

I Processori

M. Sonza Reorda

Politecnico di Torino
Dip. di Automatica e Informatica

1

M. Sonza Reorda - a.a. 2001/02

Introduzione

Un processore è un dispositivo che compie 2 tipi di operazioni:

- **esegue istruzioni contenute in memoria**
- **interagisce con il mondo esterno (attraverso opportune interfacce).**

2

M. Sonza Reorda - a.a. 2001/02

Gestione di Eventi Esterni

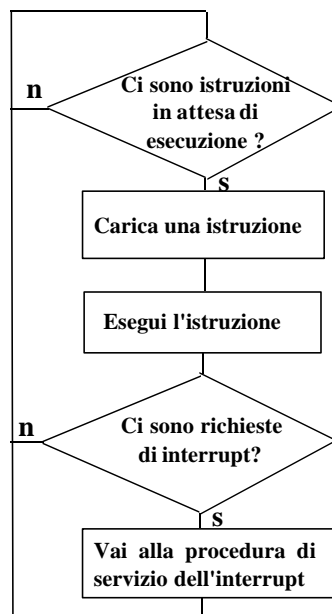
Il processore partecipa normalmente a tutte le operazioni che si svolgono sul bus del sistema, ad alcune direttamente, ad altre semplicemente partecipando al meccanismo di arbitraggio del bus.

Particolare importanza ha il meccanismo dell'*interrupt*, attraverso il quale un dispositivo esterno richiede l'esecuzione di una procedura di servizio.

3

M. Sonza Reorda - a.a. 2001/02

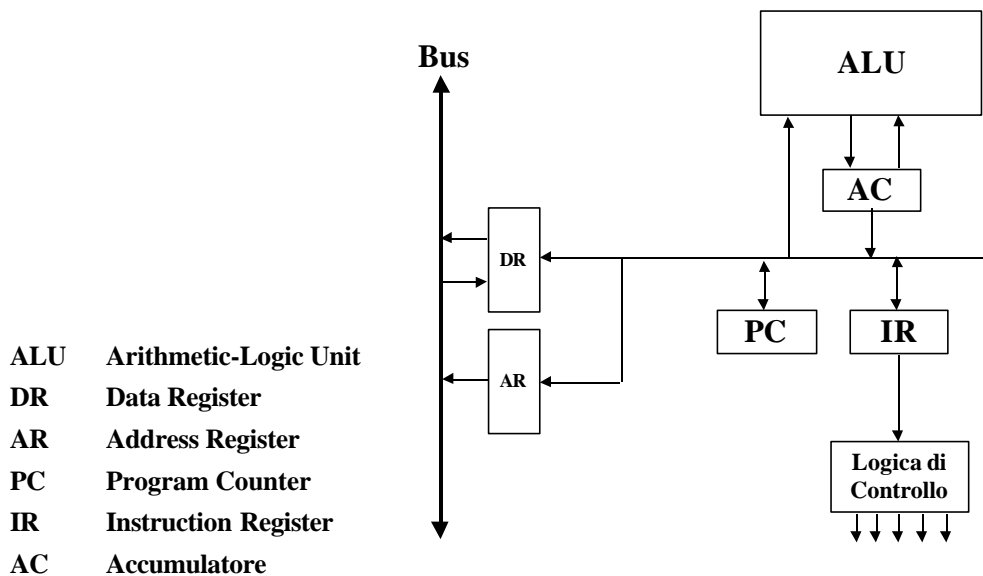
Comportamento di un Processore



4

M. Sonza Reorda - a.a. 2001/02

Architettura di una CPU



5

M. Senza Reorda - a.a. 2001/02

Istruzioni

L'esecuzione di ciascuna istruzione macchina si compone di 2 fasi:

- *fetch*: il codice dell'istruzione viene letto dalla memoria
- *execute*: il codice viene prima decodificato, poi eseguito. Questo comporta normalmente l'accesso ad uno o più operandi, l'esecuzione di una operazione su di essi, la scrittura del risultato.

