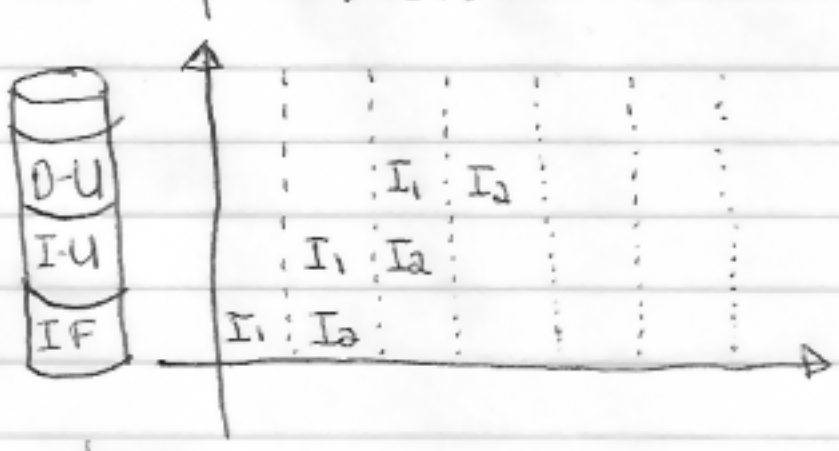
Comp 3725 Nov. 19/12

Studying ①Interstage Resource Dependency
②Instruction Prefetch (multiple instruction fetch) in a pipeline machine

We will consider:
- Only R-type Instructions
- In our machines we have 3 stages
  - IF stage (Instruction fetch + PC update)
  - IU stage (Decode + Operand Fetch)
  - D-U stage (Execution) "D-Unit"

Machine 1

Assume:
- machine can fetch only 1 instruction per IF time
- all the 3 stages take equal amount of time (= 1 unit)
- there are no data dependencies among instructions.
- there are no interstage dependencies (so all three stages can run in parallel)

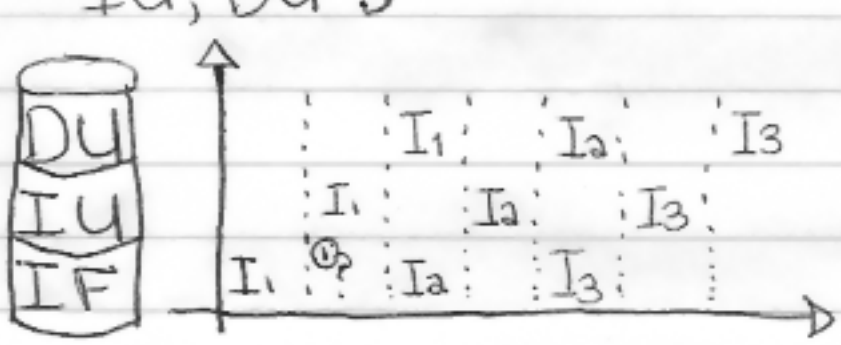


Assumptions

① I_2 can be fetch in IF while I_1 is in I-U because there is no resource dependency between IF and IU.

② Also, IU and DU has no resource dependency
③ Also, IF, IU, DU has no resource dependency.

Machine 2 (Implications of Interstage Dependency)
Similar to M1 except:
- IF and ID have resource dependency
- IF, DU } no dependency
  IU, DU }

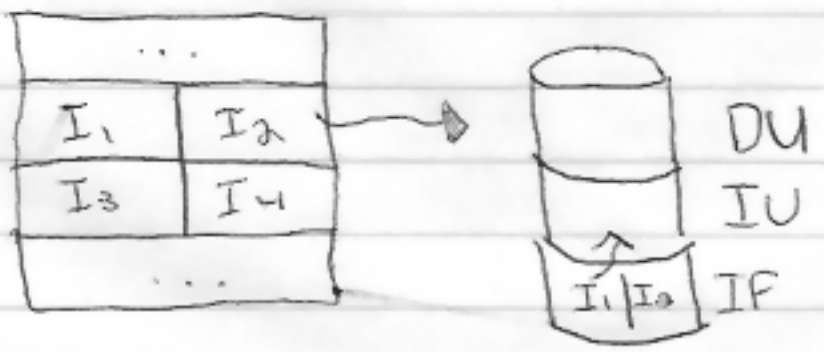


① I_2 cannot be fetched here because I_1 is in the IU stage and IF and IU has resource dependence

