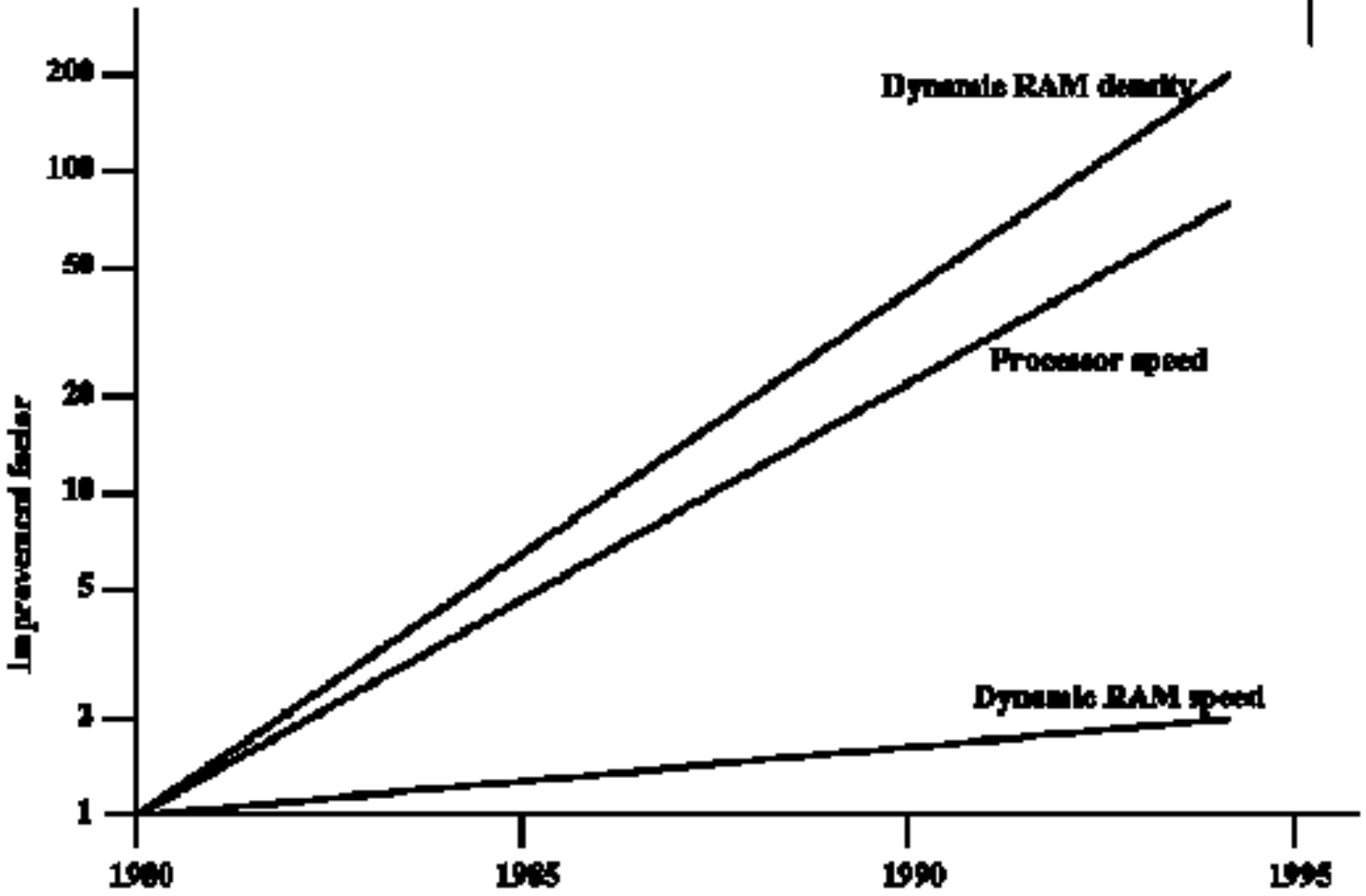
Hurtigbuffer (”cache”):





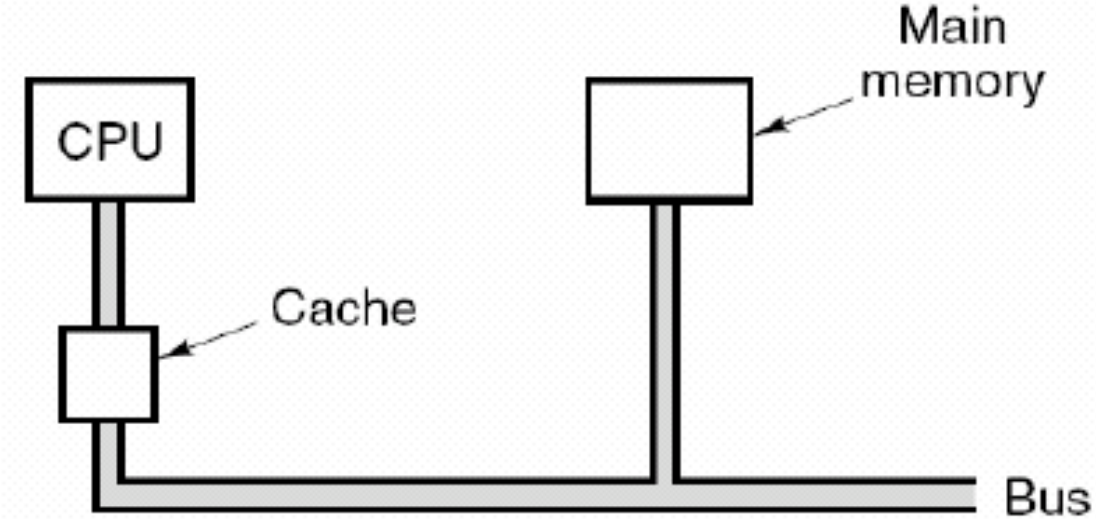
NTNU
Innovation and Creativity

Kvifor Hurtigbuffer



NTNU
Innovation and Creativity

Korleis Hurtigbuffer



- Leggast typisk på prosessor-brikke (Moores lov)
  - Hurtig!
  - Raskare men dyrare minneteknologi
  - Kortare transportavstand for data
  - Liten størrelse gir raskt oppslag

