

TDT4160

Datamaskiner Grunnkurs

2008

Gunnar Tufte

Dagens forelesing

- Kapittel 1
 - Datamaskinsystem
- Kapittel 2
 - start

Gunnar Fakta

Datamaskingruppa

Biologisk inspirerte system:

Unconventional Computing Machines

Artificial Development

Artificial Evolution

<http://crab.idi.ntnu.no/>

<http://www.idi.ntnu.no/~gunnart/>

Er også med i Soundscape Studios (eigar og arbeidar)

Elektronikk, data, kunst osv

<http://www.soundscape-studios.no>

Har dysleksi, prøv å ikkje bli for iritert på skrivvve feil

TDT4160 2008 Fagstab

- Førelasar: Gunnar Tufte
Rom 406 IT-vest

Vit. ass.: Bjørn Magnus Mathisen
Rom 445 IT-vest
Øvingar
AOC
Dei fleste spørsmål



Guru: Magnus Jahre
Rom 445 IT-vest



+ mange flinke student assistentar

TDT4160 2008 Førellesningar

- | Dag | Tid | Kvar | Kva |
|-----------|---------------|------|------------|
| • Onsdag | 10.15 – 12.00 | S5 | Forelesing |
| • Torsdag | 13.15 – 14.00 | S5 | Forelesing |
- Øvingar info om AOC kjem torsdag 28/8 13.15 (Bjørn magnus) S5
 - AoC opnar etter forelesingen

TDT4160 2008 Øvingar

- **Age of Computers**
- Dataspill-inspirert øvingsopplegg
- Fleirspelar, nettbasert, chat
- Løys oppgåver, lær historikk, lær fakta
- Inneheldr assembly-simulator (Dark)
- Ikkje 100 % match mot rekkefølgjen i læreboka

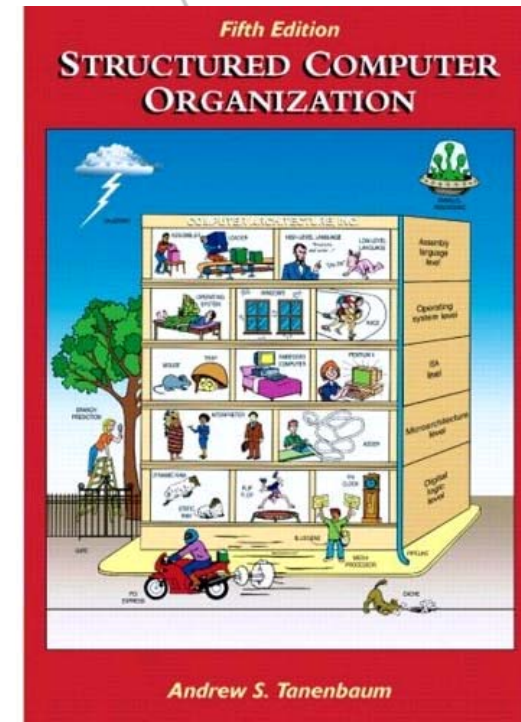
- Demo torsdag 28/8 13.15 S5 ved Bjørn Magnus

- <http://aoc.idi.ntnu.no>

- Obligatorisk

TDT4160 2008 Pensum (ikkje endeleg)

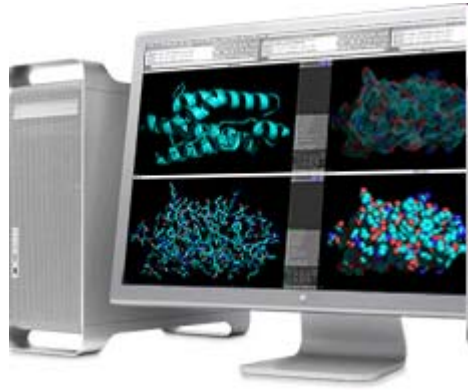
- Tanenbaum: Structured Computer Organization
- Kapittel 1,2
- Kapittel 3 unntatt 3.1 og 3.7
- Kapittel 4
- Kapittel 5 unntatt 5.7
- Appendiks B
- Alt frå forelesingane
- Øvingsopplegg (Age of Computers)
- Foiler fra forelesningene skal liggje på It's learning



Datamaskinsystemer

- Tradisjonelle datamaskiner
 - PC, arbeidsstasjoner, superdatamaskiner osv.
- ”Embedded systems” (innebygde systemer)
 - Mobiltelefoner
 - Digitale kamera
 - Navigasjonssystemer (GPS, digitale kart)
 - Laserskrivere, kopimaskiner
 - Bilar
 - Forskjellige maskiner
 - Kvitevarer osv.

Datamaskinsystemer



Tradisjonelle datamaskiner

- PCer, arbeidsstasjoner, superdatamaskiner osv.
- Skal sjå på: Intel Pentium 4 og Sun UltraSPARC III



”Embedded systems” (innebygde systemer)