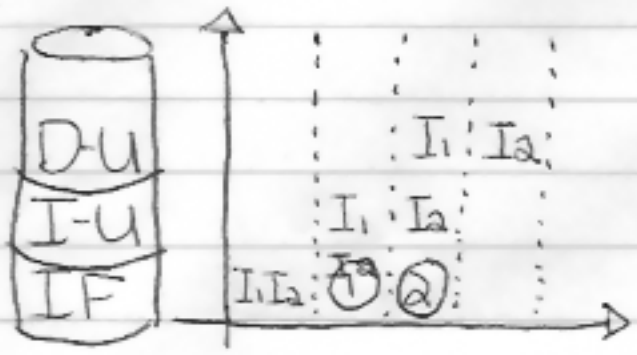
(Also the first bubble, we will have 1 input for every 2 cycles)

Machine 3 (Implications of Interstage Resource Dependency)
Similar to M2 except:
- M3 can fetch 2 instructions/cycle

Aside Implementing multiple instruction fetch



Ex:- IBM 360/155
3 instructions prefetch
- CDC 512 bit wordsize
8 instructions/word
- VLIW (Very Large Instruction Word)



① I_3, I_4 cannot be fetch in this circle because:
- IF-IU has resource dependence
- And I_2 is still in IF

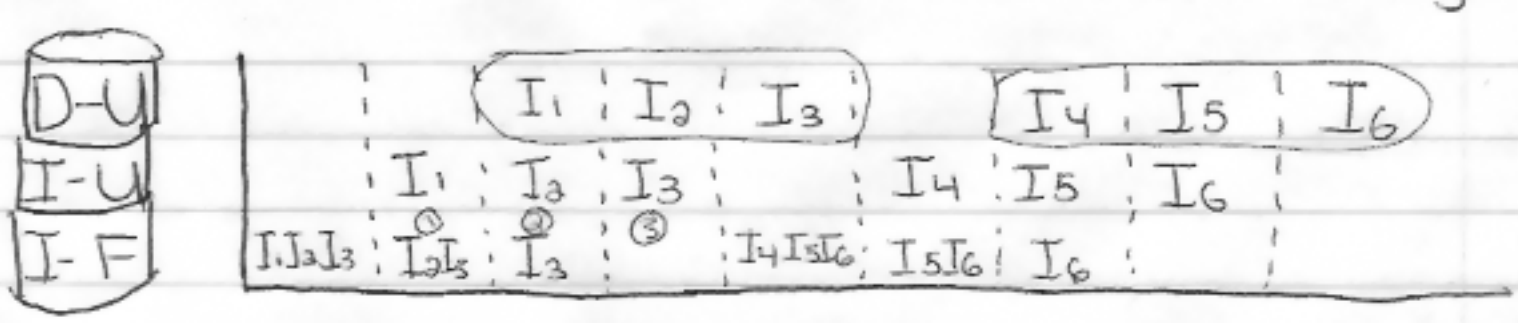
② I_3, I_4 still cannot be fetched because IF and IU are dependent

∴ T_P of $M_3 = 1 + \frac{3N}{2}$

After the first bubble we have 2 instructions per every 3 circles.

Machine 4

Same as M3 except:
- M_4 can fetch 3 instructions/cycle



∴ T_P of $M_4 = 1 + \frac{4}{3}N$

Machine 5

In General, if we prefetch R instructions/cycle

$\therefore T_p$ of M5 $= 1 + \left(\frac{R+1}{R}\right) N$

If $R \gg \infty$ Then $T_p \rightarrow 1+N \rightarrow N$

This approaches Ideal Overlap.

Look Ahead Parallel Processors.

Look ahead processors extend the idea of instruction pre-fetching to:

Performed in overlap fashion (overlap with D-unit)

- Instruction prefetch (Instruction cache/buffer)
- Advance opcode decoding (Opcode buffer/cache)
- Operand address calculating in advance (Address Cache)
- Advance operand fetch (Operand cache)

- Multiple D-units

Ex// IBM 370/165 (Next Class...)