NTNU
Innovation and Creativity

Hurtigbuffer Lokalitetsprinsippet

- Hurtigbuffer
  - Størrelse gjerne 0,1% av hovedlager
  - Likevel har hurtigbuffer ofte 95% treffrate
- Korleis er dette mulig?
  - Lager aksesseres **ikkje** i et tilfeldig mønster
  - Kan forutse framtidige lageraksesser
- Lokalitet i rom:
  - Hentar me frå adresse 100 er det sannsynlig at vi
  - snart kommer til å hente frå adresse 101
- Instruksjonar: Sekvensiell utføring
  - Hopp instruksjonar bryt sekvensiell utføring
- Data: Ofte lagra i blokker (tabellar, bilde)

NTNU
Innovation and Creativity

Hurtigbuffer Lokalitetsprinsippet

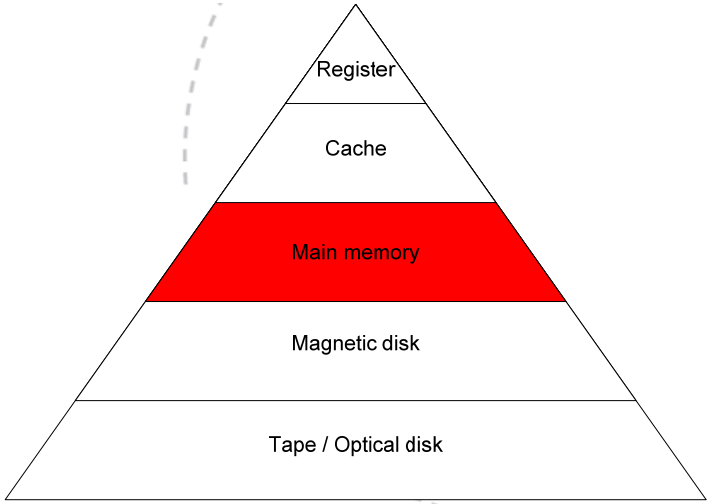
- Lokalitet i tid:
  - Hentar vi frå adresse 100 nå, er det sannsynlig at vi snart vil lese fra samme adresse
    - Instruksjonar: Løkker, subrutiner, ...
    - Data: Ofte brukte variablar, ...
- Dermed: Hentes noe frå hovedlager...
  - Legg det i hurtigbuffer
  - Hent også data frå etterfølgjande adresse(r) til hurtigb.
    - Mellom prosessor og hurtigbuffer: Byte/Ord
    - Mellom hurtigbuffer og hovedlager: Hurtigbufferlinjer (>ord)
- Sannsynlig at det vi trenger er i hurtigbuffer

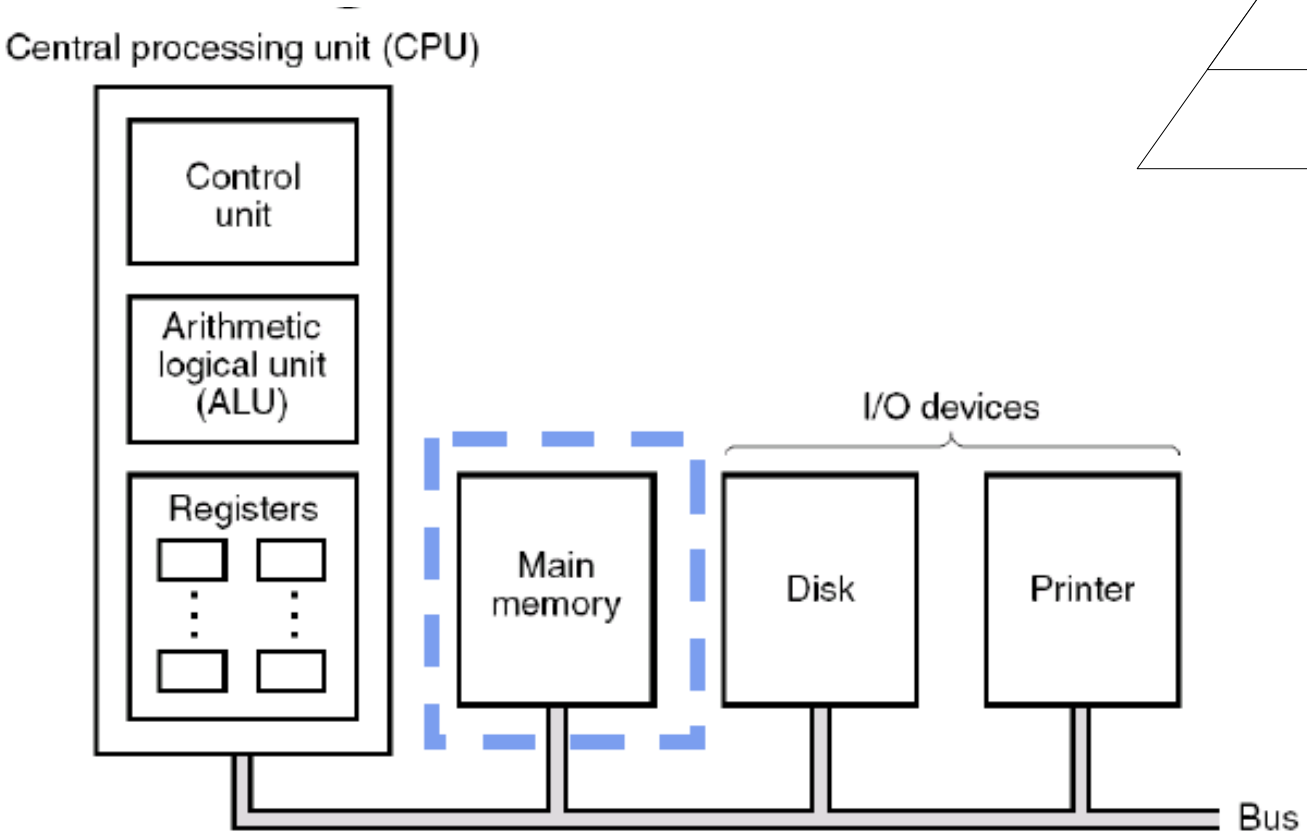
Hurtigbuffer Aksesstid

- Anta:
  - Hovedlager: 50 ns. aksesstid
  - Hurtigbuffer: 1 ns. aksesstid
  - 95% av data som trengs, finnes i hurtigbuffer
- Då får me:
  - Aksesstid = 1 ns. + 0,05 * 50 ns. = 3,5 ns.

