IN1020 uke 10

GRUPPE 10

Plan for dagen

- Abstraksjonsnivå
- Mer ALU
- Mente
- Porter
- Ukesoppgaver og obligjobbing

Risikovurdering og trusselmodellering

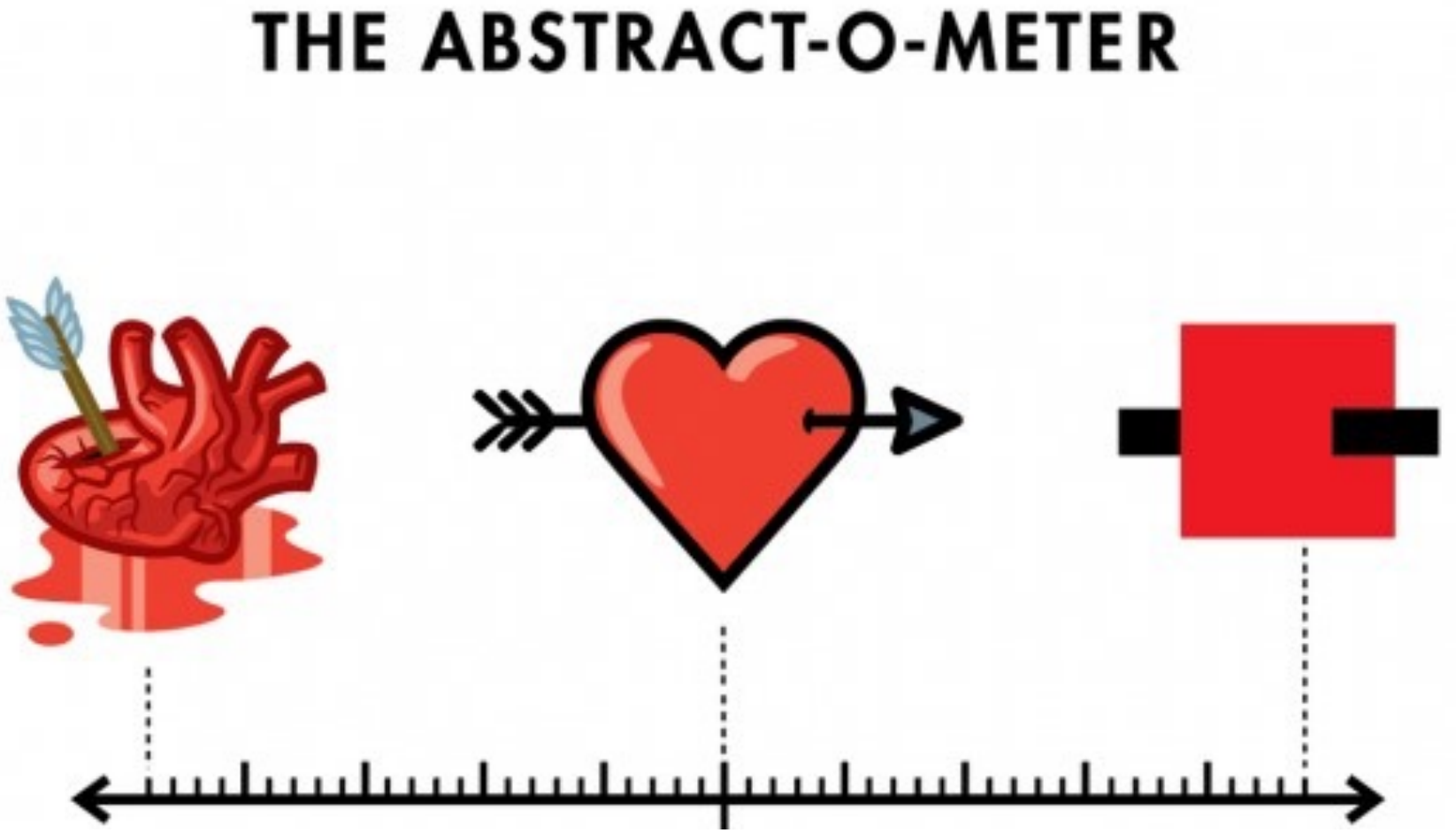
- En risikovurdering er et verktøy for å identifisere uønskede hendelser og risikoen for at disse skal inntreffe – datatilsynet
- Trusselmodellering består av å identifisere, analysere og beskrive relevante angreps-scenarier:
  - Hva kan skje?
  - Hvordan kan våre verdier skades?
  - Hvem kunne være interessert i å skade oss?
- Nødvendig for å se svakheter og hvordan man forebygger angrep
- *Nulldagssårbarhet*
  - En sårbarhet som er ukjent for eller er uadressert av leverandør

Trusselmodellering

Ulike innfallsvinkler:

- Fokus på **trusselaktør**
  - Identifisere trusselaktørens mål, hvordan går de fram for å oppnå dem, hva er angrepsvektor og angrepsflate.
- Fokus på **system**
  - «Plukke fra hverandre» systemet, identifisere de ulike delene og avdekke hvor og hvordan de kan angripes.
- Fokus på **verdi**
  - Identifisere verdiene i systemet, avdekke hvordan brudd på informasjonssikkerhet kan inntreffe.