- Mobiltelefoner, kopimaskiner, kvitevarer, osv
- Skal sjå på Intel 8051

Historie

Mekaniske datamaskiner

1642 – 1945

Radiatorer

1945 – 1955

Frå her stort sett forbedringar av von Neumann arkitektur

Transistorer

1955 – 1965

Integrerte kretser

1968 – 1980

VLSI – Very Large Scale Integration

1980 –

Usynlige datamaskiner

Historie

Mekaniske datamaskiner

1642 – 1945

Radorør

1945 – 1955

Frå her stort sett forbedringar av von Neumann arkitektur

Transistorer

1955 – 1965

Integrerte kretser

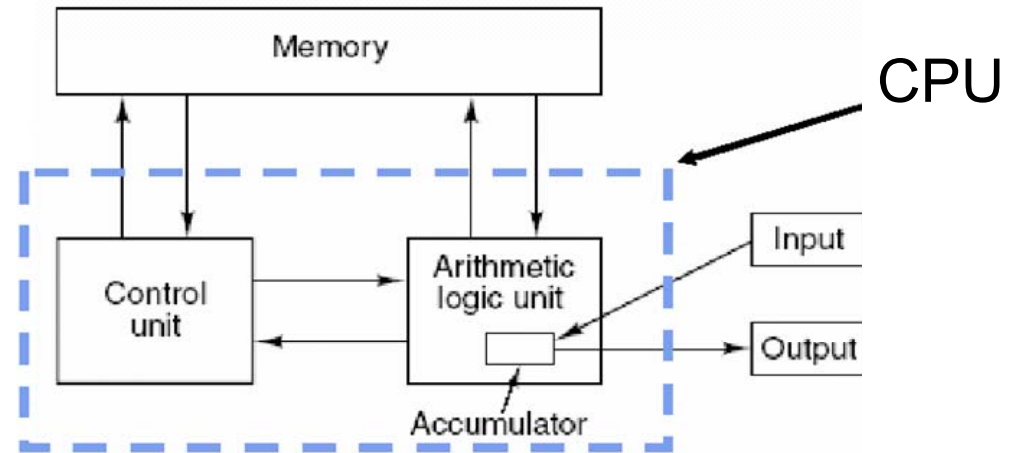
1968 – 1980

VLSI – Very Large Scale Integration

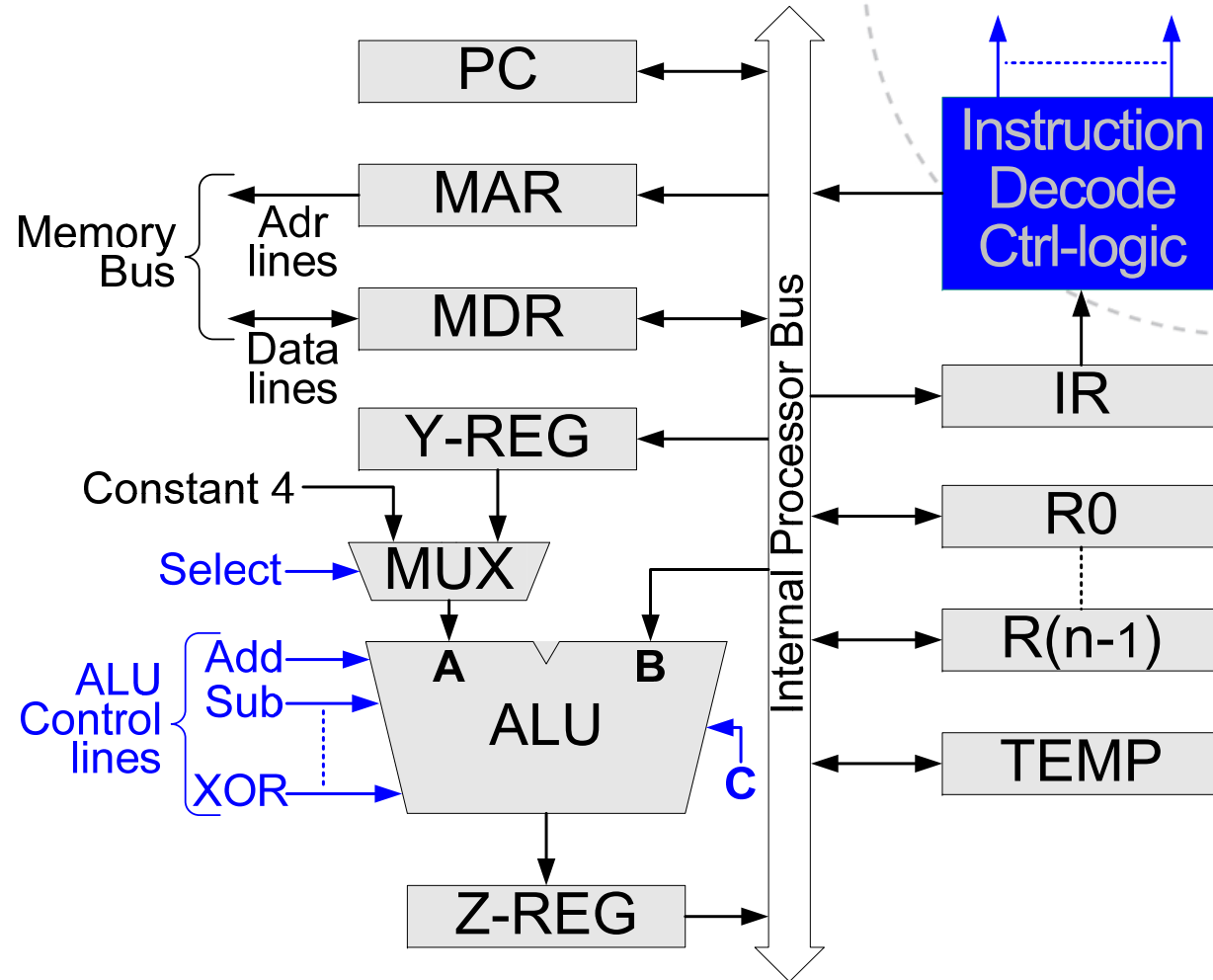
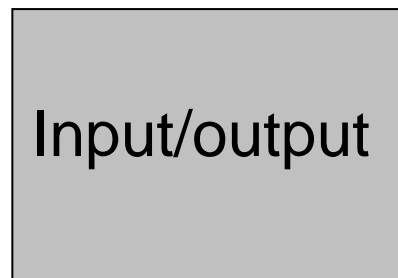
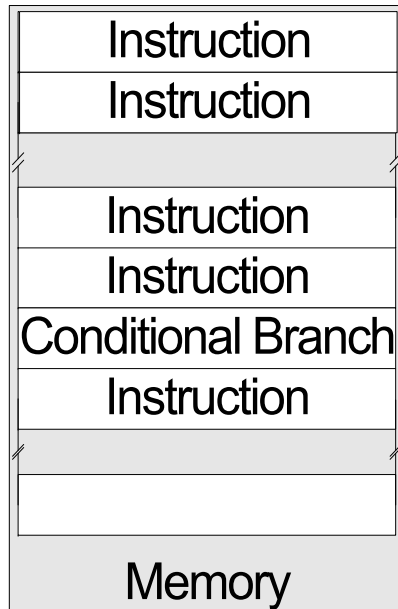
1980 –