La somma delle due fasi si dice *ciclo di istruzione*.

6

M. Senza Reorda - a.a. 2001/02

Memoria e Registri

La memoria contiene il programma eseguito dalla CPU.

Si definisce *Tempo di Ciclo della Memoria* (t_{mem}) il tempo che deve intercorrere tra 2 accessi successivi ad essa.

Il rapporto t_{mem}/t_{CPU} si aggira attorno a 10. Per questa ragione all'interno della CPU sono presenti alcune celle di memoria particolarmente veloci, note come *registri*.

La velocità di esecuzione di un programma dipende fortemente da quante volte la CPU deve fare accesso alla memoria. Ove possibile le operazioni vengono svolte utilizzando i registri per contenere operandi e risultato.

7

M. Senza Reorda - a.a. 2001/02

Microistruzioni

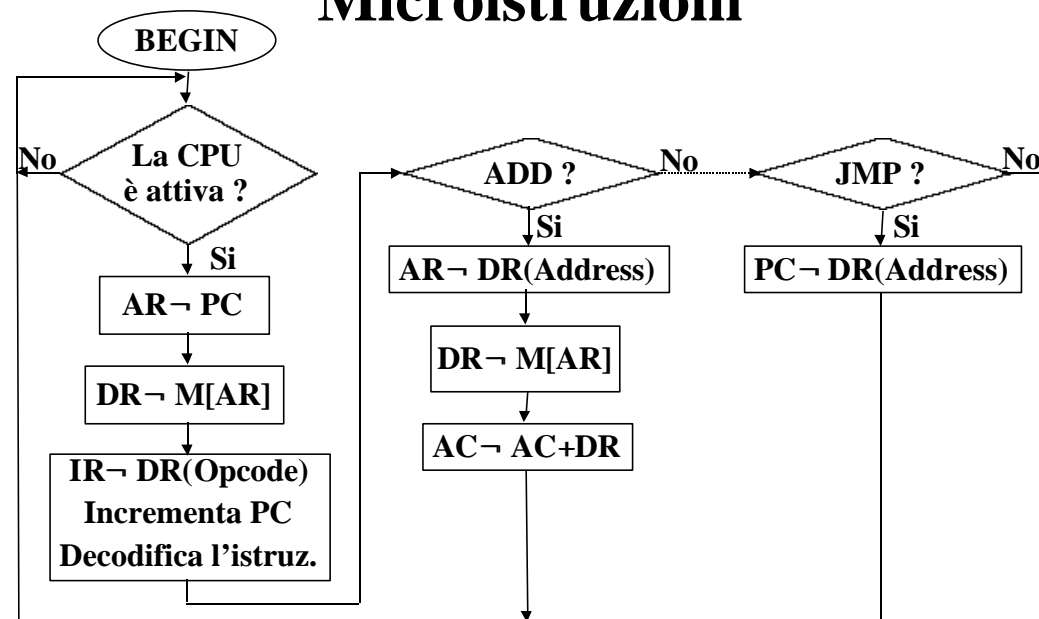
L'esecuzione delle 2 fasi di una istruzione corrisponde all'attivazione di un certo numero di *microistruzioni* (corrispondenti ad esempio al trasferimento di un valore da un registro ad un altro).

Il tempo per l'esecuzione di una microistruzione è definito *CPU cycle time* (t_{CPU}).

8

M. Senza Reorda - a.a. 2001/02

Microistruzioni



9

M. Senza Reorda - a.a. 2001/02

Input/Output

Viene realizzato in maniera simile a quanto fatto per l'accesso alla memoria: ad ogni dispositivo di I/O vengono associati uno o più indirizzi, sui quali la CPU esegue opportune operazioni di lettura o scrittura.

Tali indirizzi possono appartenere allo stesso spazio di indirizzamento della memoria (*memory-mapped I/O*), oppure appartenere ad uno spazio di indirizzamento separato (*I/O-mapped o port-addressed I/O*, o *isolated I/O*). Nel primo caso l'accesso ai dispositivi avviene tramite normali istruzioni, nel secondo invece si utilizzano istruzioni particolari (IN e OUT).