Billig, stort lager med god yting

Hovudlager

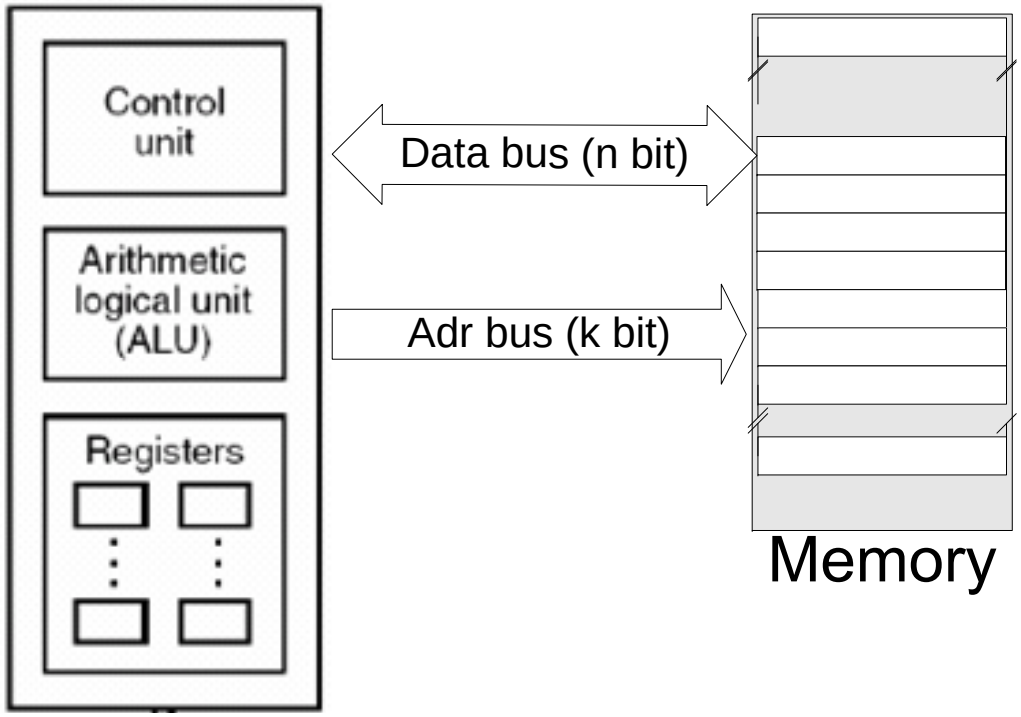

Register
Cache
Main memory
Magnetic disk
Tape / Optical disk


Central processing unit (CPU)
Control unit
Arithmetic logical unit (ALU)
Registers
I/O devices
Main memory
Disk
Printer
Bus

Hovedlageradresser

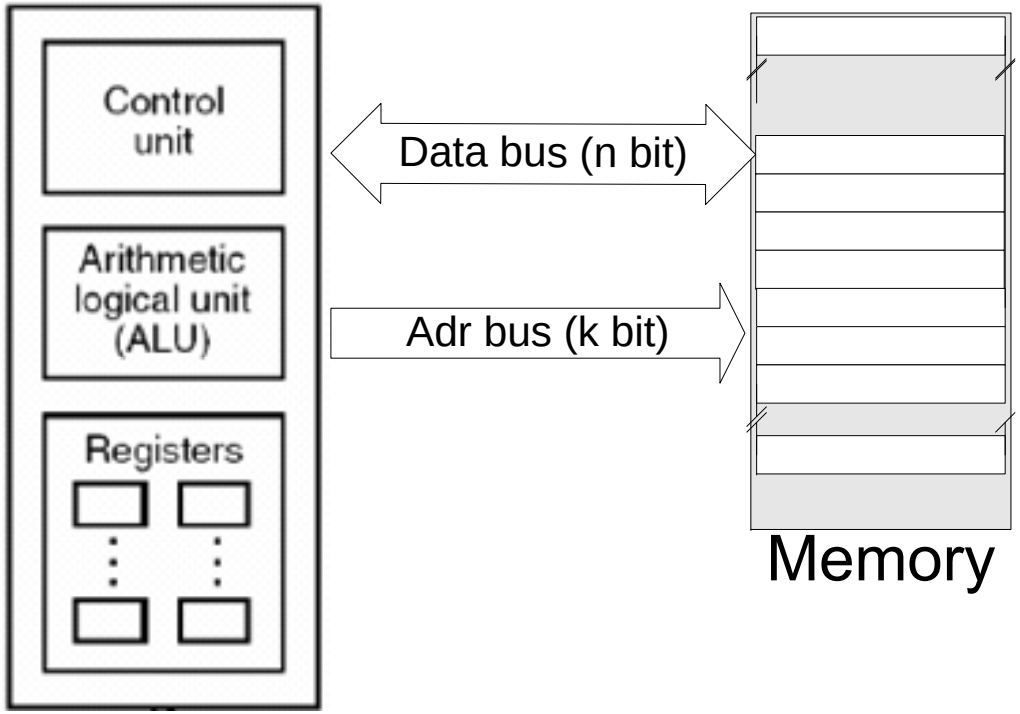
- Hovedlager organisert i celler
  - Kvar celle har en unik adresse
  - Med *k* bits kan vi adressere 2*k* celler
- Celler normalt 1 byte (=byteadresserbart lager)
  - 1 byte = 8 bit
- Bytes gruppert i ord, typisk 32/64-bits ord
  - Instruksjonar arbeider typisk på ord
  - Registerstørrelse typisk lik ordstørrelse
- 32 bit Maks 4GB hovedlager (byteadress.)