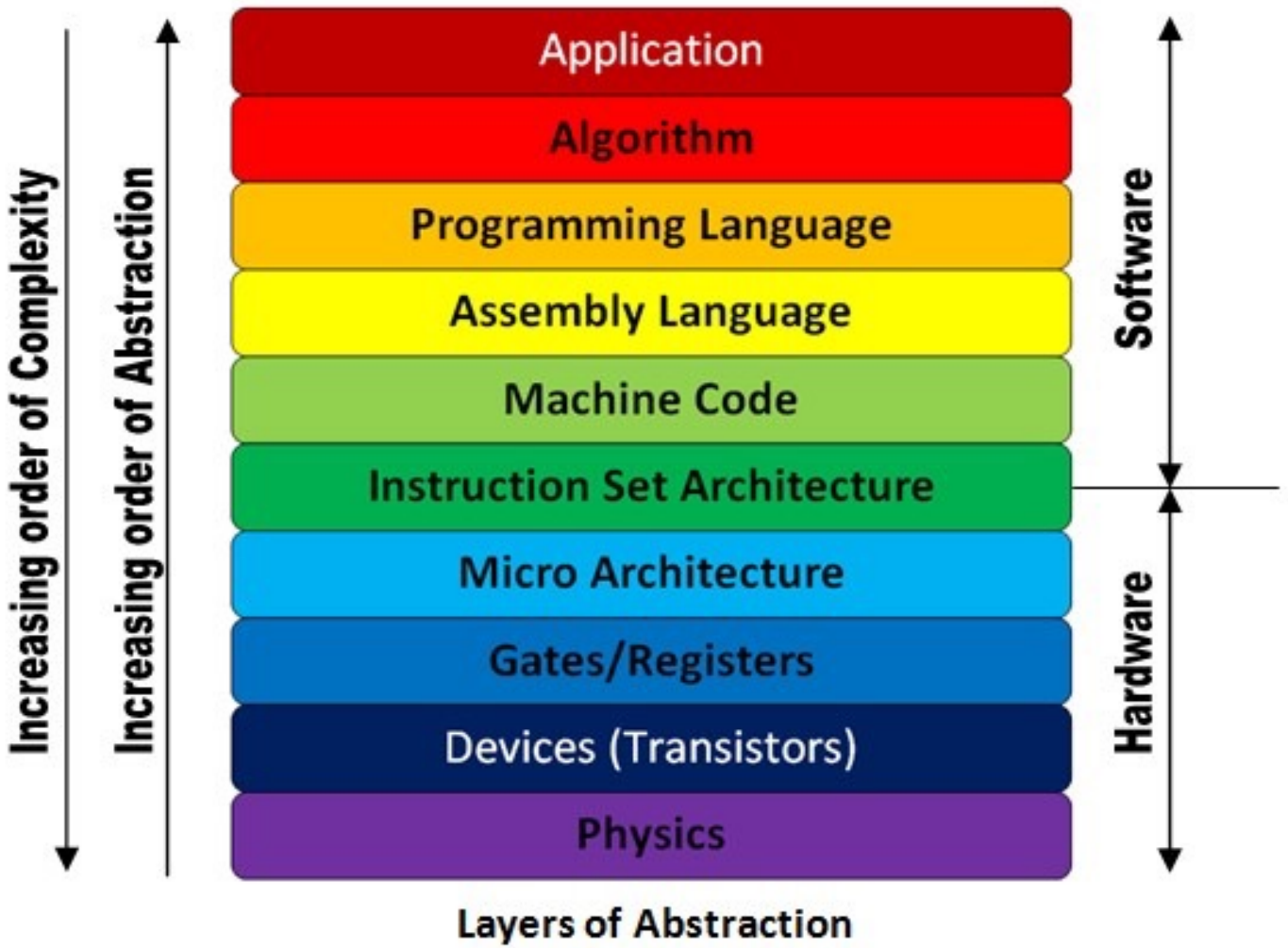
➢ Lær av feil: Hva har gått galt tidligere?



When you start to understand the concept of abstraction levels and you just can't stop getting more into it

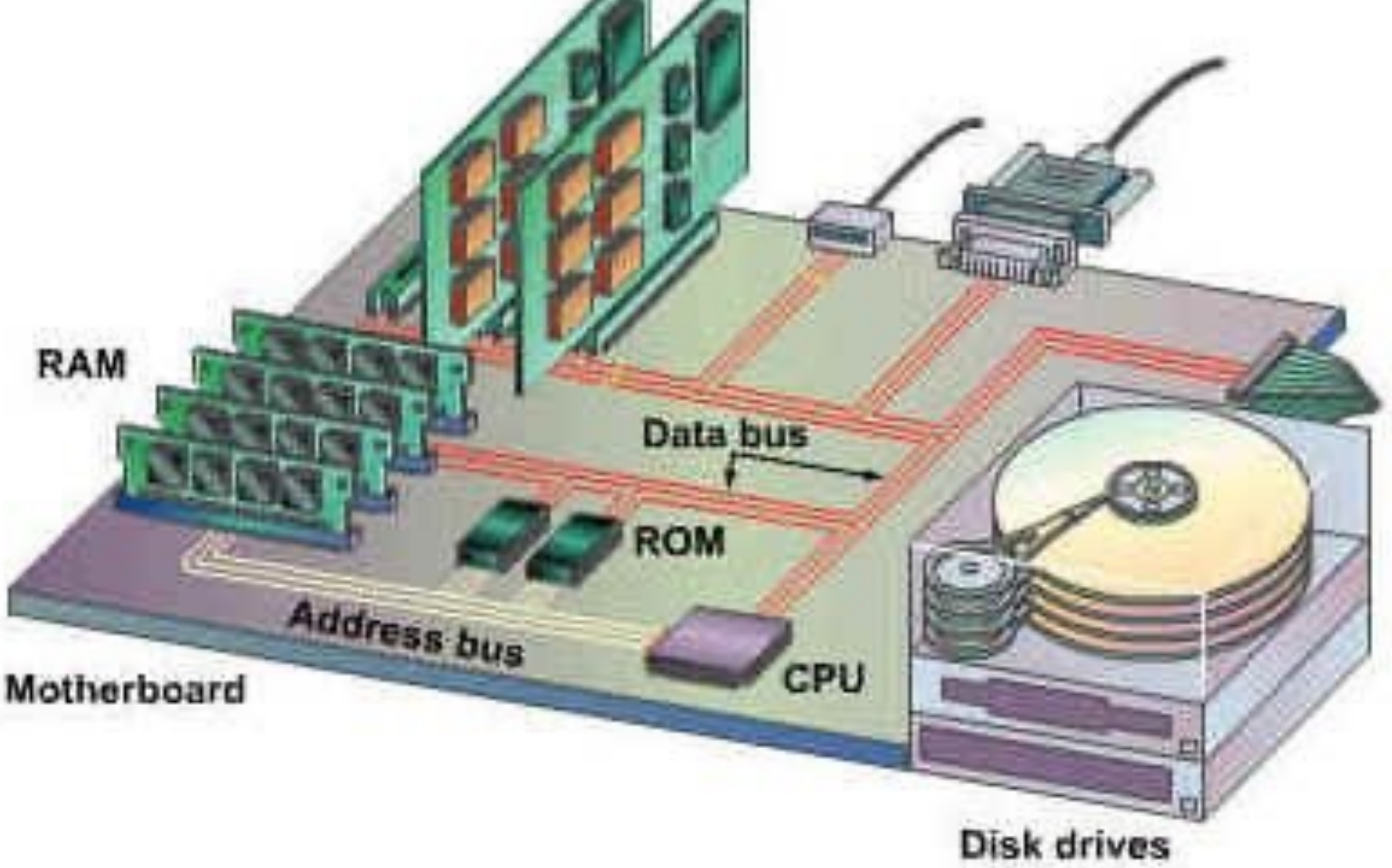


Abstraksjon


Increasing order of Complexity
Increasing order of Abstraction
Application
Algorithm
Programming Language
Assembly Language
Machine Code
Instruction Set Architecture
Micro Architecture
Gates/Registers
Devices (Transistors)
Physics
Software
Hardware
Layers of Abstraction

