

Struttura Processore

La CPU è un circuito integrato che fa da "**cervello**" del computer.

Capace di caricare ed eseguire le istruzioni elementari che costituiscono i programmi.(es. condizioni, cicli, salti ecc...)

E' formato da:

●**L'UNITÀ ARITMETICA-LOGICA (ALU)** esegue le operazioni aritmetiche e logiche necessarie ad eseguire le istruzioni

●**CIRCUITI DI CONTROLLO** generano i bit di controllo per gestire il funzionamento della CPU

●**BANCO DI REGISTRI:** blocco di memoria contenente i registri generici della CPU

●**PC e IR:** registri che contengono rispettivamente l'indirizzo della prossima istruzione e l'istruzione in esecuzione

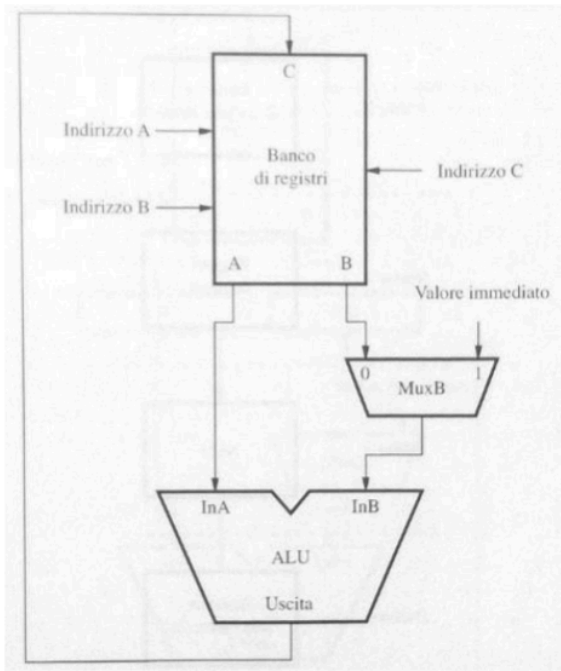
●**GENERATORE DI INDIRIZZI:** aggiorna il contenuto di PC

●**INTERFACCIA PROCESSORE MEMORIA** gestisce il trasferimento di dati tra memoria e CPU

Il banco dei registri è un blocco di memoria piccolo e veloce, e consiste in vari registri con circuiti di lettura e scrittura (lettura simultanea di 2 registri e scrittura su un terzo registro)

ALU (arithmetic logic unit) è il componente che si occupa di seguire operazioni aritmetiche (somma, sottrazione, AND, OR, XOR ecc...), ed ha 2 porte di input e una di

output.



GENERATORE DI INDIRIZZI: aggiorna il contenuto di PC con la prossima istruzione da eseguire

● **Per eseguire un'istruzione**, il processore deve eseguire i seguenti 3 passi:

1. **Prelievo dell'istruzione dalla memoria puntata da PC:** $IR \leftarrow PC$
2. **Incremento di PC di 4 unità (prossima istruzione):** $PC \leftarrow PC + 4$
3. **Esecuzione dell'istruzione prelevata**

● I primi due passi vengono chiamati fase di prelievo (**FETCH PHASE**)

● Il terzo passo è chiamato fase di esecuzione (**EXECUTION PHASE**)

● Durante la fase di **esecuzione si possono svolgere diverse azioni**: lettura/scrittura da/in una locazione di memoria, lettura da registri, esecuzione di operazioni aritmetiche e logiche, etc.

Come avviene il caricamento delle istruzioni?

Istruzioni di caricamento (FETCH-DECODE)

Load R5, X(R7)

1. Prelievo dell'istruzione ed incremento del PC
2. Decodifica dell'istruzione e lettura del contenuto del registro R7
3. Calcolo dell'indirizzo effettivo
4. Lettura dell'operando sorgente dalla memoria
5. Caricamento dell'operando nel registro di destinazione R5

Come si esegue un'istruzione aritmetico/logica?

Istruzioni di aritmetiche e logiche (EXECUTE)

Add R3, R4, R5

1. Prelievo dell'istruzione ed incremento del PC
2. Decodifica dell'istruzione e lettura dei contenuti dei registri sorgenti R4 e R5
3. Calcolo della somma $R4 + R5$
4. Caricamento del risultato nel registro di destinazione R3

Come avviene l'immagazzinamento delle operazioni?