Hovedlageradresser

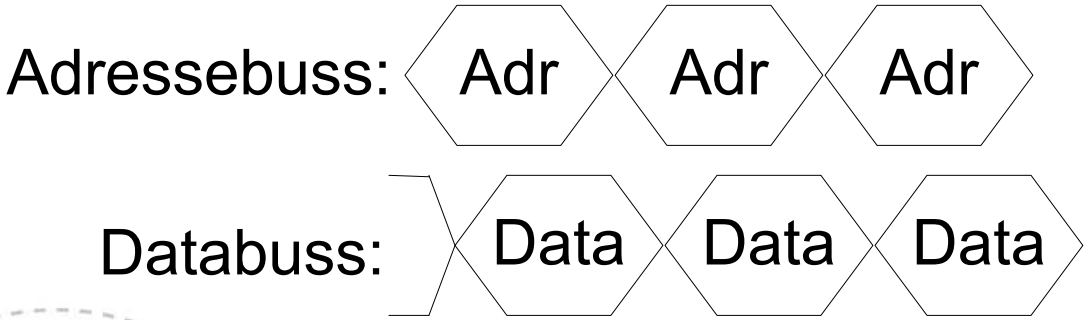


- Adressebuss
  - k bit 2^k adresserlokasjonar
- Databuss
  - n bit kvar lese/skrive operasjon
- Eksempel
  - 8 bit databuss
  - 4 lese/skrive operasjonar for 32 bit ordlengde

