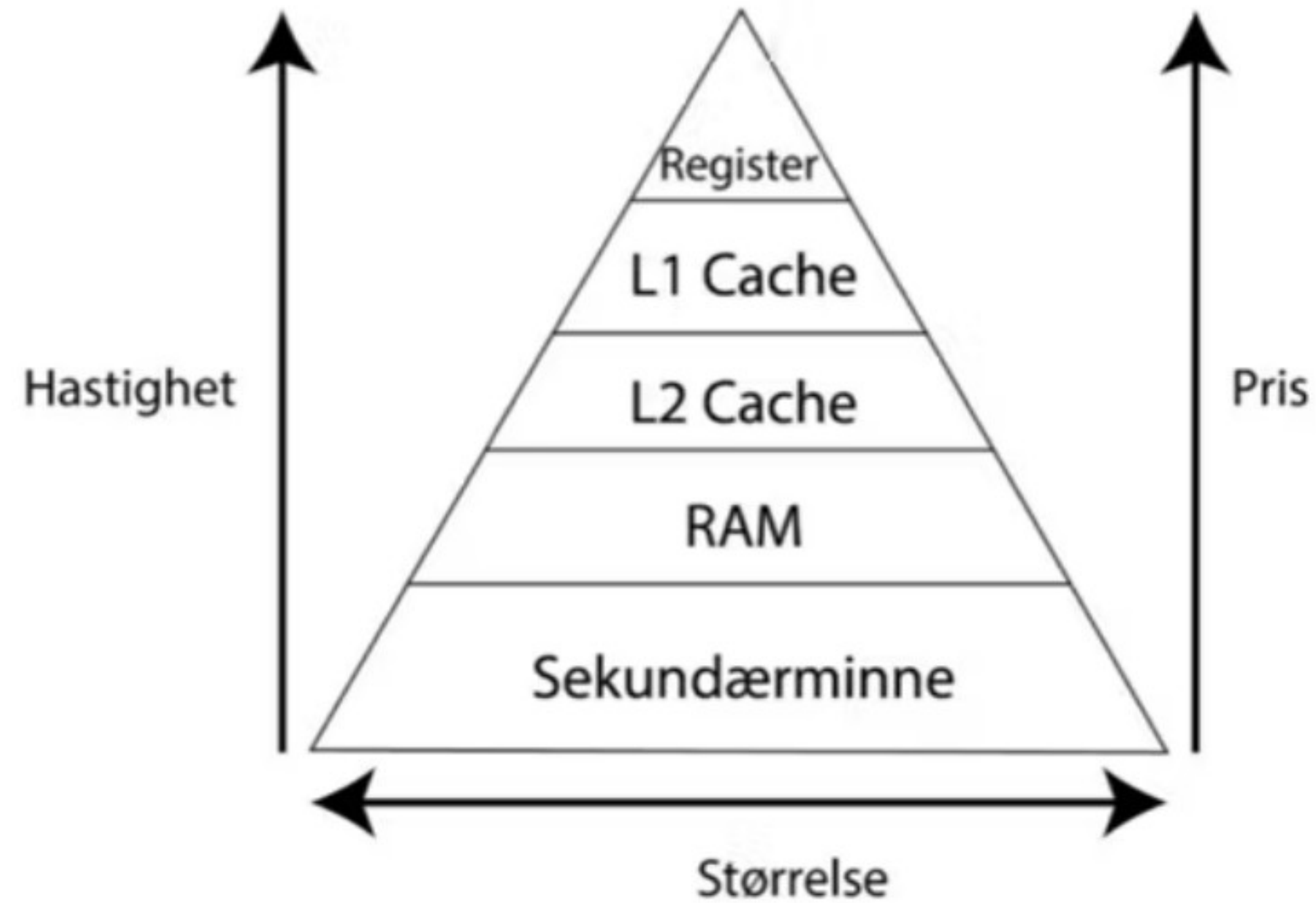


Idag:

- Minnehierarki
- cache hit / cache miss
- BUS
- Obligjobbing eller ukesoppgaver





Minnehierarki

- All data er lagret i en form for minne
- De har ulik pris, akseshastighet og kapasitet for lagring
- Øverst er de som ligger nærmest CPUen og som er raske å aksessere
- Øverst er også de som koster mest å lage
- Trekanten representerer hvor stor lagringsplass de har.

akseshastighet

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

- Mellomlager mellom CPUen og RAM
- Vi kan hente data og instruksjoner fra RAM og legge de midlertidig i Cache.
- Da kan vi raskere aksessere det vi trenger fra cache



Cache hit og cache miss

- Når en prosessor skal hente en instruksjon eller lese/skrive data vet den ikke om den skal hente fra RAM eller Cache.
- Hvis det prosessoren ber om ligger i cache får vi cache hit.
- Hvis dataen ikke ligger i cache får vi cache miss.
- Oppgave om cache hit og miss:



Eksamensoppgave 2019 (modifiserte tall)

Anta at prosessoren har 360 instruksjoner igjen å utføre. Det tar én klokkesykel pr instruksjon, og man regner med å ha en minneaksessering på 70% og med cache-miss på 50%. En cache-miss fører til en total tidsbruk på 20 klokkesykler. Cache-hit bruker totalt én klokkesykel. Hva er totalt antall gjenstående klokkesykler?



