

# Pipelining

Wednesday, 23 August 2023

11:42

- Far fare istruzioni contemporaneamente al processore MIPS multiciclo
- Diviso in 4 sezioni che vengono fatte contemporaneamente:
  - 1) Prendere istruzioni dalla memoria (Instruction Fetch, IF)
  - 2) Leggere i registri mentre si decifra l'istruzione, (Instruction decode, ID)
  - 3) Eseguire l'operazione o calcolare l'address (Execution, EX)
  - 4) Accesso ad un operazioni nella memoria (Data Memory Access, MEM)
  - 5) Scrivere risultato nel registro (Write Back, WB)
- Non sempre tutto si può fare contemporaneamente, a volte delle istruzioni non possono essere eseguite subito dopo, e questo viene chiamato hazards. Ci sono 3 tipologie di hazards:
  - 1) Structured hazards
    - L'hardware non può fare la combinazione che vogliamo
    - Es: abbiamo 1 memoria anzichè 2
  - 2) Data hazards
    - Uno step deve aspettare un altro per poter venire completato
    - Es: abbiamo 2 somme, ed il risultato della prima somma è necessaria per la seconda.
    - Può essere evitato il problema con:
      - ☐ Forwarding [Chiedere a camì se c'è]
      - ☐ Bypassing
    - Accesso a memoria che viene modificato
  - 3) Control hazards
    - Se abbiamo un if, come facciamo?
    - Risposta: Non facciamo, si fa in maniera sequenziale fino all'arrivo dell'if, non si fa nulla in concorrenza dopo l'if senza aver finito prima if
    - Oppure si può dare sempre vero che il branch sia vero, e se poi non è vero F