Hovedlageradresser: Buss

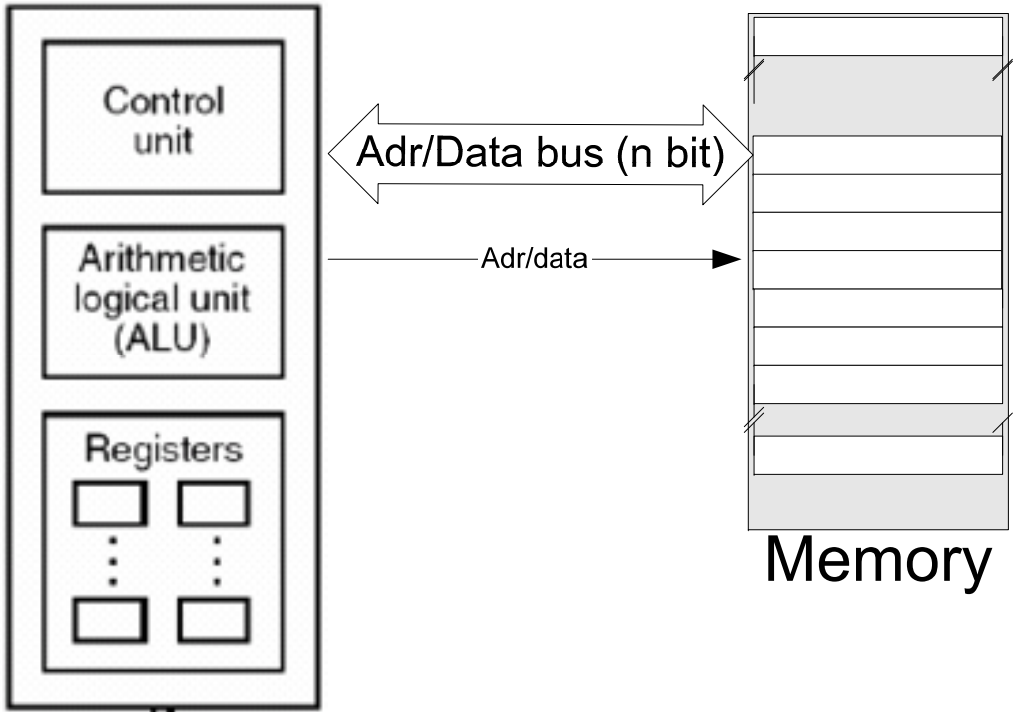


- Separat adressebuss og databuss

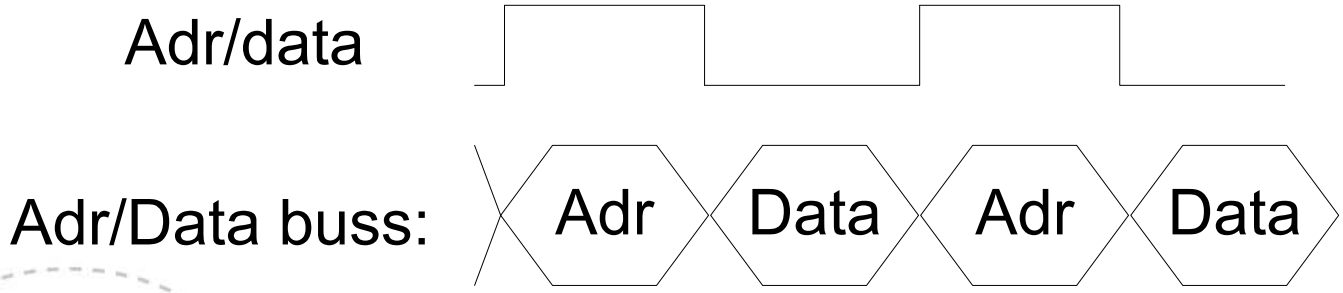


NTNU
Innovation and Creativity

Hovedlageradresser: Buss

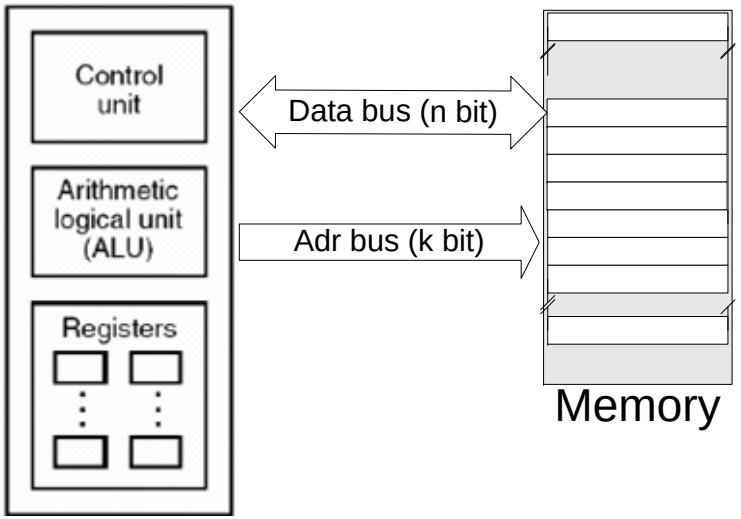


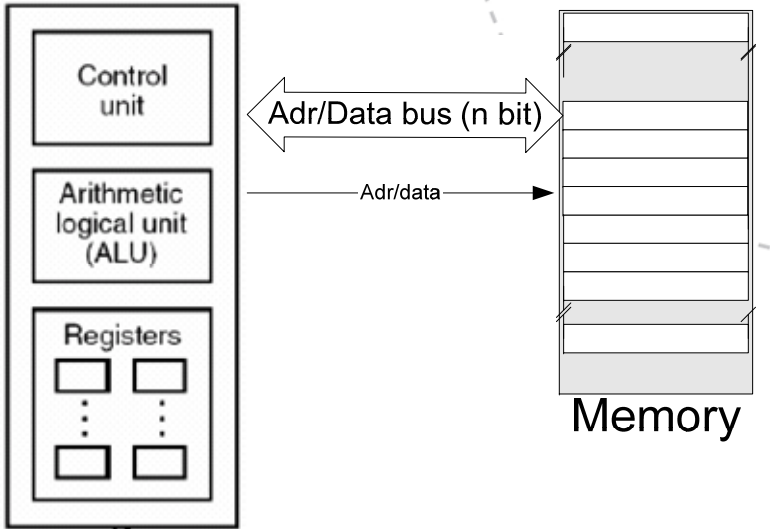
- Delt buss for adresse og data

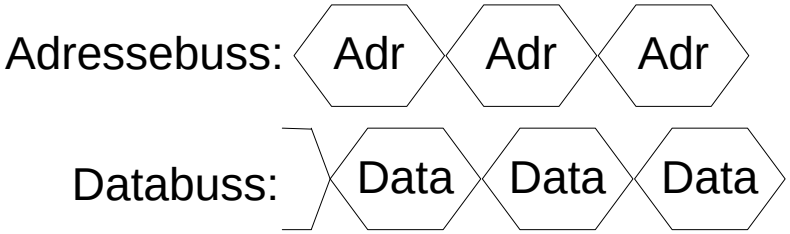


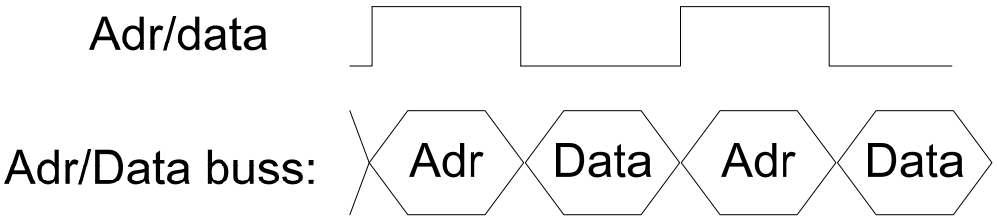
NTNU
Innovation and Creativity

Hovedlageradresser: Buss



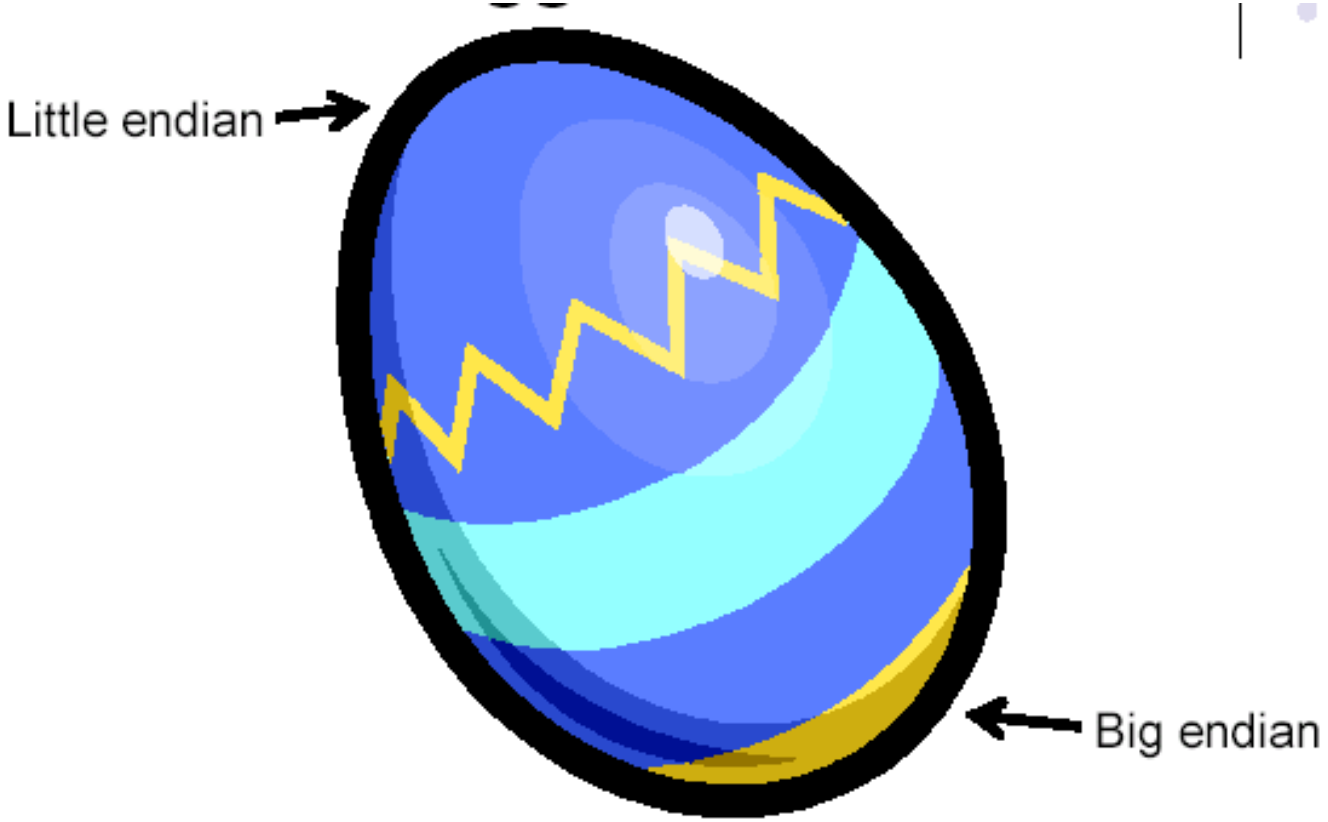






Dataorganisering i minne

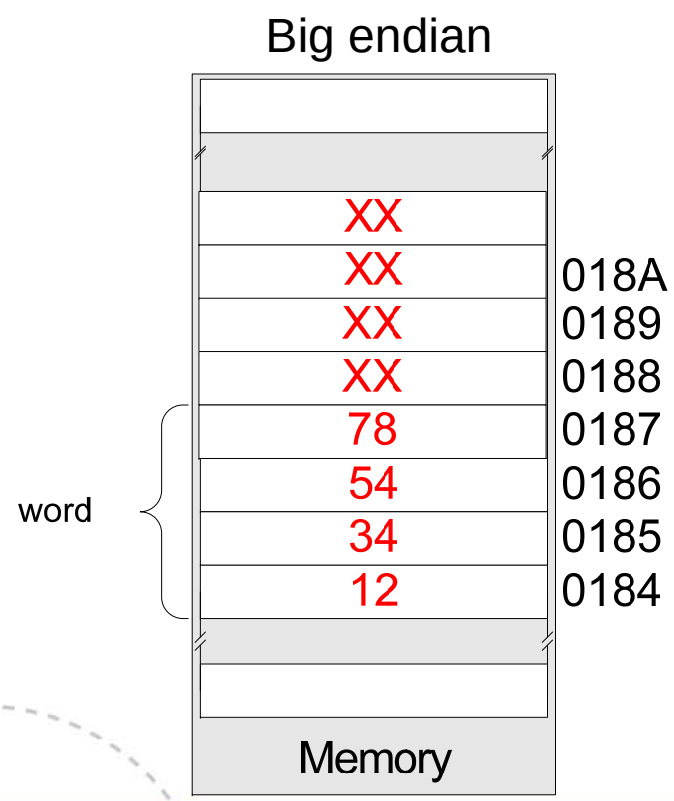
- Kvar skal kuttar ein egget