Usynlige datamaskiner

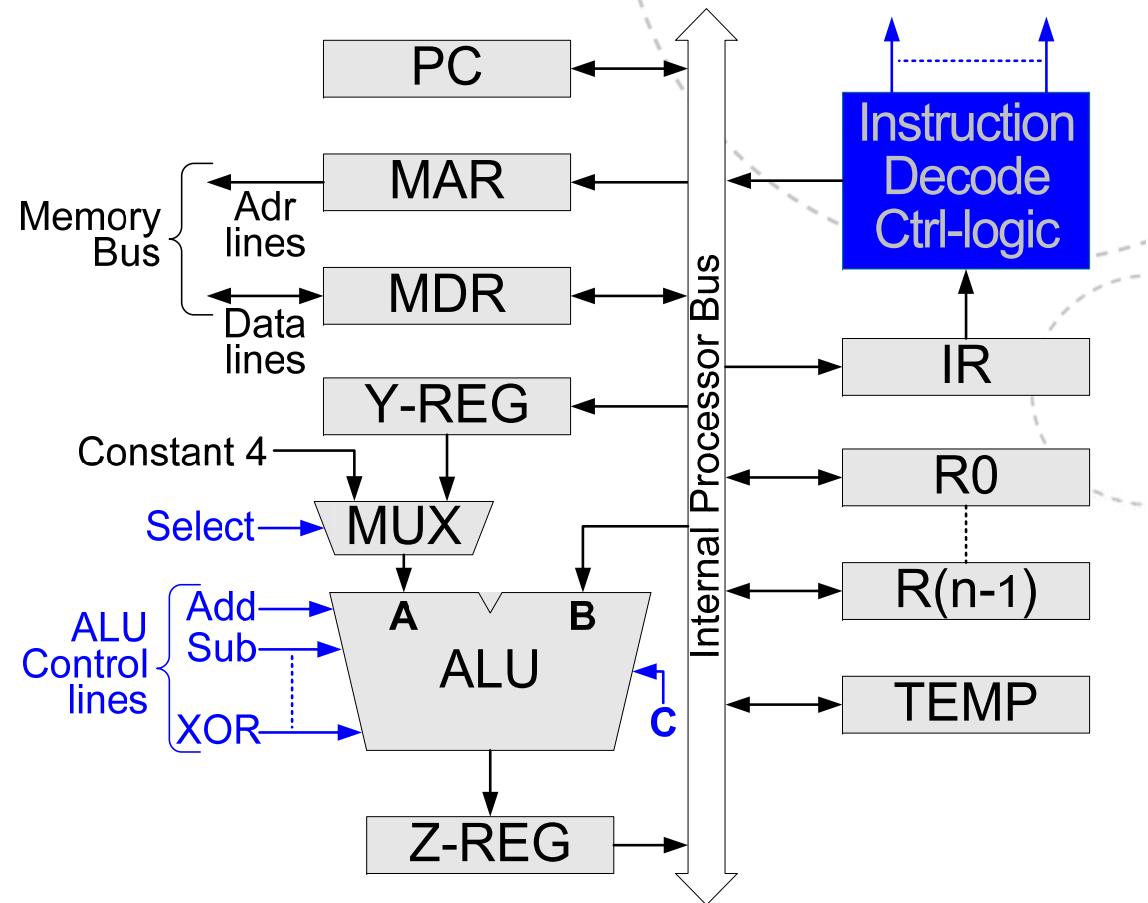
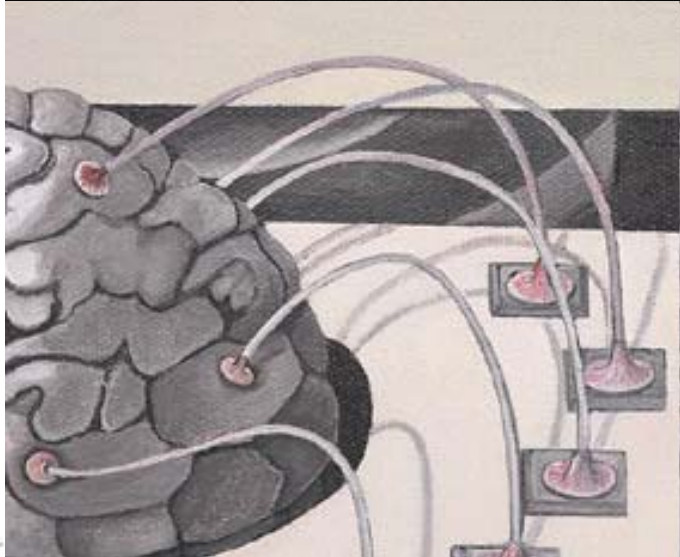
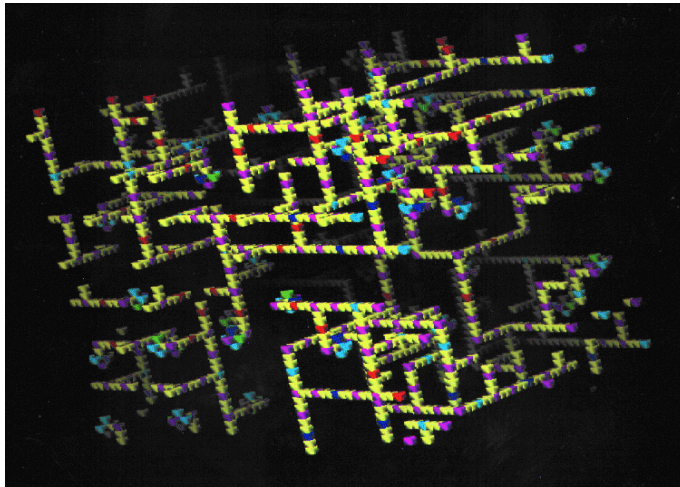
Von Neumann-arkitektur



Skal sjå på P4, SUN, 8051 etter kapittel 2

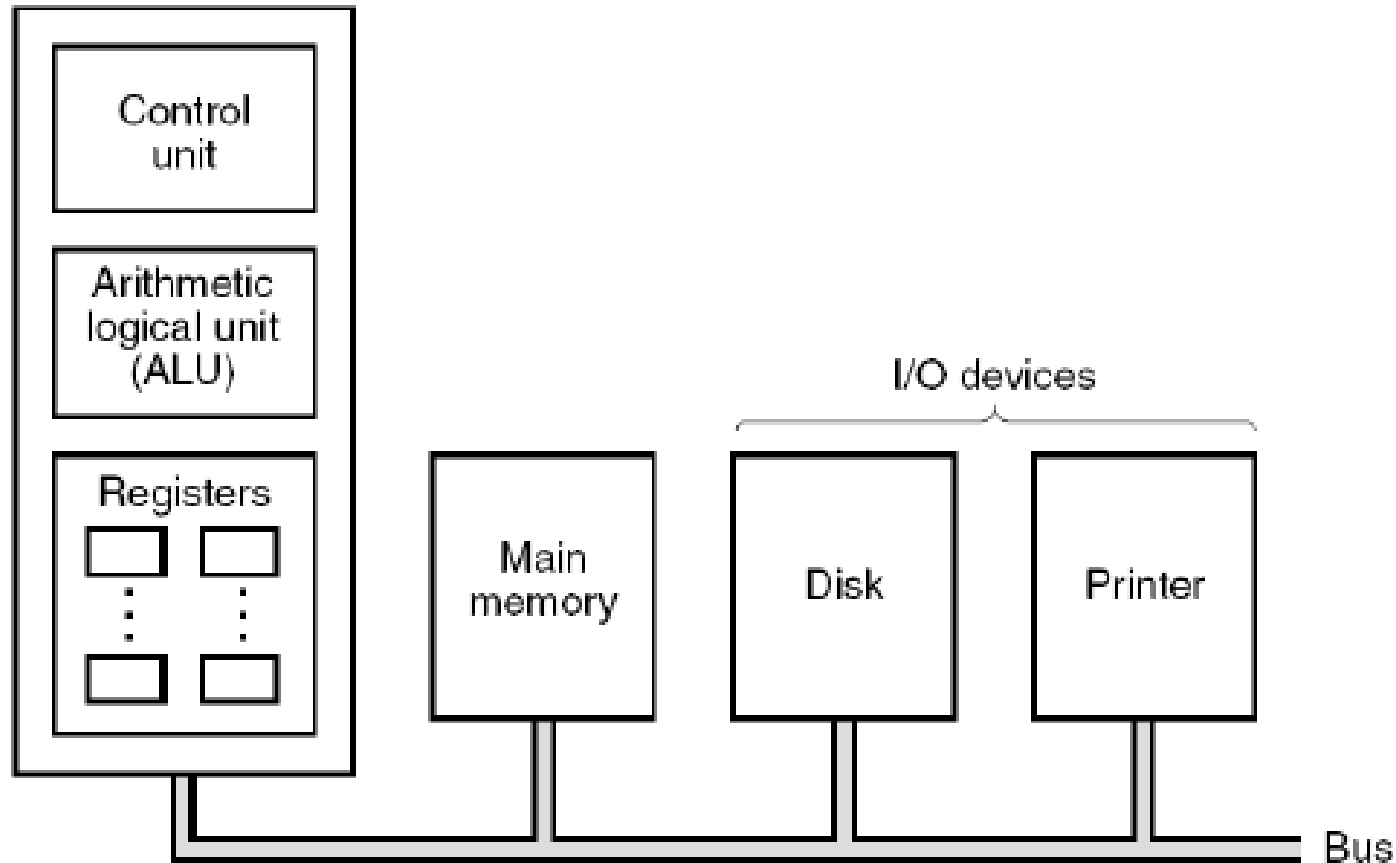


Kapittel 2: Processorar 2.1



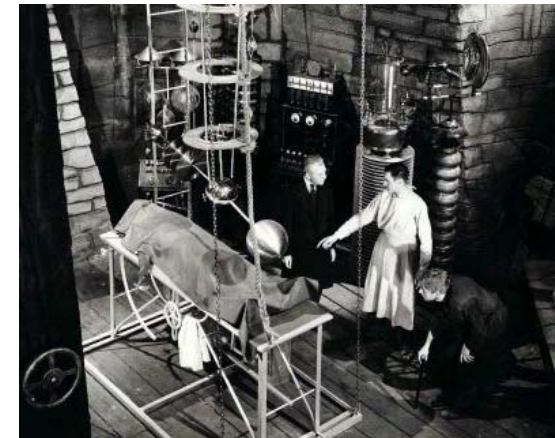
Overordna organisering

Central processing unit (CPU)