Løsning

- Instuksjoner som ikke krever minneaksessering: 1 klokkesykel
- Instuksjoner som krever minneaksessering:
 - Cache hit: 1 klokkesykel
 - Cache miss: 20 klokkesykler
- 30 % av instuksjonene (180) krever ikke minneaksessering: 180 klokkesykler
- 70 % krever minneaksessering (252), 50 % cache-miss:
 - 126 instuksjoner får cache hit: 126 klokkesykler
 - 126 instuksjoner får cache miss: $126 * 20 = 2\,520$ klokkesykler
- $108 + 126 + 2\,520 = 2754$ klokkesykler totalt



Ukesoppgave 3

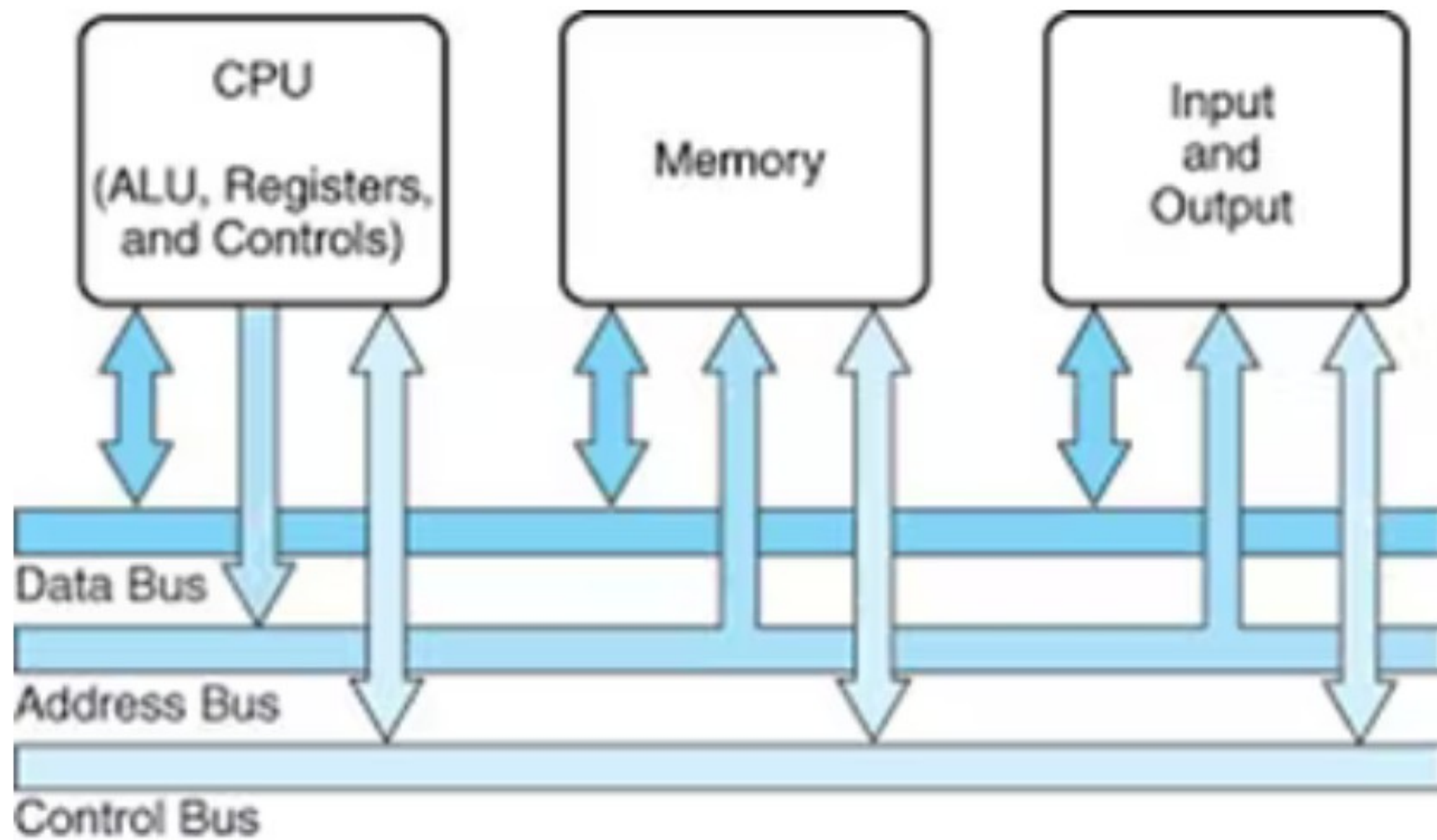
Anta at en prosessor vil prosessere en instruksjon per klokkesykel og skal totalt prosessere 1.000.000 instruksjoner som typisk vil ha behov for å hente data fra minne for 60% av instruksjonene, hvor det vil uheldigvis være 50% cache-miss. Anta videre at det typisk vil være en penalty (økt forsinkelse)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



Løsning

- Instuksjoner som ikke krever minneaksessering: 1 klokkesykel
- Instuksjoner som krever minneaksessering:
 - Cache hit: 1 klokkesykel
 - Cache miss: + 10 = 11 klokkesykler
- 40 % av instuksjonene (400 000) krever ikke minneaksessering: 400 000 klokkesykler
- 60 % krever minneaksessering (600 000), 50 % cache-miss:
 - 300 000 instuksjoner får cache hit: 300 000 klokkesykler
 - 300 000 instuksjoner får cache miss: $300\,000 * 11 = 3\,300\,000$ klokkesykler
- $400\,000 + 300\,000 + 3\,300\,000 = 4\,000\,000$ klokkesykler totalt





BUS

- BUS sørger for kommunikasjon mellom komponenter
- Feks: data som sendes mellom CPUen og RAM
- Finnes flere typer BUS for forskjellig type informasjon