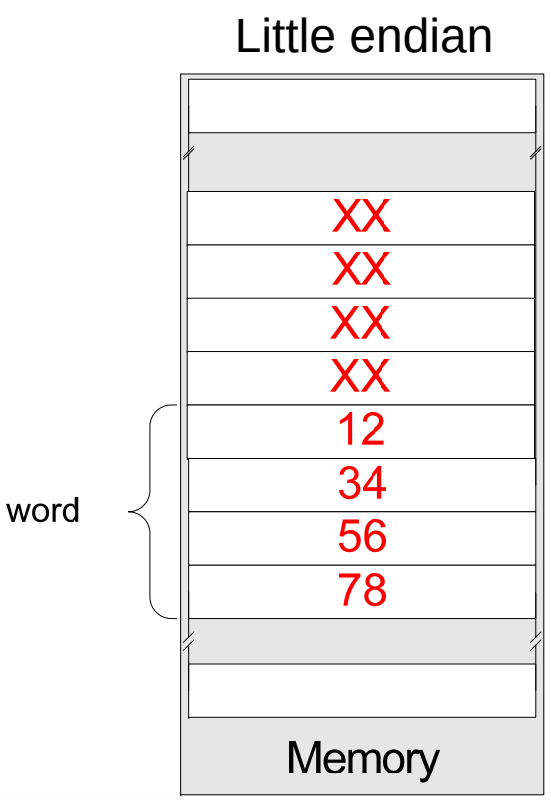


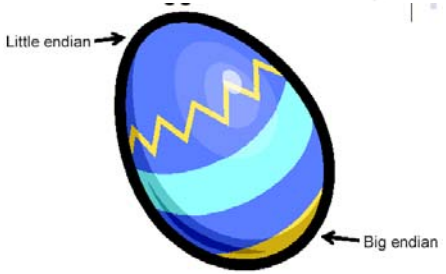
NTNU
Innovation and Creativity

Dataorganisering: Byte-rekkefølge

- Ord på meir enn en byte kan lagrast på to måtar...
- Eksempel: **0x12 34 56 78** (32 bits/4 bytes hex)
- Viktig: Adressa til *ordet* er uendra!



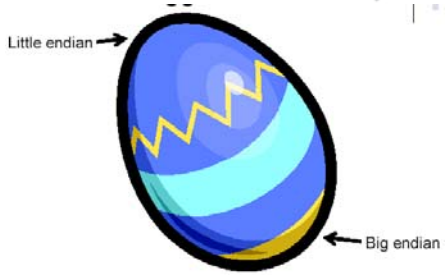




NTNU
Innovation and Creativity

Dataorganisering: Byte-rekkefølge

Address	Big endian				Little endian				Address
0	0	1	2	3	3	2	1	0	0
4	4	5	6	7	7	6	5	4	4
8	8	9	10	11	11	10	9	8	8
12	12	13	14	15	15	14	13	12	12
				Byte				Byte	
				32-bit word				32-bit word	