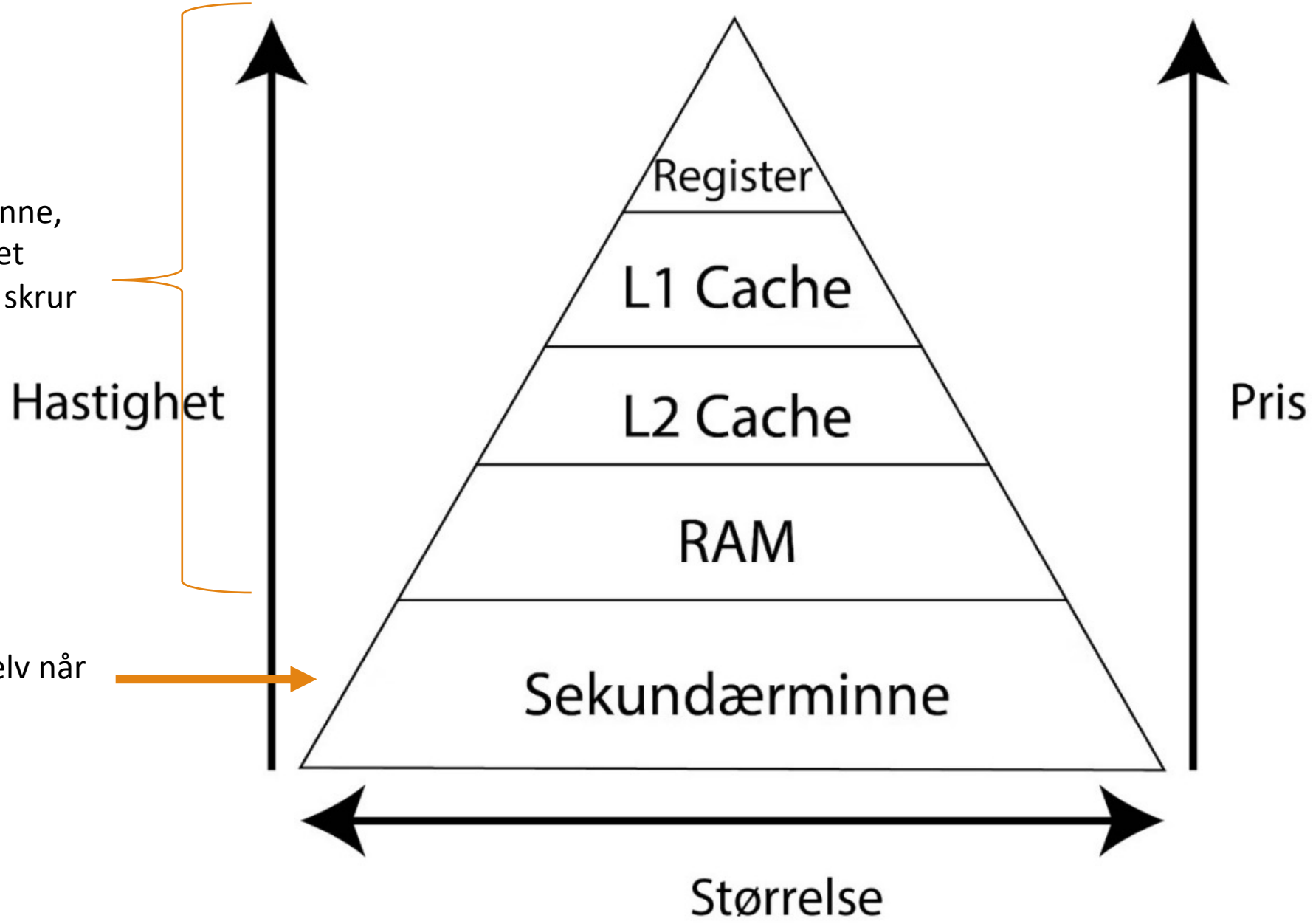
BUS

- Kommunikasjonskanal mellom registre, ALU, minne og I/O enheter
- Frakter data mellom komponenter eller i selve komponenten
  - Internt i en CPU er det en eller flere busser som overfører data mellom interne registre
- Bus kan deles mellom flere enheter, men kun en enhet kan sende om gangen
- Tenk på en vanlig buss (BUS), som frakter mennesker (data) til en holdeplass (komponent/register)


RAM
Data bus
ROM
Address bus
CPU
Motherboard
Disk drives

Mentimeter

Primærminne,
Blir resettet
når PC-en skrur
seg av



Beholder dataen selv når
PC-en skrur seg av

Mentimeter

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

Cache hit og Cache miss

- Skjer ved minneaksessering
- Cache-hit = finner dataen i cachen
- Cache-miss = finner ikke dataen i cachen
  - Må hente fra annet sted. F.eks. I Rammen, og dette er tregere
- Finnes også RAM-hit og RAM-miss

Oppgave 3 fra ukesoppgavene

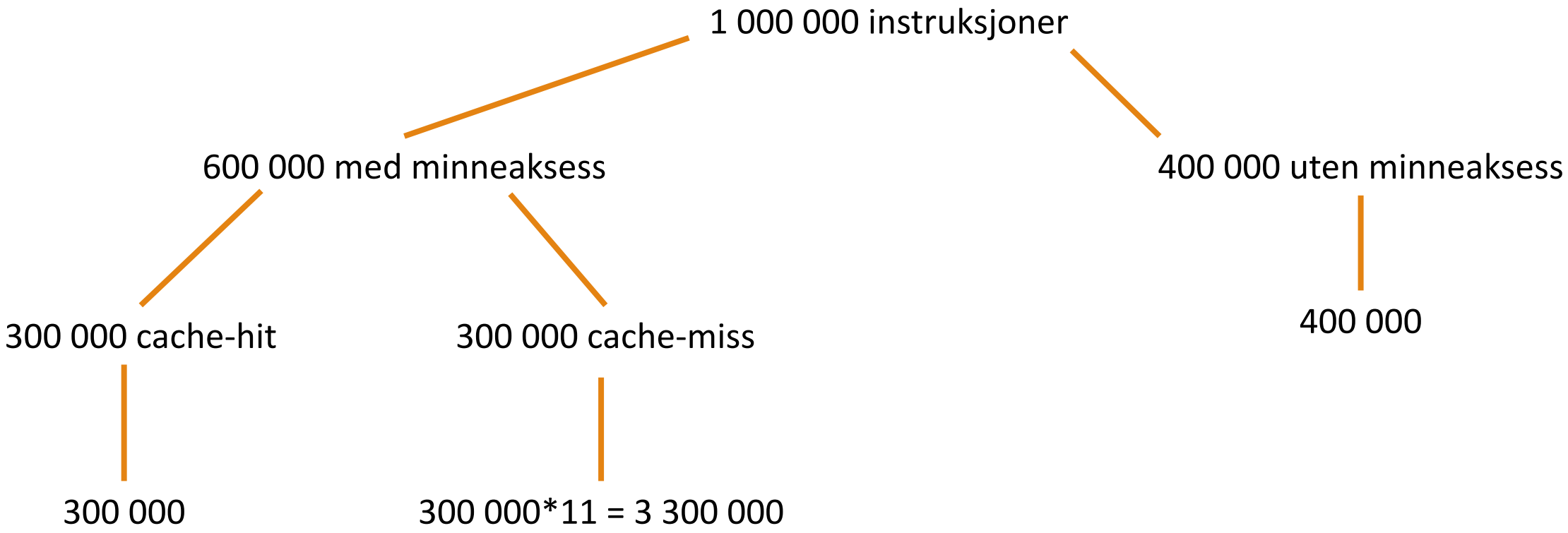
Anta at en prosessor vil prosessere en instruksjon per klokkesykel og skal totalt prosessere 1.000.000 instruksjoner som typisk vil ha behov for å hente data fra minne for 60% av instruksjonene, hvor det vil uheldigvis være 50% cache-miss. Anta videre at det typisk vil være en penalty (økt forsinkelse) for cache-miss på 10 klokkesykler. Hva mange klokkesykler vil det hele ta?