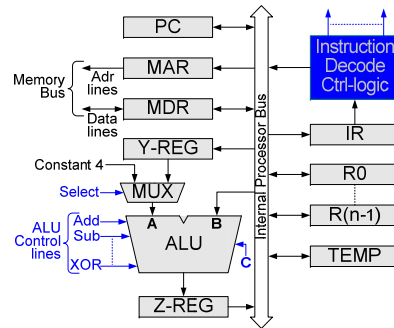
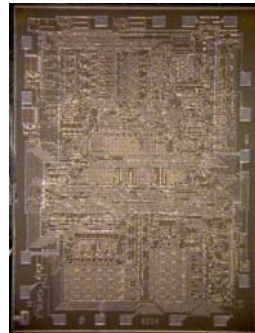
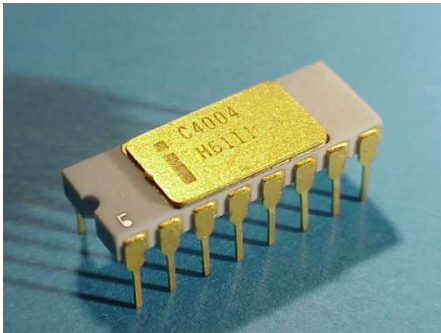


Kva er ein CPU

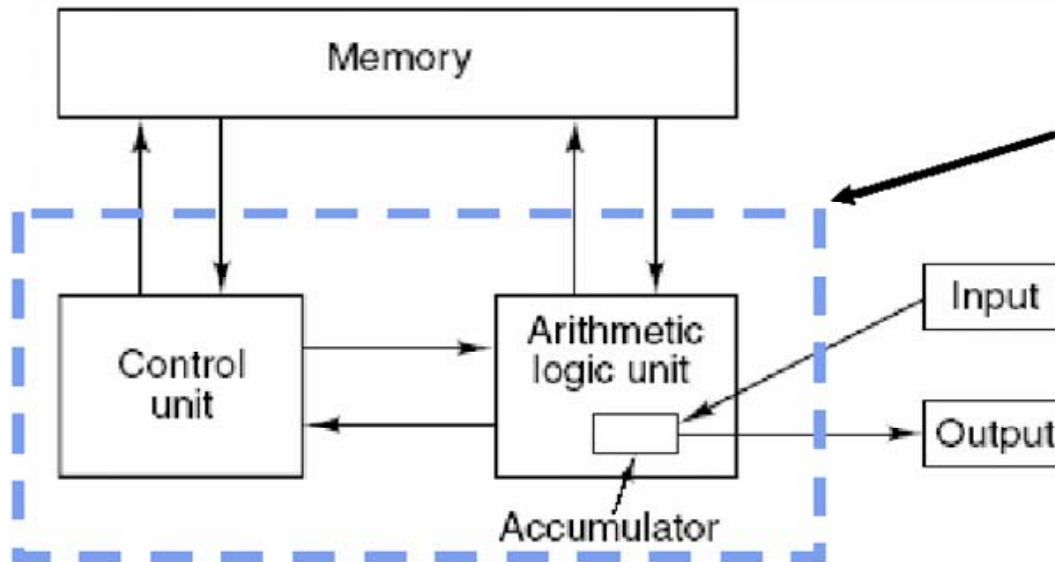
- "Hjernen" i ei datamaskin
 - Køyrrer program:
 - hentar instruksjoner
 - finner ut kva dei gjer
 - utførar
 - Addere
 - Logiske funksjonar (e.g. AND, OR, NOT)
 - Flytte data
 - Test (e.g. er eit tal null)
 - Hopp med vilkår (e.g. hoppe til ein anna plass i koden viss tal = 0)
- Hoveddeler
 - Styreeinheit (control unit)
 - Aritmetisk-logisk einheit (ALU)
- Registre
 - Programteller (PC) - Adresse til neste instr.
 - Instruksjonsregister (IR) - Instruksjon som utføres nå
 - Generelle heiltallsregistre, flyttallregistre.....



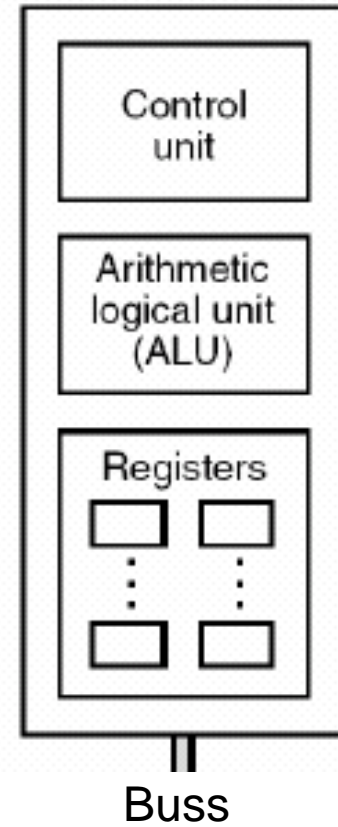
Processor (CPU)



