

IP (INSTRUCTION POINTER) / IR (INSTRUCTION REGISTER)

giovedì 10 novembre 2022 12:07

INSTRUCTION POINTER

IP

- Nell'architettura di von Neumann dati e programmi si trovano entrambi in memoria di lavoro
- Per conoscere l'indirizzo della prossima istruzione da eseguire, il processore utilizza un registro fondamentale: l'**Instruction Pointer (RIP)**
 - In alcune architetture è chiamato **Program Counter (PC)**

Esso mantiene al suo interno un indirizzo di memoria: ossia l'indirizzo della prossima istruzione da eseguire.

Punterà in memoria, all'indirizzo della prossima istruzione macchina da eseguire.

PROCESSAMENTO DELLE ISTRUZIONI. MACRO FASI DELLA CPU:

1. FETCH

Serve a prelevare dalla memoria il codice macchina della prossima istruzione che dobbiamo andare ad eseguire.

Questo codice macchina descrive la codifica della nostra istruzione!

Da dove lo preleva? dall'indirizzo contenuto nell'INSTRUCTION POINTER;

RIP deve puntare ora a quella dopo ancora, e si deve aggiornare: **incremento il contenuto di RIP**. Lo incremento della dimensione in byte dell'istruzione che ho appena prelevato dalla memoria. In questo modo RIP punterà all'istruzione successiva;

PERÒ LE ISTRUZIONI IN MEMORIA DEVONO ESSERE CONTIGUE!

- Il flusso di esecuzione può non seguire il loro ordine in memoria (salti)

2. DECODE (IR CONTIENE IL CODICE MACCHINA DELL'ISTRUZIONE)

Se la nostra parola in codice macchina è una stringa di bit, dobbiamo decodificare cosa ci sta chiedendo quell'istruzione;

Se nella fase di fetch ho letto come codice macchina "somma" nella fase di decode la CPU deve sapere che voglio andare nella ALU.

Ci sarà qualche circuito che interpreterà i bit che sono contenuti all'interno dell'istruzione macchina per pilotare altri circuiti che implementeranno quella specifica istruzione;

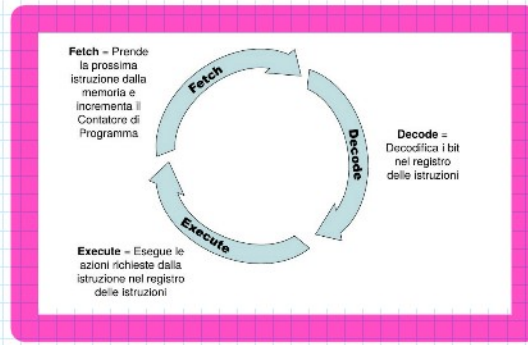
3. EXECUTE

L'unità di controllo, quindi tutti i circuiti che sono disponibili, per eseguire l'istruzione che abbiamo letto.

Program Counter (PC), ovvero il contatore delle istruzioni, contiene l'indirizzo della prossima operazione da eseguire.

Instruction Register (IR), il registro delle istruzioni, contiene l'istruzione da eseguire.

INSTRUCTION REGISTER
↑
b[IR]



IR ≠ IP
= PC

L'IR ABBATTE IL COLLO DI BOTTIGLIA DELL'ARCHITETTURA DI VON NEUMANN.

Accedere in memoria ha un costo elevato, voglio accedere una volta sola, e quando la rappresentazione binaria arriva all'interno del mio processore, me la servo da una parte!
↳ dell'istruzione macchina

IL PROGRAMMATTORE NON CI PUÒ SCRIVERE DIRETTAMENTE dentro, lo gestisce la CU, nella fase di fetch. La CU abilita alla scrittura o alla lettura i vari registri ed interpreta le istruzioni.

MANTENERE stabili i bit della rappresentazione binaria della mia istruzione PER TUTTO QUANTO
(L'IR) IL CICLO ISTRUZIONE!

- La fase di esecuzione può richiedere un numero variabile di cicli macchina, in cui può essere necessario accedere nuovamente in memoria
- Accedere ripetutamente in memoria è molto costoso
- Prima di essere eseguita, l'istruzione viene copiata (fase di fetch) in un registro fondamentale: l'Instruction Register (IR)

- Ciclo istruzione:** intervallo temporale necessario ad eseguire una istruzione nella sua interezza
- Ciclo macchina:** intervallo temporale necessario ad eseguire una fase (fetch, decode, execute)
 - A seconda del tipo di istruzione, possono essere necessari un numero diverso di cicli macchina
- Stato macchina:** periodo di tempo necessario per stabilizzare la rete dell'unità di calcolo (corrispondente al clock)