Økt forsinkelse = 11 klokkesykler med cache miss

Slide fra gruppe 2

1 instruksjon = 1 klokkesykel

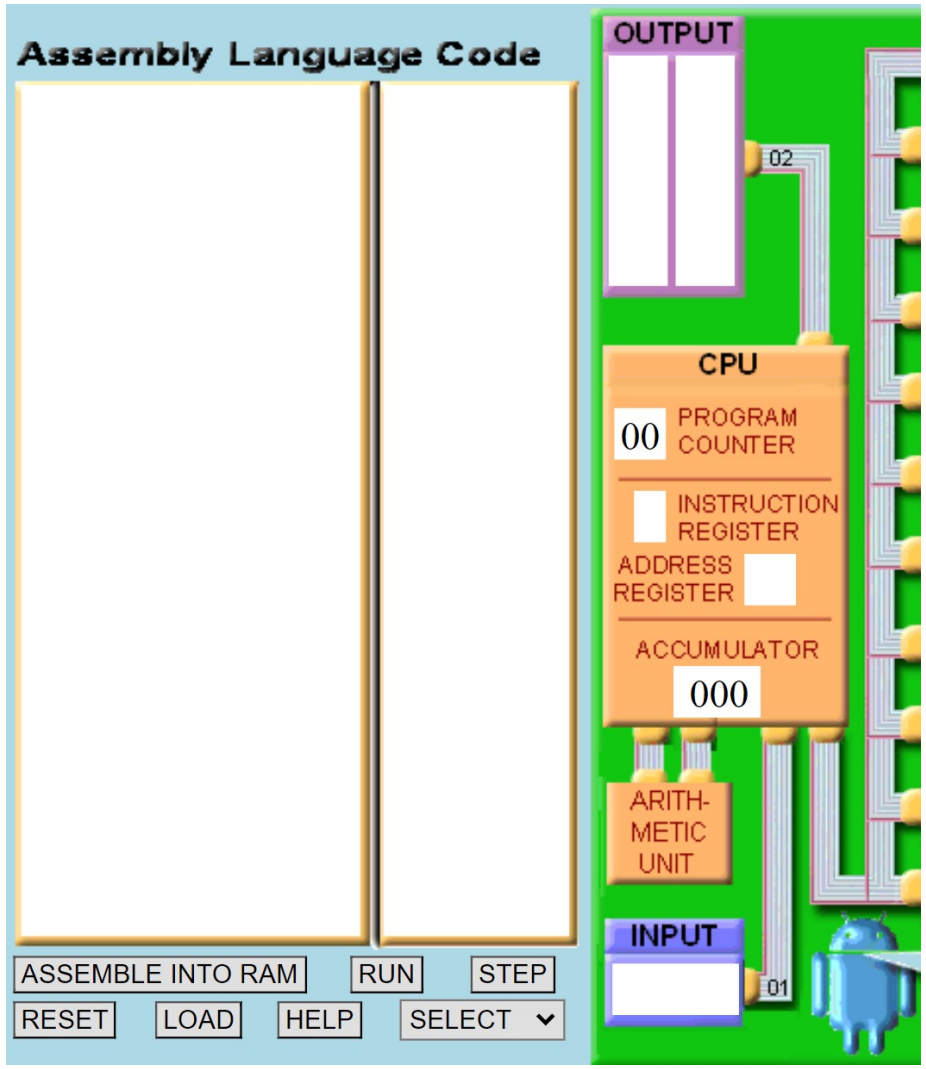
Løsningsforslag

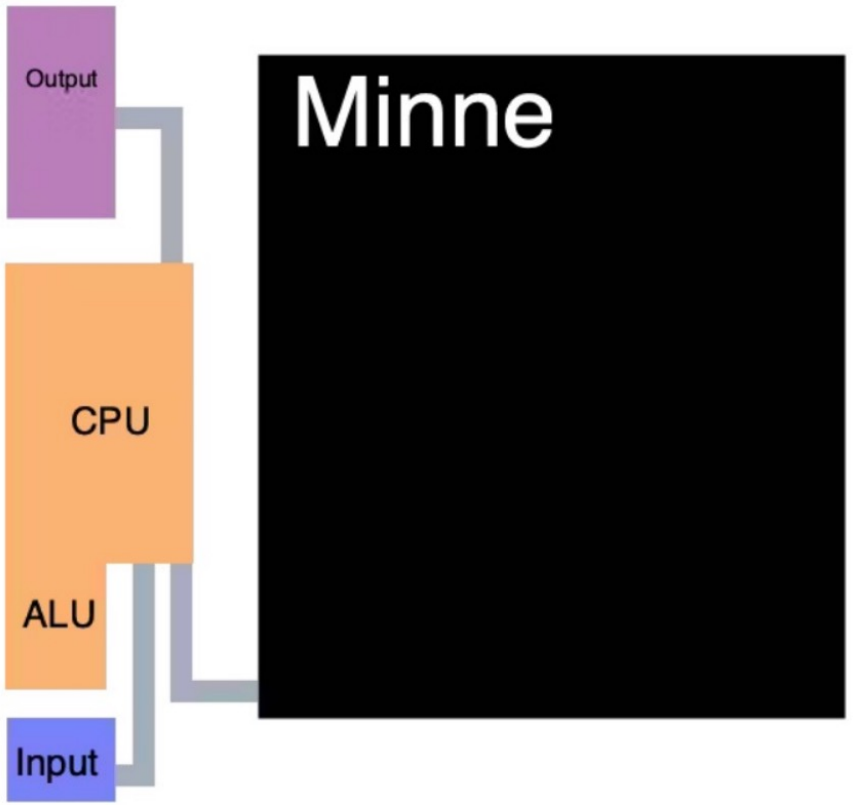


Totalt antall klokkesykler: 300 000 + 3 300 000 + 400 000 = 4 000 000

Datamaskinarkitekter

- CPU
- ALU
- ADDER
- Logiske Porter
- Transistor

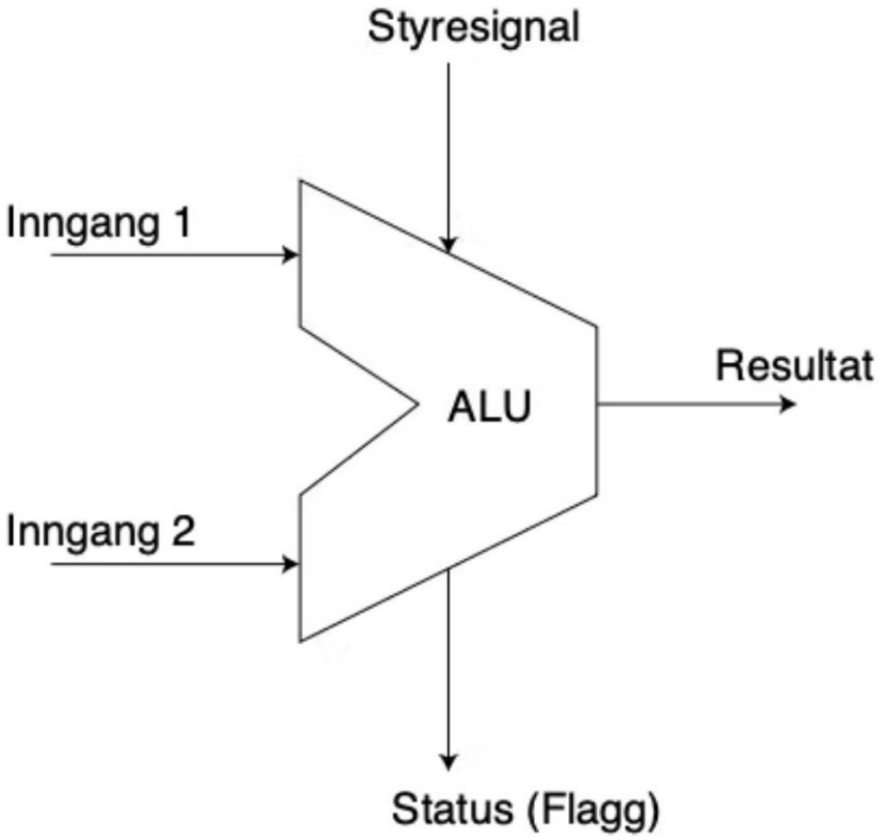




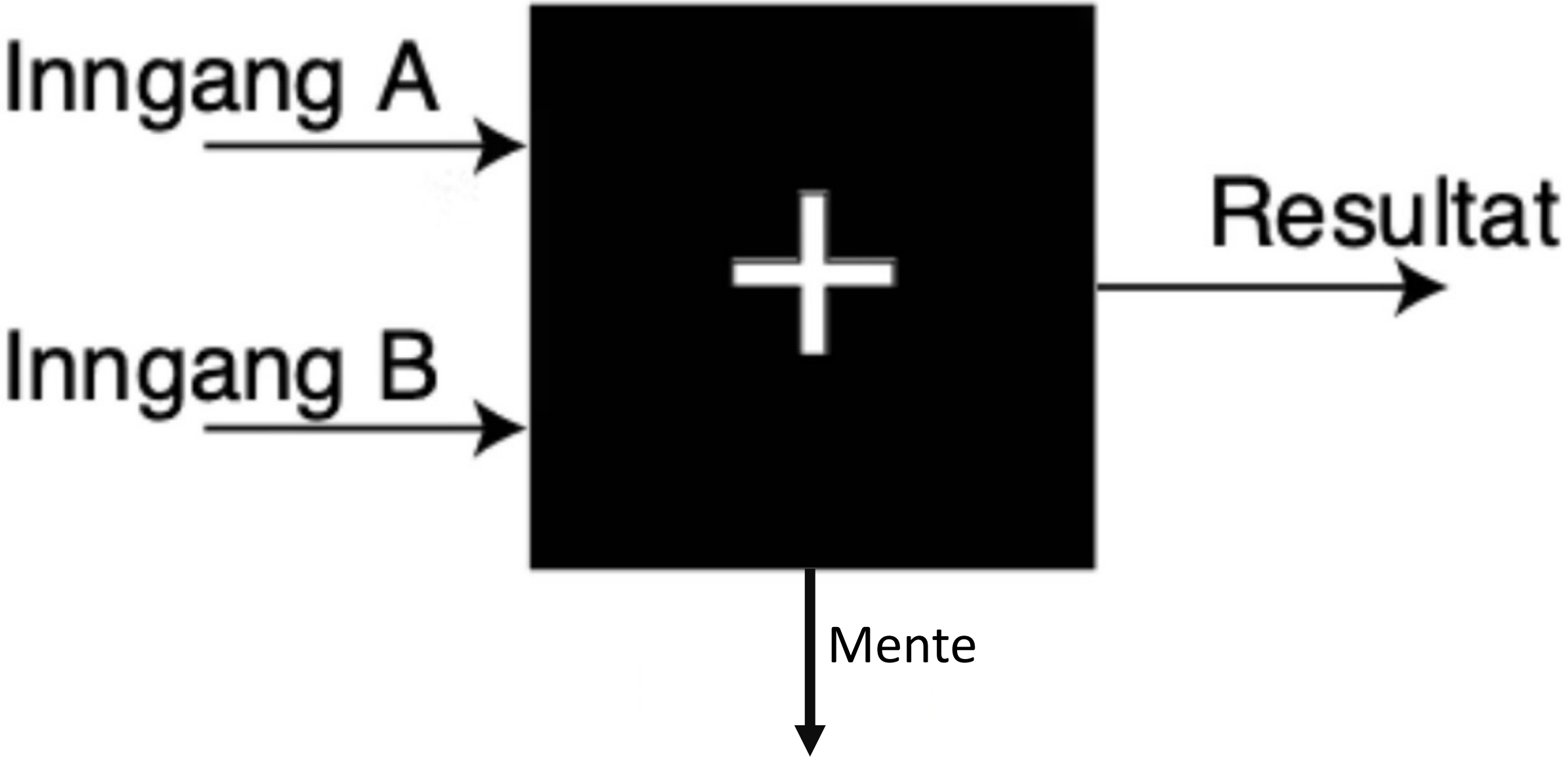
LMC - forenklet

ALU – Aritmetisk logisk enhet

- Tre innganger:
  - Styresignal: bestemmer operasjonen som skal gjøre i ALU-en.
    - Addisjon
    - Subtraksjon
  - Inngang 1: 1 bit