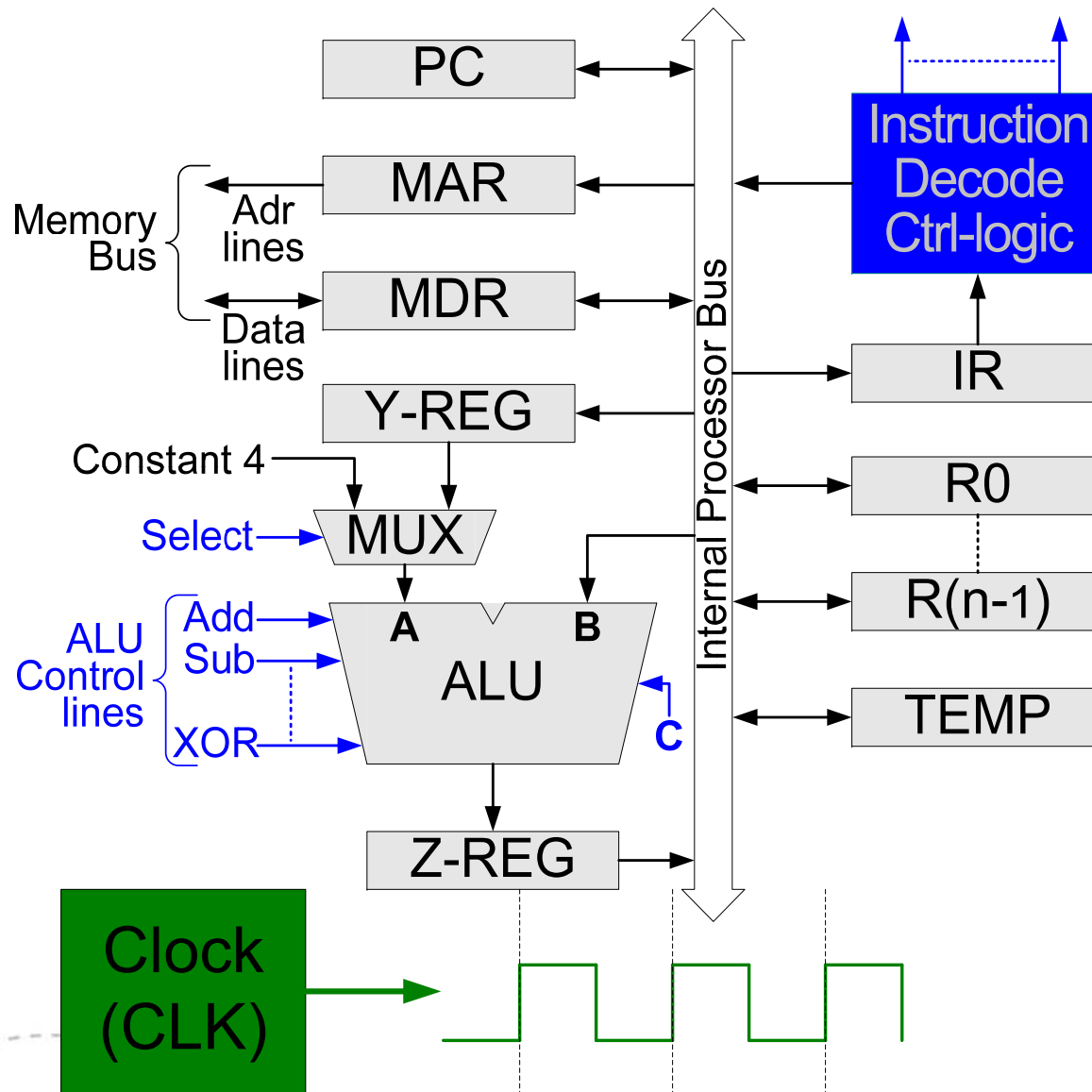
Von Neumann-arkitektur



CPU



Kva er inni ein CPU



- PC: Program Counter
- IR: Instruction Register
- MAR: Memory Address Register
- MDR: Memory Data Register
- ALU: Arithmetic Logic Unit

Korleis vert eit program utført

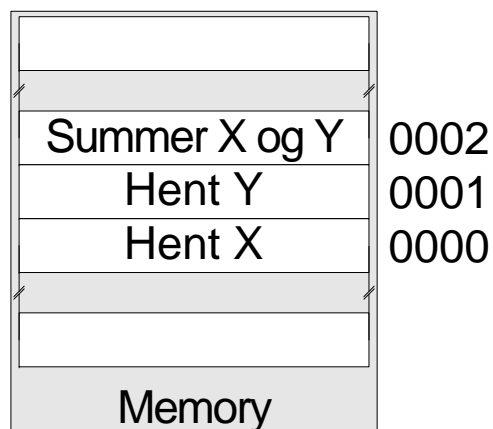
- Styre einheit gjer som følgjer:
 - Hent instruksjon (Fetch)
 - Dekodar instruksjonen (Decode)
 - Utfør instruksjonen (Execute)
- Utførande einheit gjer som følgjer:
 - Execute: brukar utførande einheit til å utføre instruksjonen

Liten avsporing: Program

- Frå forelesing 1 Ein prosessor kan:
 - Addere
 - Logiske funksjonar (e.g. AND, OR, NOT)
 - Flytte data
 - Test (e.g. er eit tal null)
 - Hopp med vilkår (e.g. hoppe til ein anna plass i koden viss tal = 0)
- Ein CPU består av
 - styreeinheit
 - utførandeeinheit
- Program
 - Ei rekkje av enkle instruksjonar
 - Kvar instruksjon utfører ein bit av ei oppgåve
 - Instruksjonar og data ligg i minne
 - Programminne
 - Dataminne
 - Programmet ligg i minne

Liten avsporing: Program 2

- Enkelt eksempel
 - Leggje saman to tal **X** og **Y** som ligg i minne
- Program:
 - Hent **X**
 - Hent **Y**
 - Summer **X** og **Y**

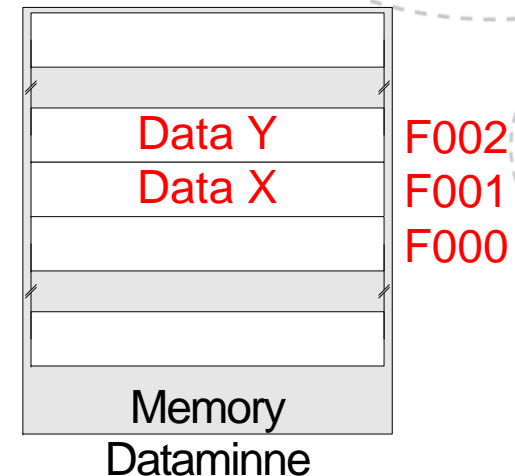
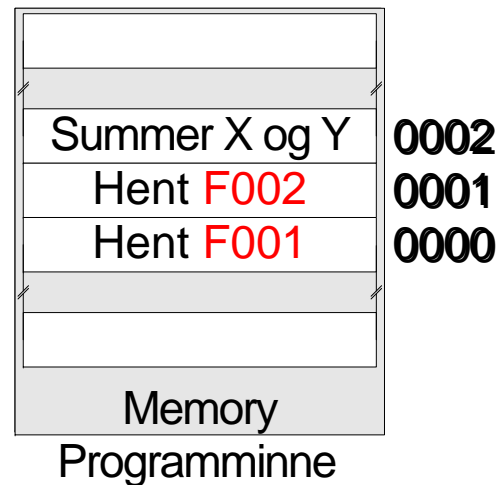


Liten avsporing: Program 3

- MEN program (instruksjonar) og data ligg i forskjellige deler av minne **Programminne** og **Dataminne** (sjølv om det kan være det same fysiskeminne)

- Program:

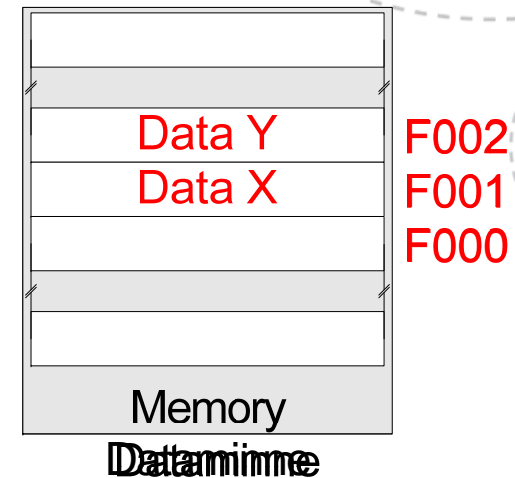
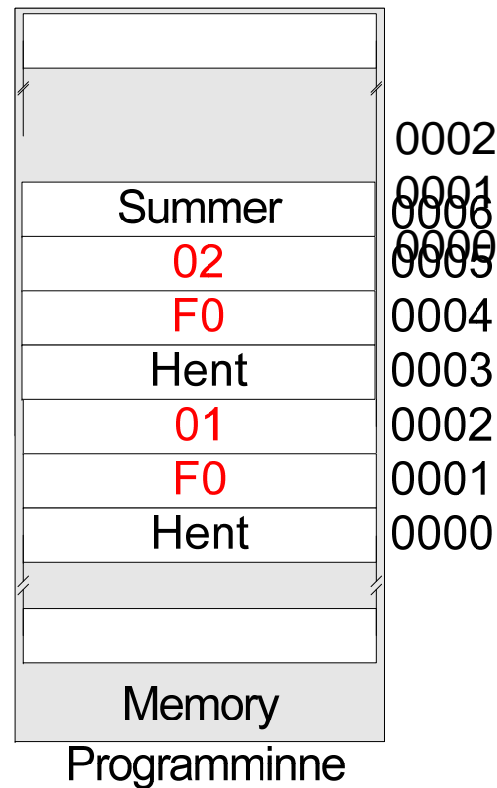
- Hent X
- Hent Y
- Summer X og Y



- I programminne ligg det peikar (pointer) til data
 - Data (variablar) får si minne adresse av kompilatoren
 - Peikar på ei adresse i minne
 - Les/skriv data frå/til adressa

Liten avsporing: Program 4

- MEN kva med minne som er byte adresserbart?
- Ikkje plass til (f.eks) 16 bit som må til for "FOXX"
- Program:
 - Hent X
 - Hent Y
 - Summer X og Y



Korleis vert eit program utført 2

- Nokre viktige utførande einingar
 - ALU (Arithmetic Logical Unit)
 - ALU Register
 - Register for eksternt minne
 - Program Counter (PC) register
 - Instruksjons Register (IR)
 - General Purpose Registers

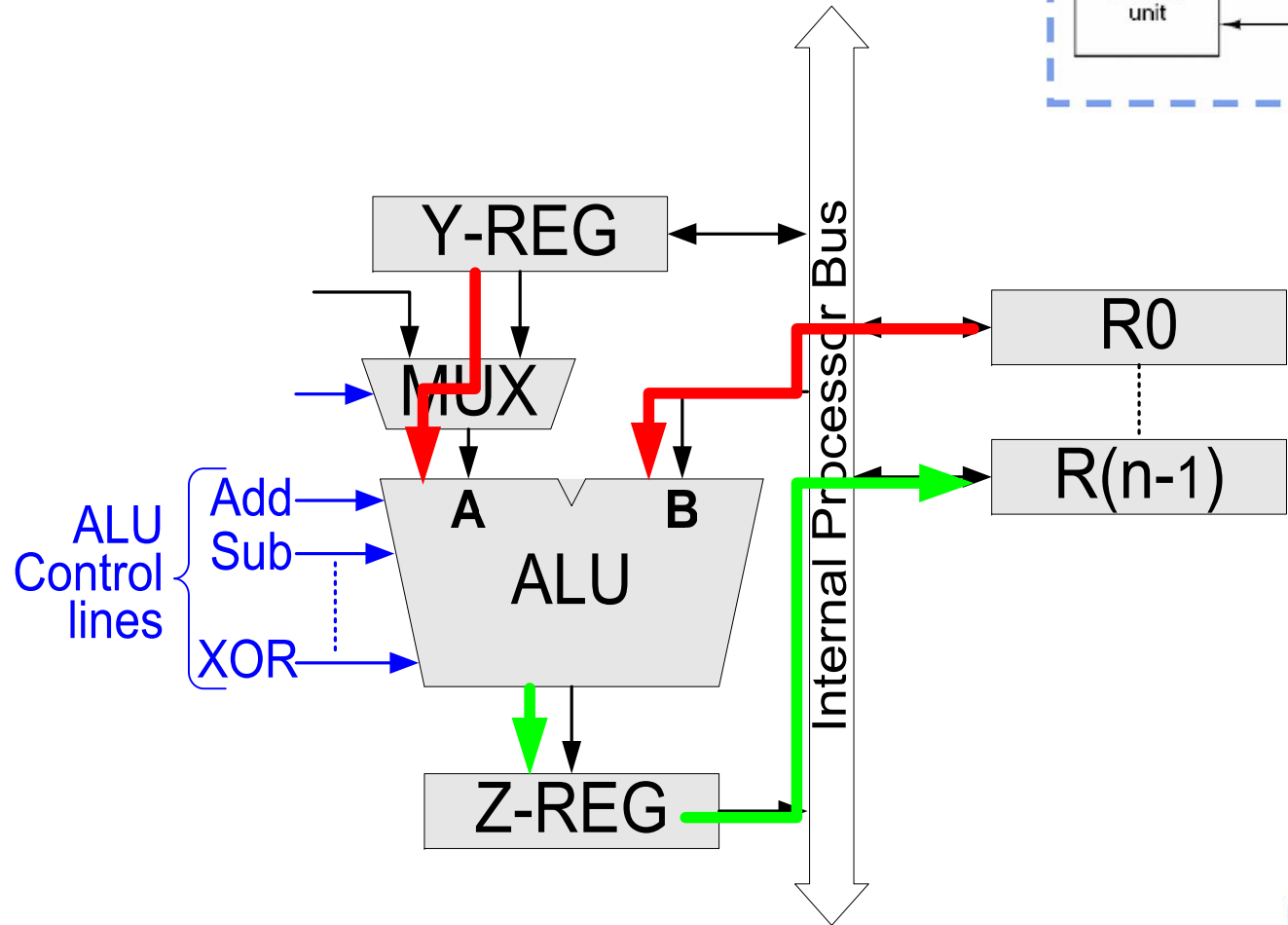
Hukselapp:

Hent instruksjon (Fetch)

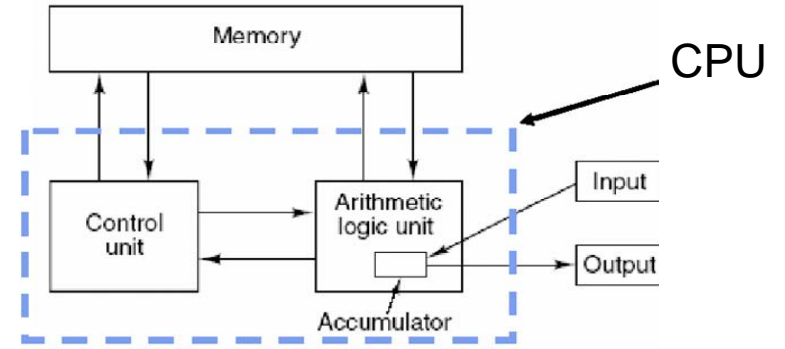
Dekodar instruksjonen (Decode)

Utfør instruksjonen (Execute)