Istruzioni di immagazzinamento (STORE)

Store R6, X(R8)

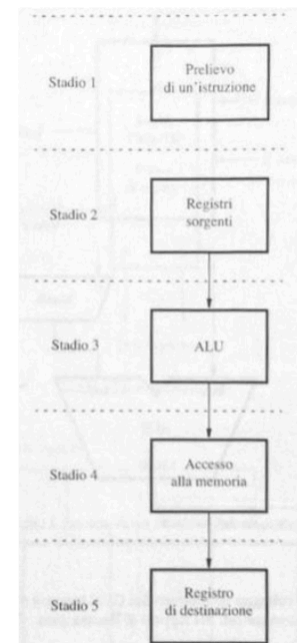
1. Prelievo dell'istruzione ed incremento del PC
2. Decodifica dell'istruzione e lettura dei registri R6 e R8
3. Calcolo dell'indirizzo effettivo $X + R8$
4. Immagazzinamento del contenuto di R6 nella locazione di memoria $X + R8$

Organizzazione a cinque stadi



.Tutte le operazioni possono essere eseguite in 5 stadi distinti (saltandone alcuni nel caso sia necessario):

1. Preleva un'istruzione e incrementa il contatore del programma
2. Decodifica l'istruzione e leggi i registri dal banco dei registri
3. Esegui un'operazione dell'ALU
4. Leggi o scrivi dati in memoria
5. Scrivi il risultato nel registro di destinazione



per temporizzare (coordinare) il trasferimento dei dati nel processore si usano i cicli di clock.

Il datapath di una CPU è il percorso che fanno i dati per essere elaborati. E' costituito da una serie di componenti interconnessi che lavorano insieme per una corretta elaborazione. I dati recuperati dai registri vengono elaborati nelle ALU e il risultato viene immagazzinato nel registro di output.

.Stadio 2:

•Le porte di uscita A e B del banco di registri vengono copiate nei registri temporanei RA e RB

.Stadio 3:

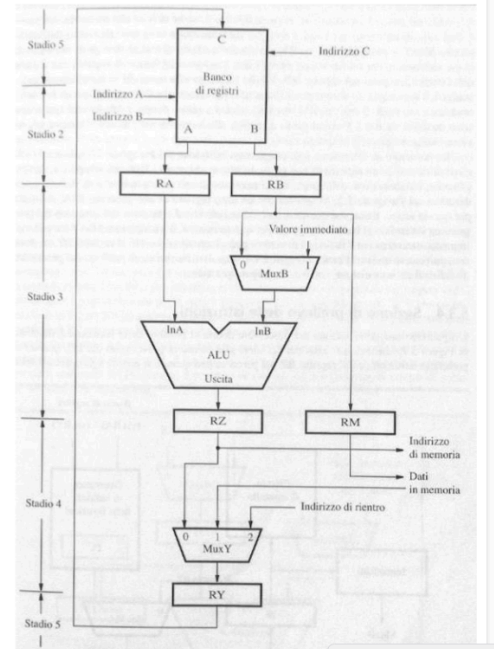
- RA viene usato come primo ingresso dell'ALU
- MuxB sceglie RB o un valore immediato come secondo ingresso
- Il risultato dell'ALU viene copiato in RZ
- RB viene copiato in RM

.Stadio 4:

- Se necessario l'indirizzo in RZ viene mandato all'interfaccia processore memoria
- Se necessario i dati in RM vengono salvati in memoria
- MuxY sceglie se salvare su RY il risultato di un'operazione, dei dati della memoria o l'indirizzo di rientro da sottoprogramma

.Stadio 5:

- Il contenuto di RY viene salvato nel Banco dei registri



(per guardare degli esempi vai nel power point del processore da pag. 15 a 23 [5.3](#)

[Struttura-base-del-processore.pdf](#))

Attesa della memoria

●Non sempre gli accessi alla memoria possono essere eseguiti in un ciclo di clock

●Se il dato o l'istruzione da prelevare non si trovano nella cache, l'esecuzione deve bloccarsi al passo corrente fintanto che l'operazione di memoria richiesta non è stata eseguita