  - Inngang 2: 1 bit
- To utganger:
  - Status (flagg): gjør at vi kan ha mente
  - Resultat: resultat av operasjonen



Mentimeter



Mente

VANLIG ADDISJON

```
  26      26      26
+ 98    + 98    + 98
=       = 14    =  4
```

```
    1       1 1     1 1
  26      26      26
+ 98    + 98    + 98
= 124   =  24   = 124
```

BINÆR ADDISJON

```
    1       1       1
  1       1       1
+ 1     + 1     + 1
=       = 0     = 1 0
```

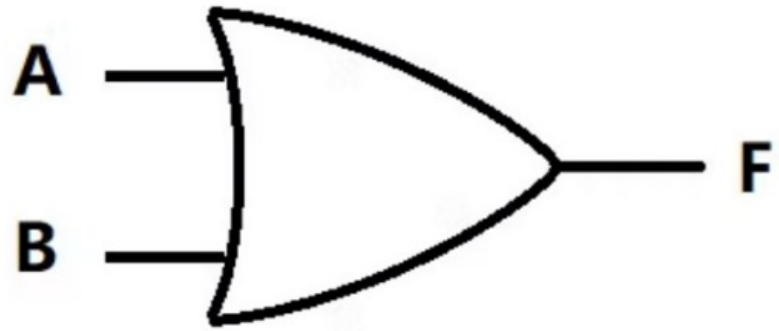
```
      1       1 1     1 1
  1 1     1 1     1 1
+   1   +   1   +   1   +   1
=       =   0   = 0 0   = 1 0 0
```