Dathpath



Von Neumann-arkitektur



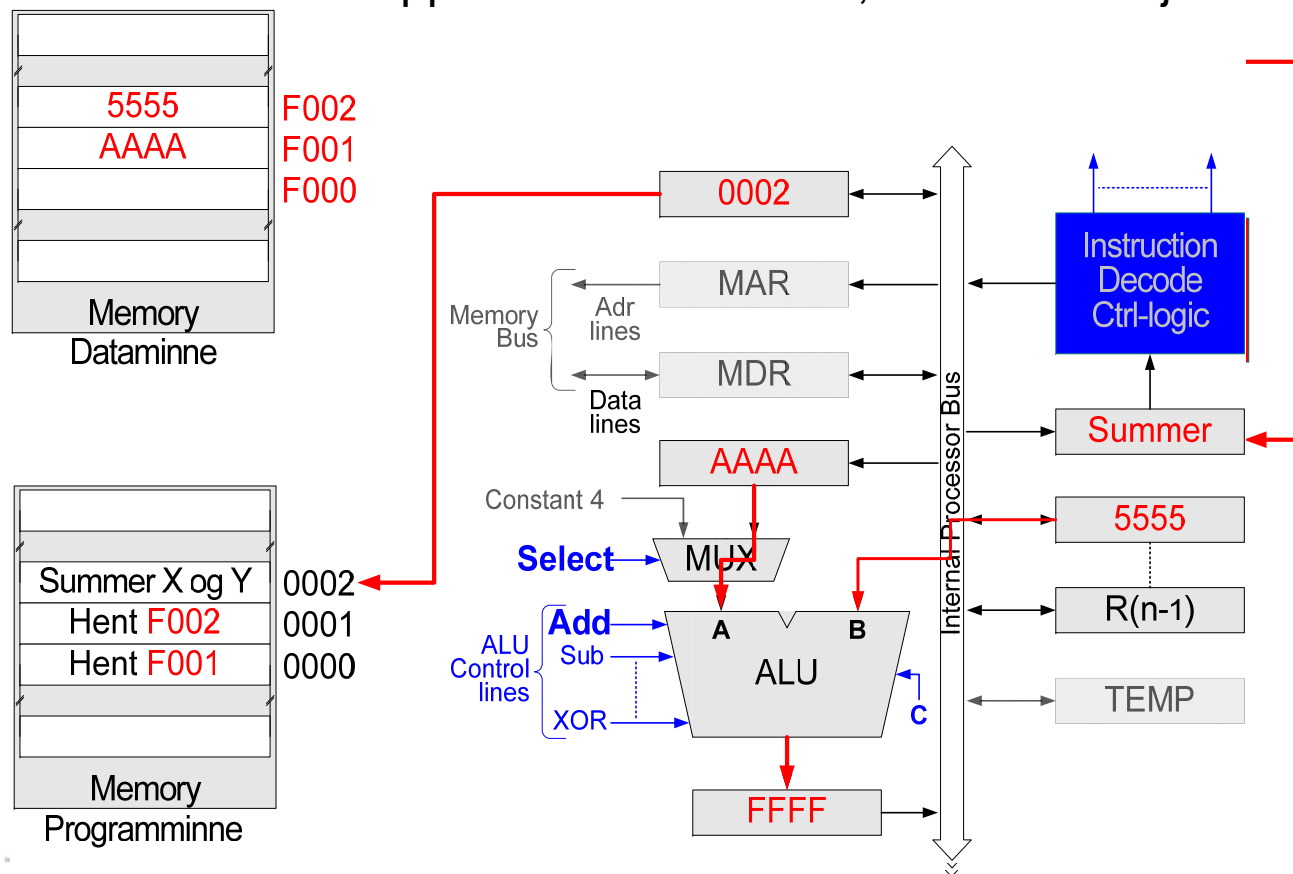
Instruksjons utførelse

Har henta data og lagt dei i Register 0 og Register Y

Fetch: hent neste instruksjon til Instruksjons registeret

Decode: Finn ut kva instruksjon det er

Execute: Sett opp utførandeeinheiter, utfør instruksjon



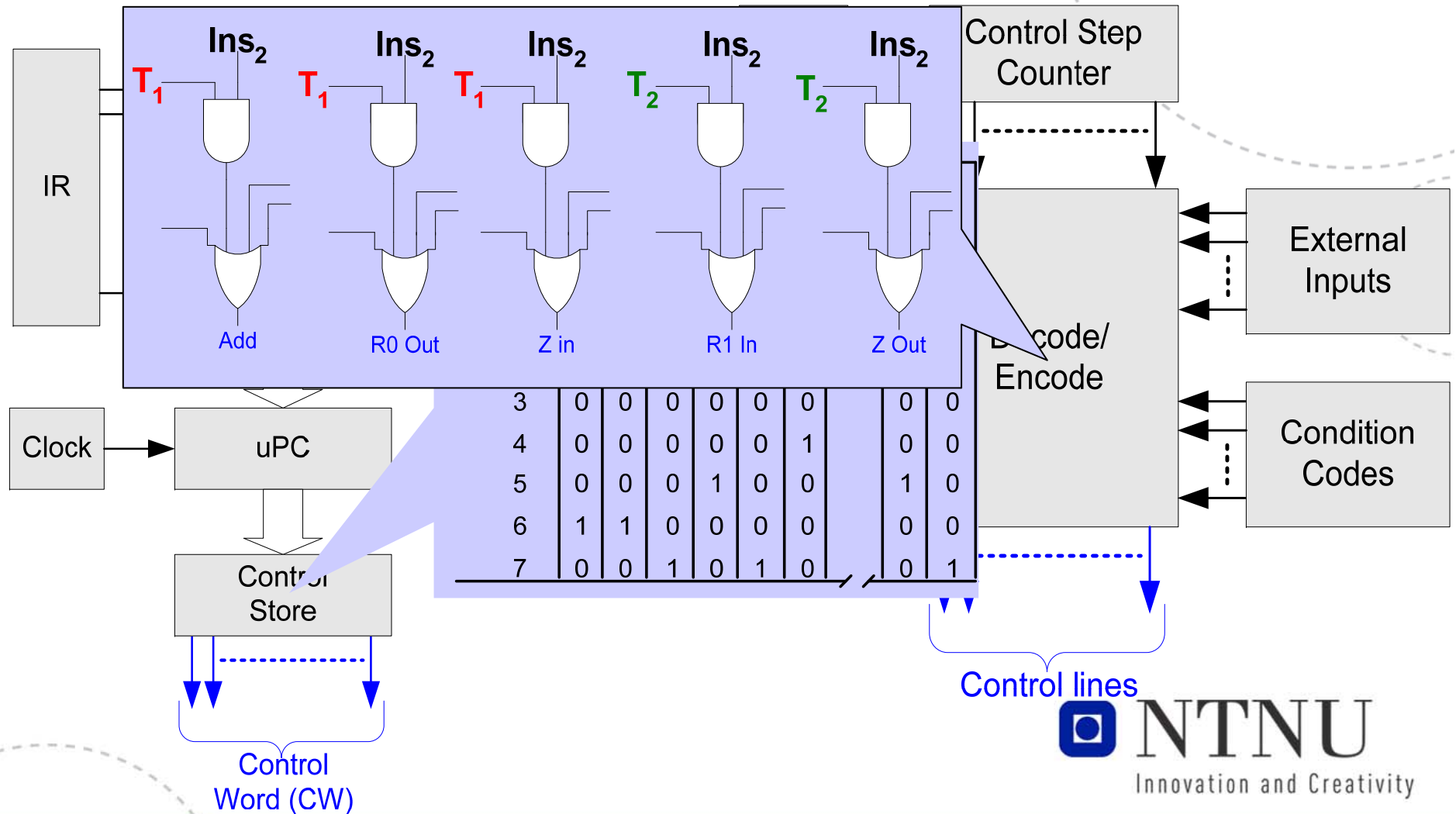
Instruksjonset og styreeinheit 1

- To hovud typar styreeinheit
- Microprogram
 - Instruksjonar er bygd opp av mikroinstruksjonar
 - Maskin i styreeinheit som utfører instruksjonane stegvis
 - Fleksibel (kan endre på instruksjonssete)
 - Mange typar instruksjonar og adresseringsmåtar
- "Hardwired"
 - Kunn maskinvare
 - Ei maskinvare statemaskin som styreeinheit (eller helst berre logic)
 - Kan ikkje endrast (oppdaterast)
 - Rask