10

M. Senza Reorda - a.a. 2001/02

Memory Mapped e Isolated I/O

Memory Mapped

- Unico spazio di indirizzamento
- Stesse istruzioni per accesso a memoria e I/O

Isolated I/O

- 2 spazi di indirizzamento
- Apposito segnale di controllo per distinguere tra cicli di accesso a memoria o a periferici
- Istruzioni speciali per I/O (IN e OUT)

11

M. Senza Reorda - a.a. 2001/02

Estensioni

Rispetto allo schema di base di un processore sono possibili vari miglioramenti relativi ai seguenti punti:

- organizzazione dei registri
- unità aritmetiche
- status register
- stack
- pipeline.

12

M. Senza Reorda - a.a. 2001/02

Organizzazione dei registri

Il numero dei registri influenza fortemente le prestazioni di un processore. Quando sono molti si parla di *register-file* o *scratch-pad memory*.

È possibile che per ogni registro venga imposto un particolare uso; si possono avere

- registri dati
- registri indice
- registri contatore.

13

M. Sonza Reorda - a.a. 2001/02

Unità Aritmetiche

Possono essere più o meno potenti, a seconda delle operazioni che implementano:

- somma e sottrazione in fixed-point
- moltiplicazione e divisione in fixed-point
- operazioni in floating-point.

14

M. Sonza Reorda - a.a. 2001/02

Status Register

È un registro particolare contenente due gruppi di bit.

Il primo gruppo (*status flag*) corrisponde a bit che sono

- forzati dal verificarsi di particolari condizioni (*carry*, *overflow*, segno positivo o negativo, risultato nullo, parità) a seguito delle esecuzioni delle istruzioni
- testati dalle istruzioni di salto condizionato (e da eventuali altre).

Il secondo gruppo (*control flag*) è composto da bit che contengono informazioni sul modo di funzionamento del sistema (ad esempio sensibilità agli interrupt) e sono di solito direttamente modificabili dal programmatore.

Stack

È una struttura LIFO allocata in memoria.

Molti processori possiedono:

- un registro special-purpose, denominato *Stack Pointer* (SP) che punta all'elemento top
- due istruzioni (PUSH e POP) che manipolano implicitamente lo SP.

Stack: usi

Lo stack viene usato:

- per le chiamate alle procedure
- per le procedure di servizio dell'interrupt
- per il passaggio di parametri alle procedure
- per la memorizzazione di variabili locali.

In taluni casi (IBM 360-370) i primi due problemi vengono risolti utilizzando una particolare locazione di memoria. In tal caso non è possibile la recursione.

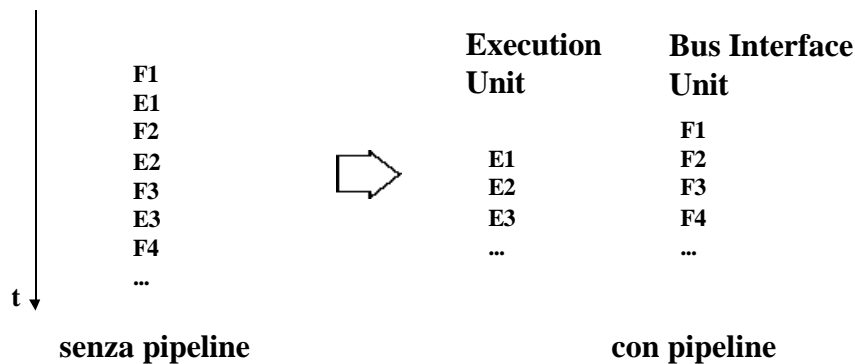
17

M. Senza Reorda - a.a. 2001/02

Pipeline

Molti processori hanno un'architettura a moduli, dove i moduli possono lavorare in parallelo.

In tal modo è possibile eseguire in parallelo le operazioni di *fetch* ed *execute*.



18

M. Senza Reorda - a.a. 2001/02