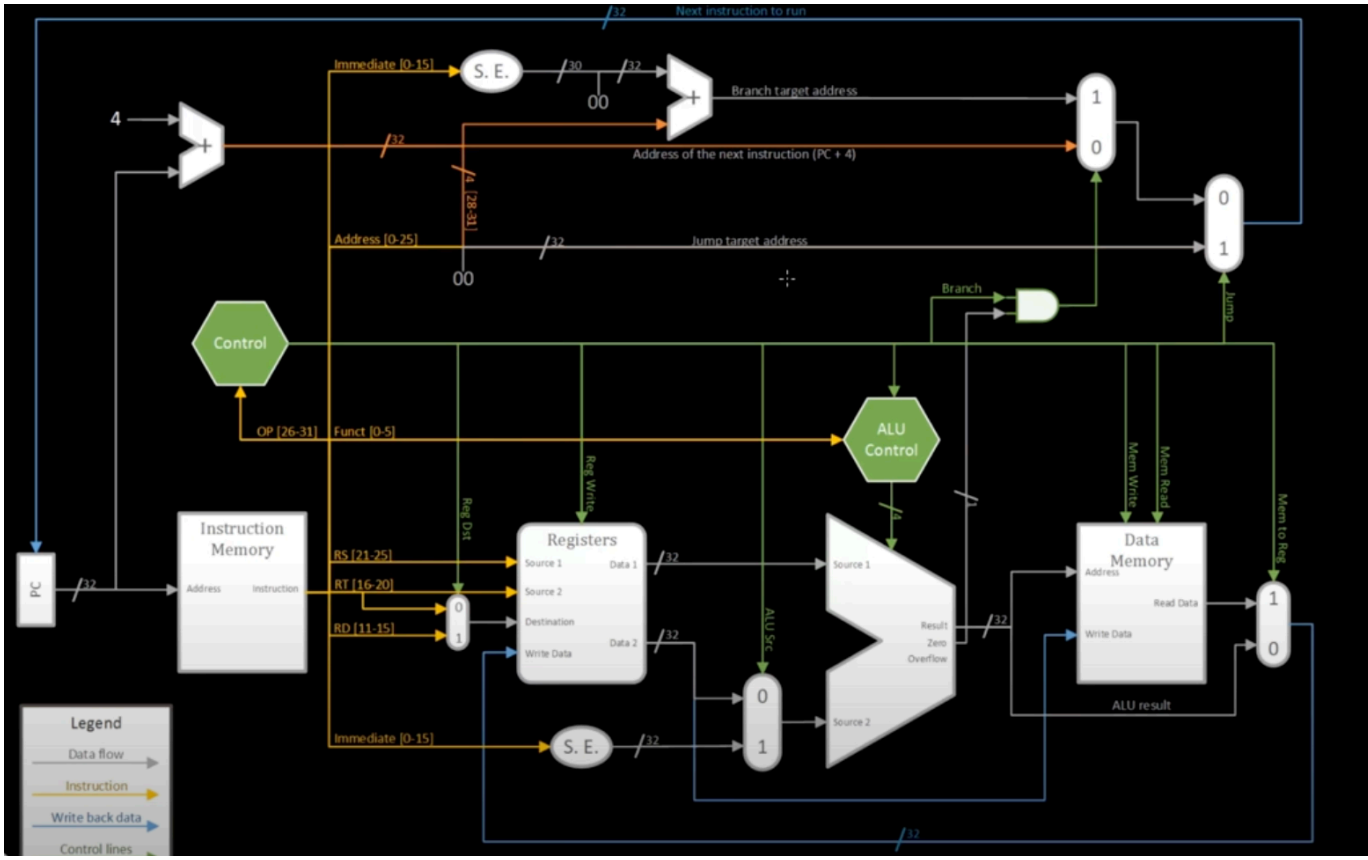
Datapath

Saturday, 19 August 2023 15:15

- Il datapath è l'unita che esegue le istruzioni assembly di un computer
- Esistono 2 tipologie di datapath:
  - Singolo ciclo
    - Risolve le istruzioni in 1 singolo ciclo
      - Questo però crea un problema; per risolvere tutto in 1 ciclo il clock d abbastanza basso da permettere a tutte le istruzioni di risolversi
      - Quindi si rallentano anche operazioni estremamente veloci per via d lente
    - Non viene richiesto in esame però consiglio vivamente di studiarlo per pot approfonditamente il multiciclo
    - Schema:



    - Qualche nota:
      - Le parentesi sono i bit dell'istruzione
      - Nientaltro in realtà, in caso di amnesia guarda qualche video online, semplice spiegato dalla giusta persona
  - Multiciclo
    - Risolve tutto in step, e ci sono vari step:

1) Instruction Fetch (IF)
    a) Si prendono le istruzioni dal PC e si portano all'istruction mem
        i) PCWrite = 0
        ii) IorD = 0
    b) L'istruzione è letto dalla memoria
        i) MemRead = 1
        ii) MemWrite = 0
    c) L'istruzione viene poi messa in un registro temporaneo: IR, ins
    d) Viene incrementato il PC di 4
        > Viene subito incrementato
        i) ALUSrcB = 1
        ii) PCSource=0
2) Instruction Decode
    a) Vengono presi i registri desiderati dal Register File (RF)
    b) Vengono messi nei registri A, B
    c) Decodifica il comando, aka inviamo l'operazione al controllo
        i) Verrà utilizzato nel ciclo successivo
    d) Nel mentre viene inviato il PC, i primi 15 bit vengono fatti il sig
    shiftati, viene calcolata la somma e messa nel registro ALUout
    fatto siccome è possibile che ci ritroviamo allo step successivo
    facendo questo risparmiamo tempo.
3) Execute
    Ora sappiamo quale operazione dobbiamo fare, ed abbiamo 4 casi:
    a) L'ALU deve fare quale tipologia di operazione (R type)
        i) L'ALU fà un operazione tra A e B
        ii) Si usa l'ALU Controll per dare ordini all'ALU
        iii) Ed esso lo sa grazie all'intrusction code [I primi 5 bit]
    b) Si fa il SW/LW
        i) L'ALU calcola la vera memoria aggiungendo l'OFFSET al
    c) E' un Branch
        i) L'indirizzo lo avevamo già calcolato prima
        ii) Calcoliamo la sottrazione tra A e B, se è 0 allora l'output
        si può fare la jump
        iii) Fine
    d) E' una jump
        i) Come il branch però PCWrite è attivo, quindi il valore de
        futile
        ii) Fine
4) Memory
    Questa capita solo per il caso a e b, i casi c e d sono già finiti
    a) R Type qui scriviamo nel registro il risultato

i) Il risultato viene scritto in memoria, il write data e regist attivati
ii) Fine

b) SW
i) Dobbiamo prendere dal registro alla memoria
ii) L'adress calcolato viene messo nella memoria
iii) Il write data arriva da B
iv) Fine

c) LW
i) Portiamo dalla memoria al registro
ii) L'address calcolato va in memoria, e sta volta viene lett
iii) Va dentro al memory data register

5) Write back
Qui manca solo LW/LB/LH
a) Il write register e write data vengono attivati
b) Viene scritto
c) Fine