Instruksjonset og styreeinheit 2

Microprogrammed

Hardwired

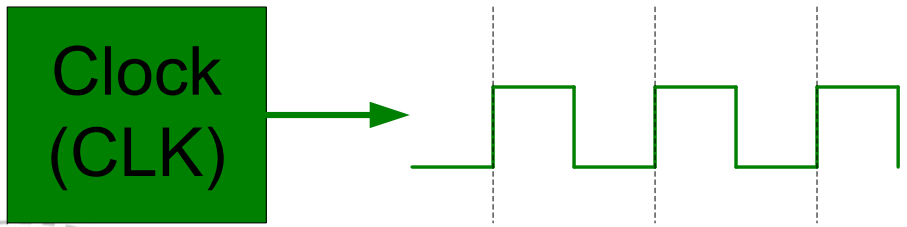
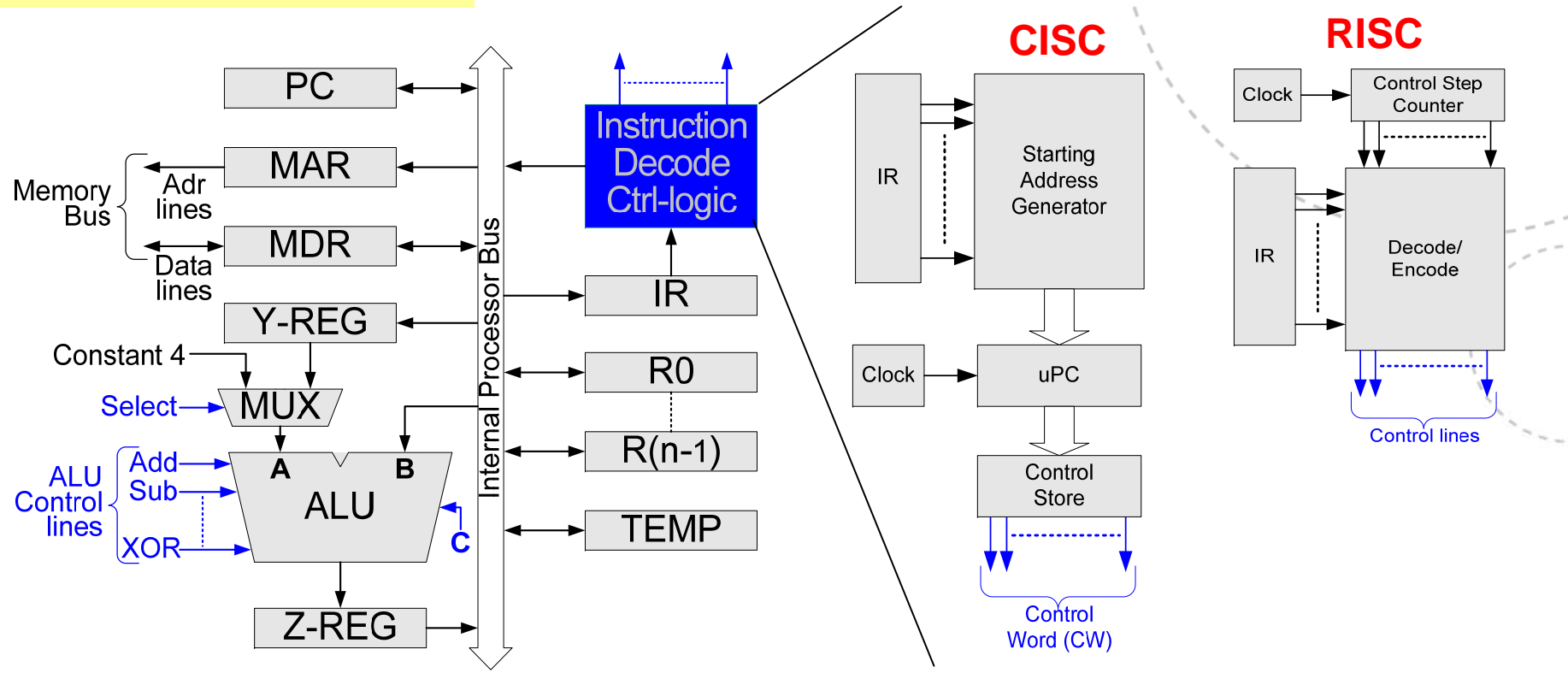


Instruksjonsett type CISC/RISC

- CISC (Complex Instruction Set Computer)
 - Avanserte instruksjonar (kan utføre mykje)
 - Bruke fleire einingar **ADD MinneAdrX, MinneAdrY, MinneAdrSvar**
 - Nærare høgnivå språk
 - Fleksibelt instruksjonsett (Viss mikroprogram kan oppdaterast/endrast)
 - Variabel lengde på instruksjonar
 - Variabel tid på å utføre instruksjonar
 - Mikroprogram (ikkje alltid)
 - Mange typar instruksjonar og adresseringsmåtar
- RISC (Reduced Instruction Set Computer)
 - Enkle instruksjonar
 - Register til register instruksjonar
 - Bruke mange enkle instruksjonar på å utføre oppgåver **ADD R0, R1**
 - Fast instruksjonsett (Hardwired)
 - Fast lengde på instruksjonar
 - Fast tid på å utføre instruksjonar
 - Ei maskinvare statemaskin som styreeinheit (eller helst berre logic)
 - Kunn ein type adresering LOAD/STORE
 - Rask

CISC: ADD MinneAdrX, MinneAdrY, MinneAdrSvar

RISC: ADD R0, R1



CISC/RISC Prosessorar

| | CISC | | | RISC | |
|----------------------|----------------|---------------|----------------|--------|---------------|
| Prosessor | IBM 370/168 | VAX 11/780 | Intel 80486 | SPARC | MIPS R4000 |
| Årstall | 1973 | 1978 | 1989 | 1987 | 1991 |
| # instruksjoner | 208 | 303 | 235 | 69 | 94 |
| Instr. størrelse (B) | 2-6 | 2-57 | 1-11 | 4 | 4 |
| # adr.modi | 4 | 22 | 11 | 1 | 1 |
| # gen. registre | 16 | 16 | 8 | 40-520 | 32 |
| Mikroprogram (KB) | 420 | 480 | 246 | - | - |