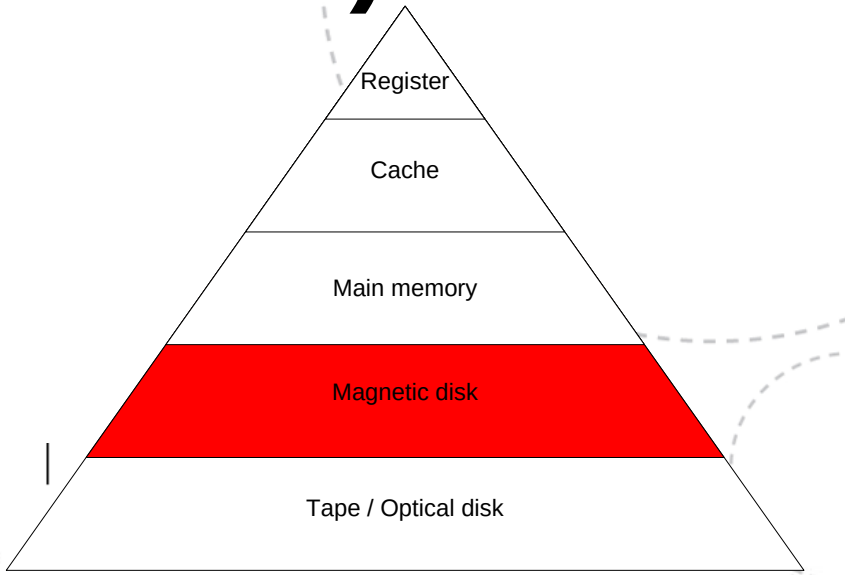


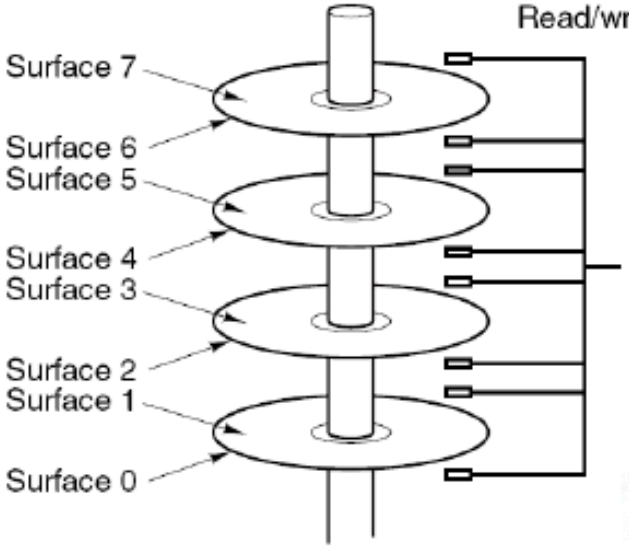
	Big endian				Little endian				
0	J	I	M			M	I	J	0
4	S	M	I	T	T	I	M	S	4
8	H	0	0	0	0	0	0	H	8
12	0	0	0	21	0	0	0	21	12
16	0	0	1	4	0	0	1	4	16

NTNU
Innovation and Creativity

Magnetiskdisk (Harddisk)

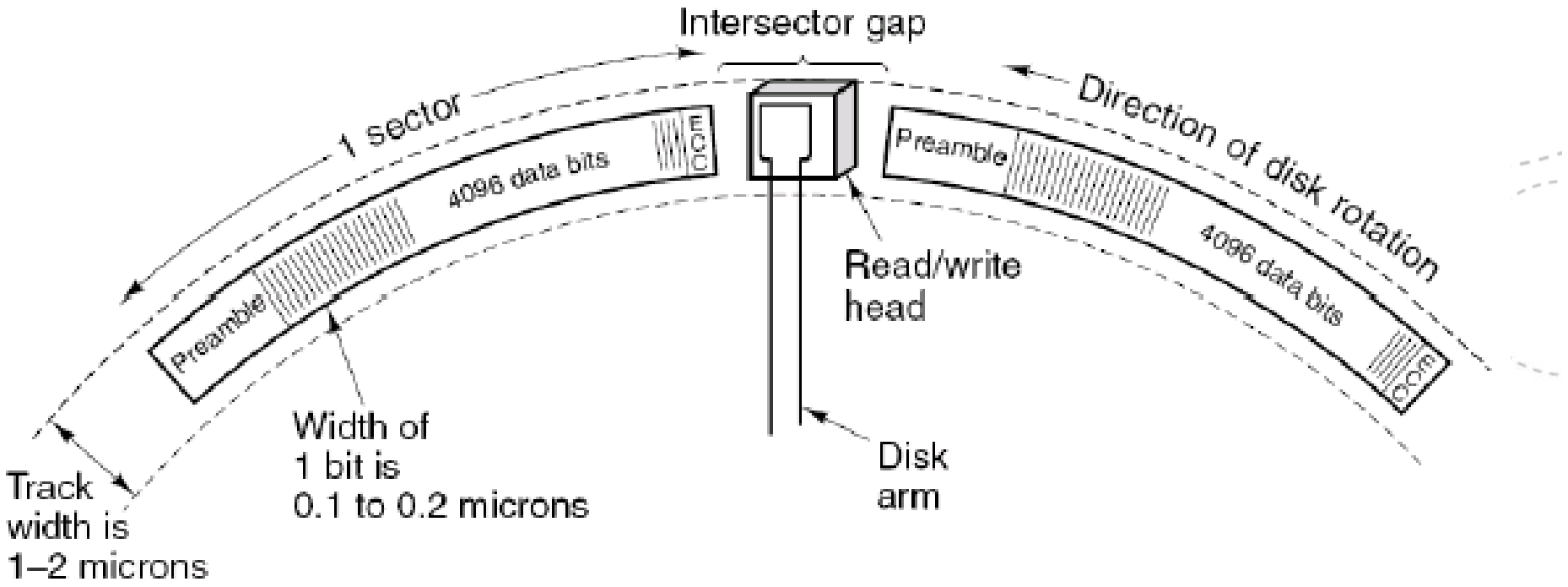
- Harddisk
  - Stor lagringskapasitet
  - Ikkje rask (ms)





NTNU
Innovation and Creativity

Magnetiskdisk (Harddisk)



Magnetiskdisk (Harddisk)