Funksjonsuttrykk

OR - port

$F = A+B$

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1



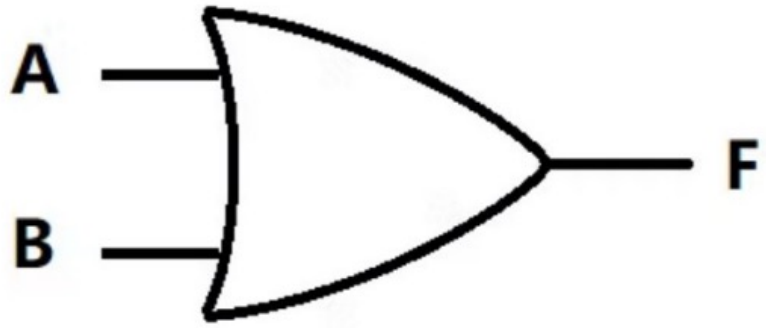
Port representert med ulike former/figurer. F.eks er spiss pil = OR

Sannhetstabell = viser alle mulige kombinasjoner

OR - port

$$F = A+B$$

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

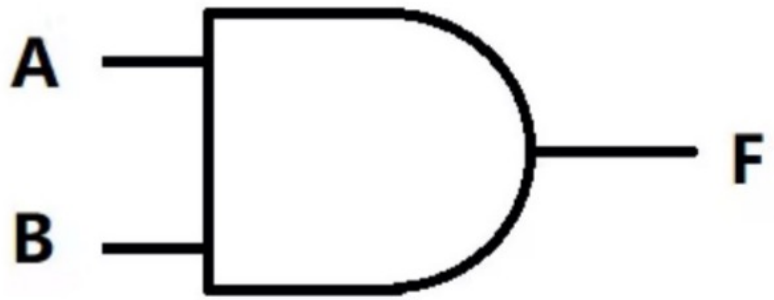


Gir ut 1 når en av signalene er 1

AND - port

$F = AB$

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

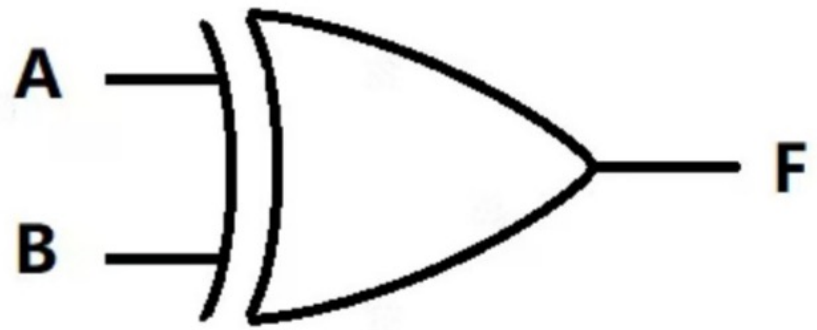


Gir ut 1 når begge signalene er 1

XOR - port

$$F = A \oplus B$$

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

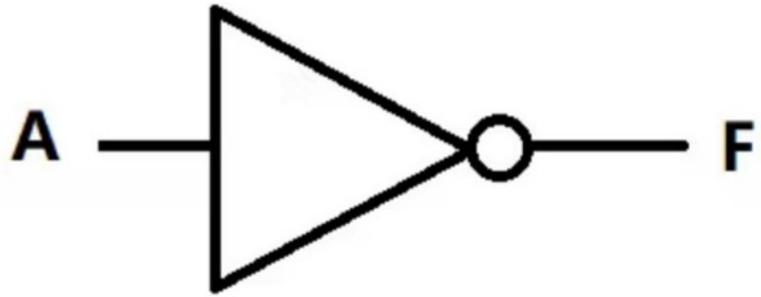


Gir ut 1 BARE hvis en av signalene er 1

INV - port

$F = A`$

A	F
0	1
1	0



Inverterer signalen, signalet blir endret

NOT porter inverterer resultatet i sannhetstabellen