●Ad operazione di memoria eseguita viene generato il segnale **MFC (memory function completed)**

●Il circuito di controllo interrompe l'esecuzione dell'istruzione finché MFC non diventa uguale a 1

●L'attesa di MFC avviene nel primo passo di ogni istruzione (prelievo istruzione dalla memoria) e nel passo 4 delle istruzioni di load e store

Per eseguire le istruzioni macchina il processore deve generare le sequenze di segnali di controllo per ogni stadio

I segnali di controllo consistono in:

- Segnali di selezione per i multiplatori
- Segnali di attivazione di alcuni registri
- Segnali di condizione

- Segnali per la gestione della memoria
- Indirizzi, codice operativo e dati letti dall'istruzione nel registro IR
- Operazione da eseguire nella ALU

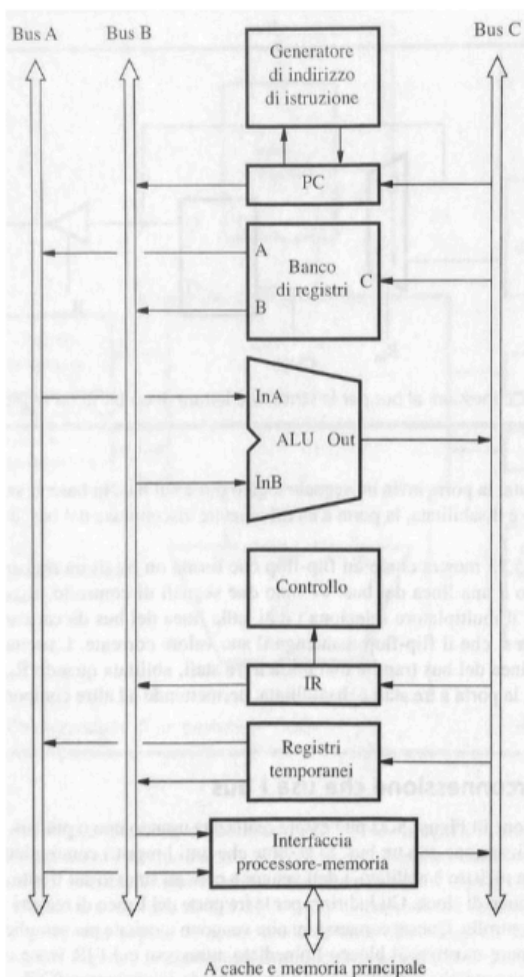
L'attesa per i segnali di controllo si verifica quando la CPU aspetta che un segnale arriva alla sua destinazione, in questo tempo la CPU resta ferma, questo chiaramente influisce sulle prestazioni

Un approccio per generare i segnali di controllo consiste nel **CONTROLLO CABLATO**

Il controllo cablato di una CPU è un metodo le cui operazioni vengono gestite a livello hardware, in pratica i segnali che controllano la CPU sono segnali predefiniti e non cambiano mai, in base a questi segnali la CPU esegue varie istruzioni. Il controllo su queste cose vengono fatte dalla Central Unit (CU)

Le istruzioni CISC possono occupare più di una parola, non è possibile un approccio a stadi come in RISC

quindi si utilizza un **blocco di interconnessione** per trasmettere dati tra i componenti, questo blocco è realizzato dal BUS, quindi il BUS connette vari dispositivi, ma solo 1 dispositivo alla volta può mandare un segnale nel BUS e per questo si utilizzano 3 linee di BUS 2 per gli operandi e 1 per il risultato



(per guardare degli esempi vai nel power point del processore da pag. 32 a 42 [5.3 Struttura-base-del-processore.pdf](#))

Controllo microprogrammato (sempre per i segnali)

- I segnali di controllo di ogni passo vengono raccolti in una parola di memoria chiamata microistruzione
- L'insieme di microistruzioni rappresentanti i passi di un'istruzione macchina si chiamano microroutine
- Le microistruzioni di ciascuna microroutine vengono immagazzinate in locazioni consecutive della memoria di controllo
- Il registro μPC contiene l'istruzione della prossima microistruzione da caricare
- All'inizio di un'istruzione macchina il generatore di indirizzi delle microistruzioni carica sul μPC la prima istruzione della microroutine corrispondente
- Ogni passo μPC viene incrementato di un passo