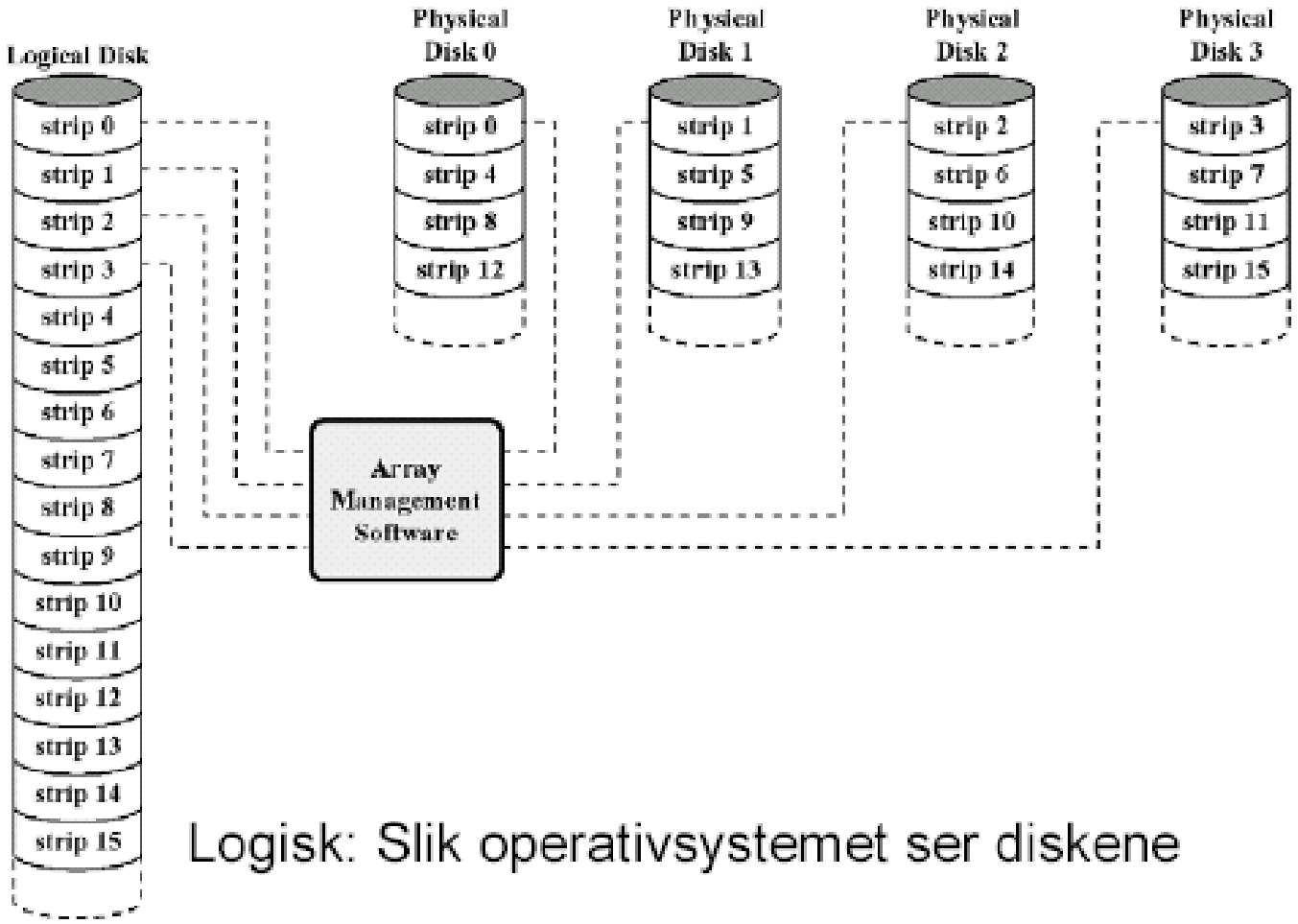
- Kva brukar ein disk tid på?
  - Søketid – flytt arm
  - Rotasjonstid
  - Overføringstid
  - Kontrollertid
- Hva er best nå?
  - Aksesstid = 5.4 ms (målt)
  - Maximum Transfer Rate 97 MB/s (ytterst)
  - Minimum Transfer Rate 74 MB/s (innerst)
- (www.storagereview.com)

Harddisk RAID

- **RAID: Redundant Array of Independent disks**

- Problem: Harddisker ikkje raske nok
  - Moores lov hjelper ikkje
  - Ofte fysiske/mekaniske avgrensingar
- Løysning: Fleire diskar som nyttast i parallell
- Fordeler:
  - Høgare yting:
    - Parallell aksess (fleire filer samtidig)
    - Parallell dataoverføring (samarbeid om 1 fil)
  - Redundans kan brukast for å oppnå feiltoleranse

Harddisk RAID

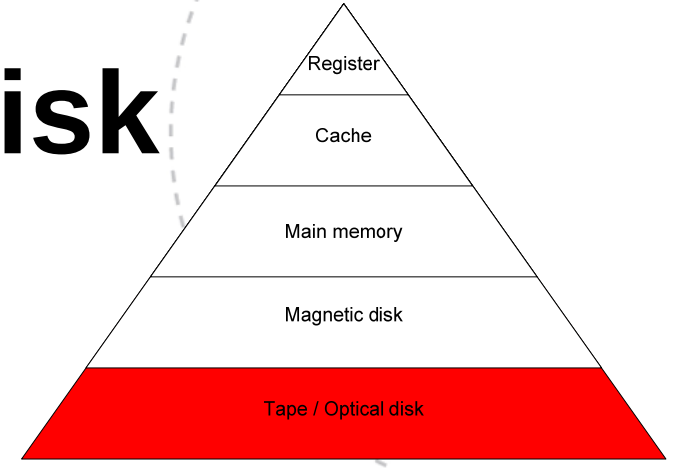


Logisk: Slik operativsystemet ser diskene

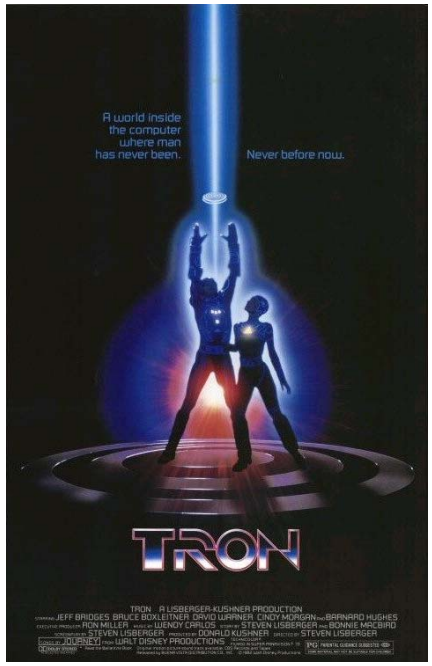
Harddisk RAID

Mange måtar å organisere eit RAID på, les i boka

Masselager Tape Optisk



- Lagre store datamengder
  - Sikkerheitskopi
  - Data som ikkje er i bruk
  - Data som ikkje krev kort aksesstid



NTNU
Innovation and Creativity

Lagerhirarki