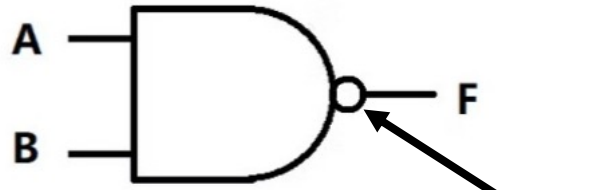
AND

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

NAND - port

$F = (AB)`$

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

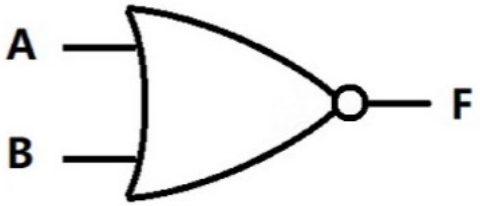


NOT porter illustrert med en ball

NOR - port

$F = (A+B)`$

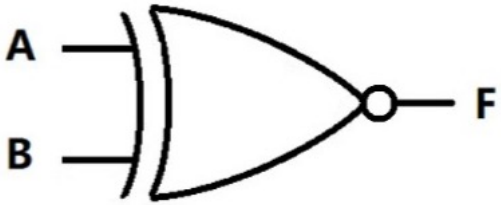
A	B	F
0	0	1
0	1	0
1	0	0
1	1	0



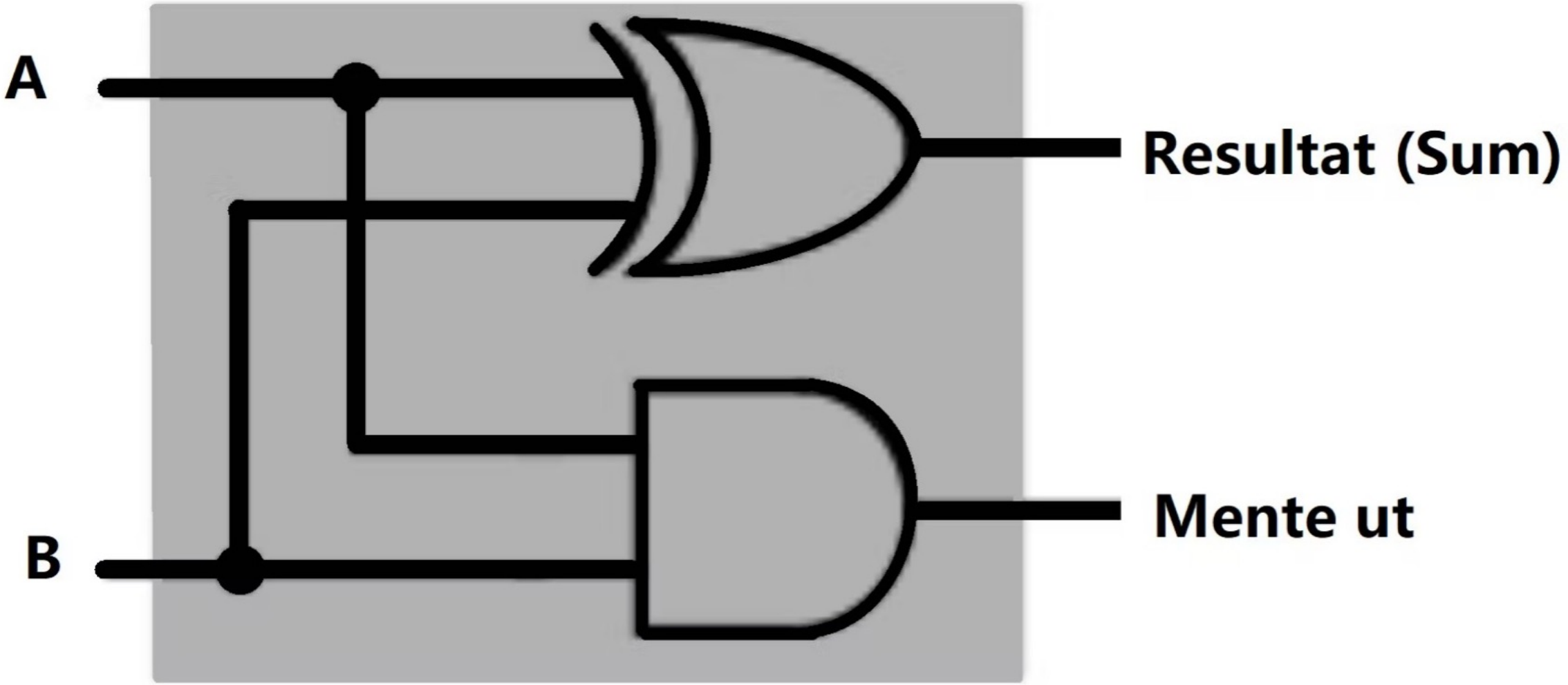
XNOR - port

$F = (A \oplus B)`$

A	B	F
0	0	1
0	1	0
1	0	0
1	1	1



Mentimeter



HALV-ADDER (HA)

Hvordan vite hva som blir input?

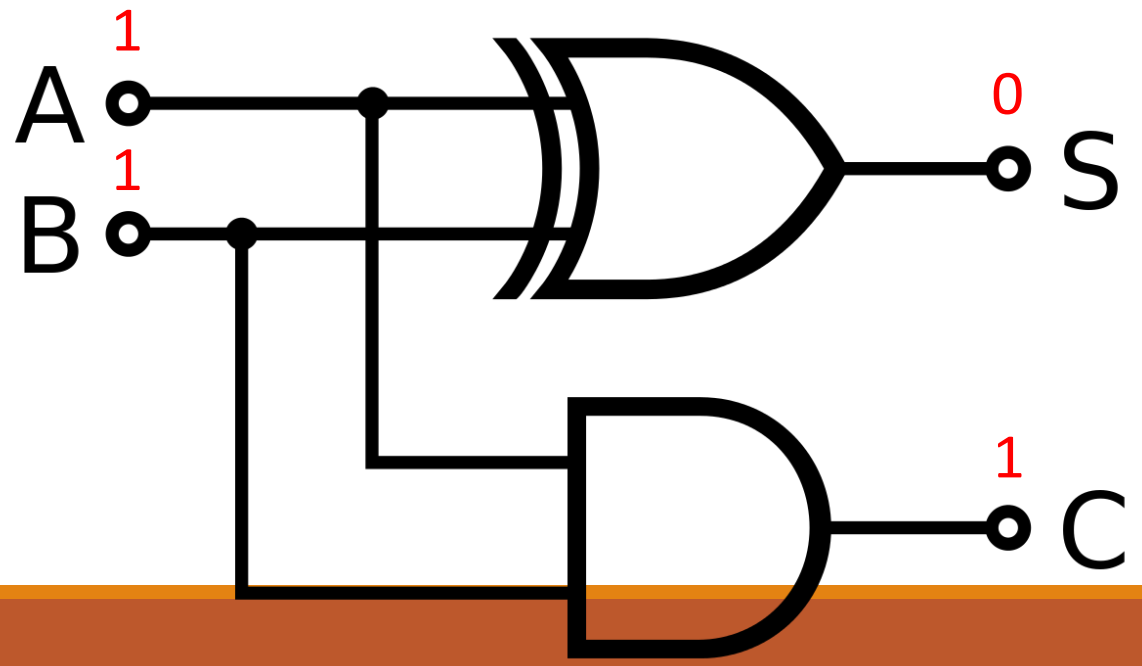
1. Leser hva signal A er og signal B er (kan enten være 1 eller 0)
2. Se på hva hvilke signal som går inn i kretsen (2 signaler som enten er 0 eller 1)
3. Identifisere hvilken port signalet går gjennom (ved å gjenkjenne formen)
4. Bruke sannhetstabell til å finne output til porten

