Kontrollenheten

* **CPU oppbygging**
  + Å få utført en instruksjon går ut på p bryte ned instruksjoner til en sekvens av mikro-operasjoner.
    - I har tre typer mikro-operasjoner
      * Flytte data mellom to registre
        + Ved hjelp av brytere som styres av kontrollenheten kontrollerer vi dataflyten mellom registre.
      * Utføre aritmetiske eller logiske operasjoner (ALU operasjoner)
        + En kombinasjon av binære kontrollsignal velger ALU-operasjon dette styres av kontrollenheten.
      * Gi beskjed at noe skal leses eller skrives/ IO via systembuss.
        + Når vi leverer en instruksjon til instruksjonsregisteret er det kontrollenheten som finner ut hvilken instruksjon som skal utføres
  + Kontrollenheten har to oppgaver:
    - Sekvensering
      * Kontrollenheten skal sørge for at CPU-en stepper gjennom en serie av mikrooperasjoner i rett rekkefølge, denne rekkefølgen bestemmes av programmet som er under utføring.
    - Eksekvering
      * Kontrollenheten bruker binære kontrollsignaler til å sørge for at mikrooperasjoner blir utført.
  + Ved hver ny klokkepuls setter kontrollenheten opp en ny kombinasjon av styresignaler slik at rett mikrooperasjon blir utført.
    - Derfor må kontrollenheten vite:
      * Hvilken instruksjon er under utføring
      * Hvor langt i sekvensen av mikrooperasjoner har den kommet
      * Hvorvidt det må tas hensyn til eksterne hendelser (f.eks. avbrudd)
  + Kontrollenheten er en elektronisk krets, så all informasjon som den skal ta hensyn til må finnes som binære inngangssignaler.
  + Kontrollenheten har mange binære innganger slik at den kan gi utsignaler til CPU-en om hva den skal gjøre.
  + Det finnes to måter å bygge opp en kontrollenhet på:
    - 1. **Hardlogikk:** kontrollenheten sees på som en kombinatorisk krets som bygges opp av mange logiske porter.
      * Ved design av hardlogikk tar man for seg kontrollsignalene ett for ett.
      * For hvert av kontrollsignalene finner man et boolsk uttrykk for den kombinasjon av alle innganger som skal sette akkurat det kontrollsignaler.
      * Deretter designes kontrollenheten med hjelp av kombinatoriske grunnkretser som oppfyller disse boolske utrykkene.
      * **Fordel:** Svært hurtig
      * **Ulempe:** vanskelig å implementere avanserte instruksjoner
      * **Historie:** De første prosessorene (på 50- og 60-tallet) hadde hardlogikk kontrollenheter.
        + Fra midten av 80-tallet har de fått sin renessanse, og alle nyere design baserer seg på hardlogikk.
    - **2. Mikroprogrammert kontrollenhet**: kontrollenheten bygges opp som en programmerbar enhet som kjører et lavnivå program.
      * Svært annerledes fra hardlogikk.
      * Vi lager et bitmønster for hver mikrooperasjon og samler en sekvens av slike bitmønstre for hver instruksjon.
      * Mikroinstruksjon er et bitmønster (jukselapp) som forteller hvilke kontrollsignal som skal settes for å utføre en mikro-operasjonen. I det bitmønstre skriver vi 1 for kontrollsignalene som skal sette, og 0 for de som skal nullstilles.
      * Vi setter sammen sekvenser av mikroinstruksjoner som til sammen utgjør en hel instruksjon.
      * Ved hver ny klokkepuls finner kontrollenheten neste mikroinstruksjon og setter styresignaler i henhold til mikroinstruksjonen
        + Kontrollminnet (mikrominnet) inneholder følgende sekvenser av mikroinstruksjoner:

Hente syklus (lik for alle instruksjoner)

Utføringssyklus for hver instruksjon (ulik for alle instruksjoner)

Avbruddssyklus (lik for alle instruksjoner)

* + - * Vi må lage en elektronisk krets som:
        + Venter til det kommer en ny klokkesyklus
        + Finner neste mikroinstruksjon
        + Setter kontrollsignalene slik at de korresponderer med mikroinstruksjoner
      * **Fordel:** Svært fleksibel, man kan implementere svært avanserte og komplekse instruksjoner
      * **Ulempe:** lav hastighet. Aksesstiden til kontrollminnet begrenser ytelsen
      * **Historie:** Mikroprogrammerte kontrollenheter hadde sin storhetstid på 70-tallet og 80-tallet.
        + Etter hvert som man har greid å lage mer og mer avanserte hardlogikk-enheter har det blitt mindre bruk av mikroprogrammering.