XOR

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

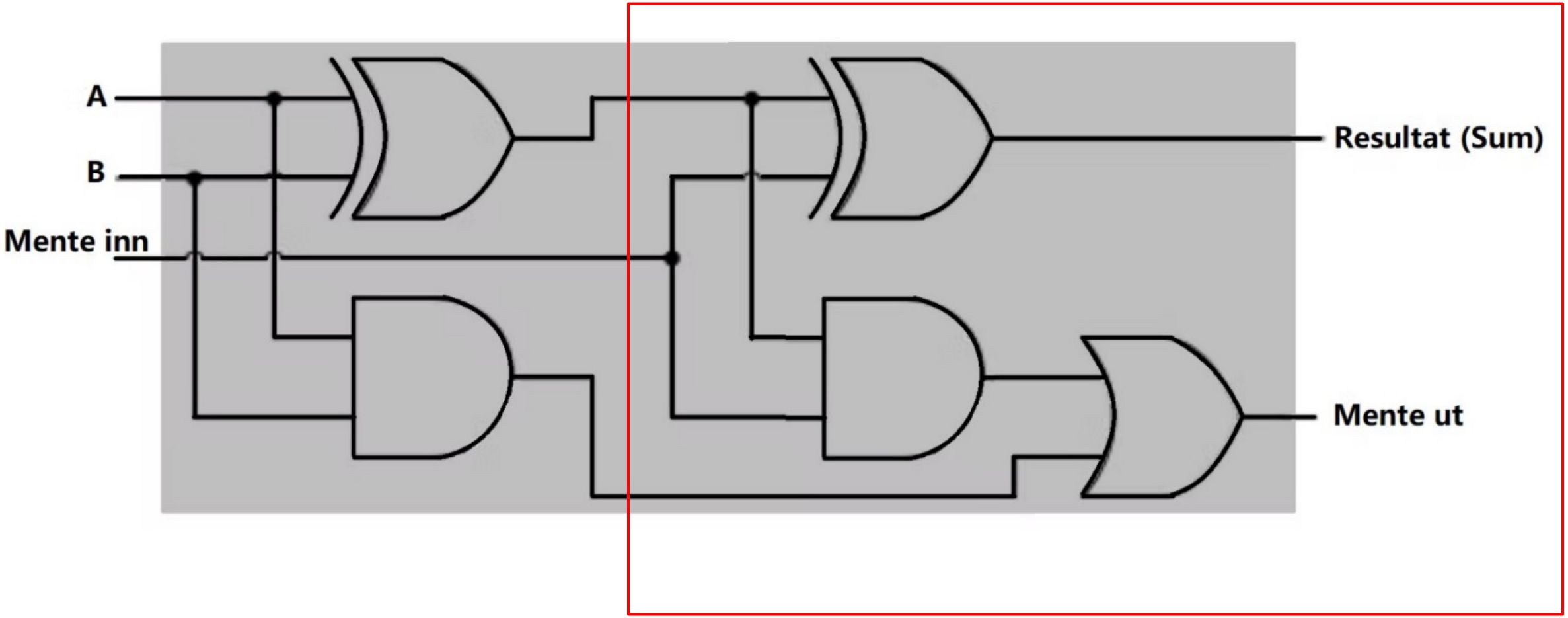
AND

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1



Mentimeter

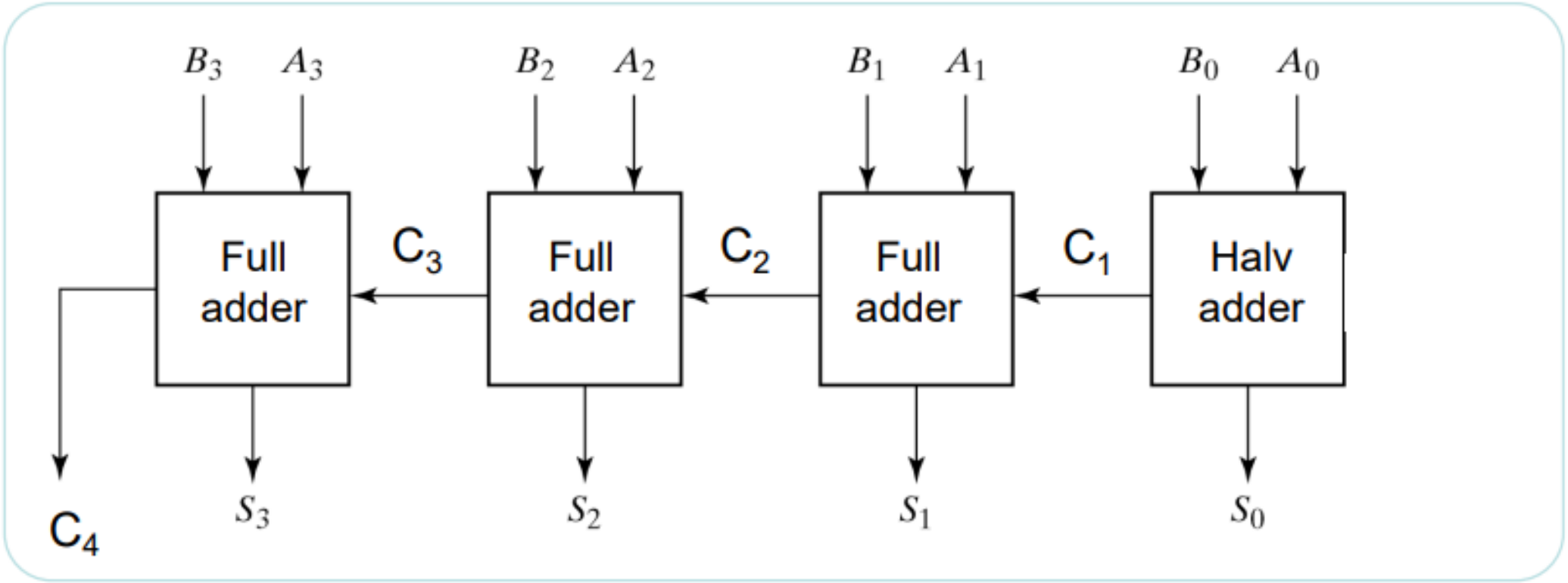
Dette er lagt til + mente inn



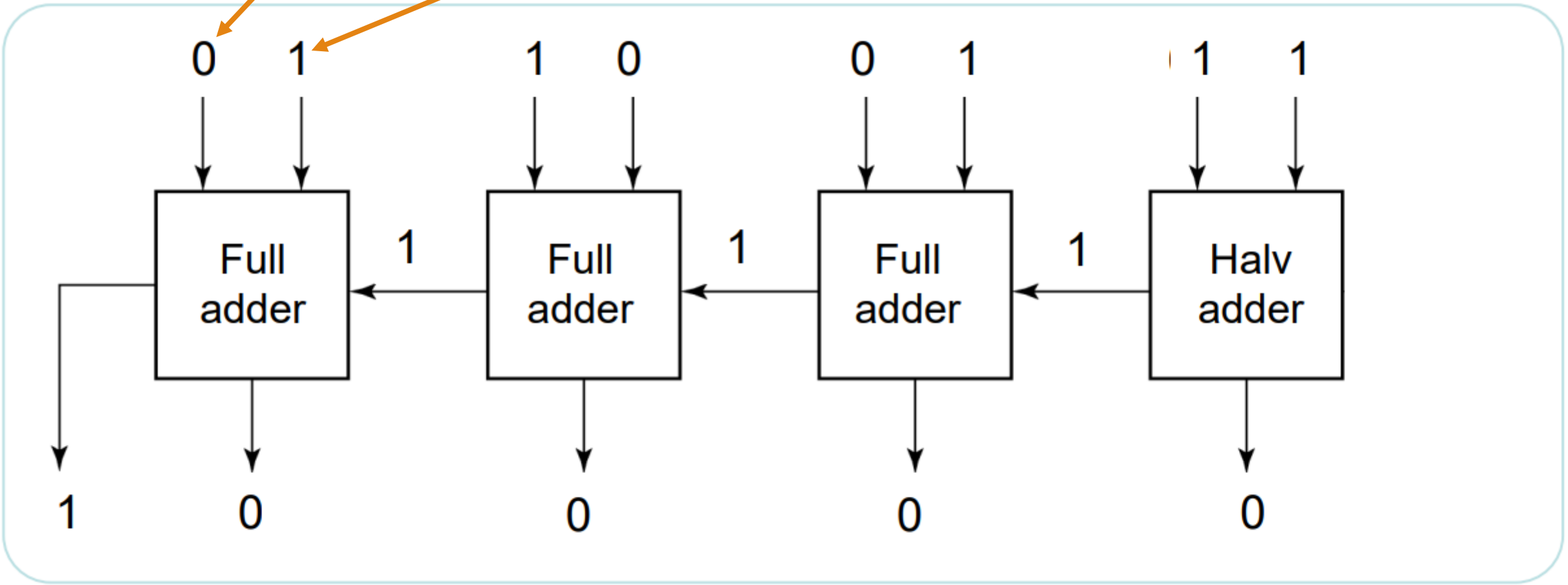
FULL-ADDER (FA)

Seriell adder

4-bits binær adder



Hva blir 0101 + 1011? (Svar=10000)



Lab

- Jobbe videre med oblig eller ukesoppgaver
- Send meg mail eller teamsmelding hvis dere lurer på noe eller trenger hjelp til oblig