

# Laboratorio di Informatica per chimica industriale e chimica applicata e ambientale

## LEZIONE 4

### La CPU e l'esecuzione dei programmi

AA 2002/2003  
© Morpurgo, Zanaboni

1

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## Nelle lezioni precedenti abbiamo detto che

Un computer è costituito da 3 principali componenti:

- Il processore
- La memoria
- Le periferiche

Interconnesse attraverso il bus

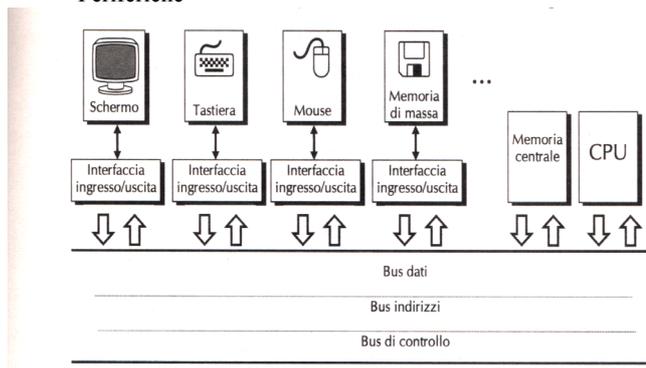
AA 2002/2003  
© Morpurgo, Zanaboni

2

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## Struttura logica

- processore (CPU: Central Processing Unit)
- memoria centrale ad accesso diretto (RAM: Random Access Memory)
- Canale di comunicazione (bus)
- Periferiche

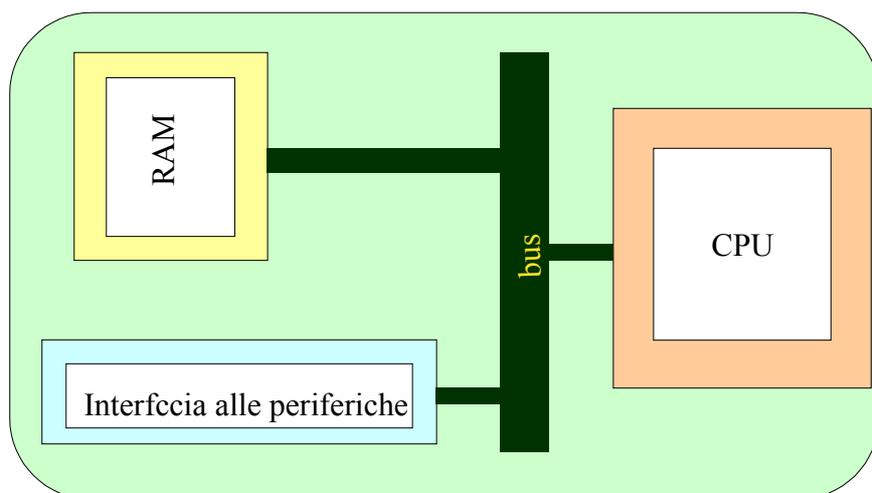


AA 2002/2003  
© Morpurgo, Zanaboni

3

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## Struttura fisica



AA 2002/2003  
© Morpurgo, Zanaboni

4

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

# PROGRAMMI e LINGUAGGIO MACCHINA

AA 2002/2003  
© Morpurgo, Zanaboni

5

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## Programmi

- I programmi sono sequenze di istruzioni.
- I programmi eseguibili da un computer devono essere scritti usando un *linguaggio (di programmazione)* che il computer è in grado di “comprendere”.

AA 2002/2003  
© Morpurgo, Zanaboni

6

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

**Computer** = macchina che svolge rapidamente operazioni elementari

**Programma** = insieme di istruzioni che indicano come svolgere operazioni complesse su dei dati attraverso successioni di operazioni elementari **specificate** in un linguaggio “comprensibile” dal computer.

**Dato** = informazione da elaborare **rappresentata** in un formato che consenta al programma di operare su di essa

## LA CPU

## Il processore - chip

- Il processore è l'unità di elaborazione centrale (CPU: *central processing unit*) del computer.
- Oggi le CPU sono costituite da circuiti che contengono milioni di microscopici interruttori acceso/spento (i transistor) collegati tra loro da sottilissimi fili.
- I circuiti vengono “stampati” su una sottile lamina di silicio. Il *chip* che corrisponde a una CPU ha una superficie dell'ordine dei mm<sup>2</sup>.

## Il processore - chip

- Pertanto oggi si parla di *microprocessore* costituito da circuiti *integrati*: milioni di elementi tra loro collegati contenuti in una superficie quasi microscopica.

## Il processore



## Il linguaggio macchina

- Il processore è in grado di riconoscere (e quindi di eseguire) solo programmi scritti in un proprio linguaggio (*linguaggio macchina*)
- Ogni modello di processore (es: Intel, Pentium, Motorola, PowerPC) ha un proprio linguaggio macchina diverso da quello degli altri processori

## Il processore - clock

- In corrispondenza di ogni istruzione espressa nel linguaggio macchina il processore svolge una serie di operazioni elementari (il ciclo della macchina).
- Il numero operazioni elementari svolte per ogni istruzione dipende dal tipo di processore (tra le 7 e le 10).

## Il processore - clock

- Il tempo impiegato dal processore per eseguire una operazione elementare è detto *durata del ciclo di clock*
- Il numero di operazioni elementari svolte dal processore in una unità di tempo (cioè il numero di cicli di clock al secondo) è la *frequenza di clock*

## Il processore - clock

Quindi:

$$\text{Frequenza di clock} = \frac{1}{\text{Durata di un ciclo di clock}}$$

E' un indicatore della "velocità" del processore nell'eseguire i programmi.

Oggi si misura in Mhz (MegaHertz).

1 Mhz = 1 milione di cicli al secondo.

## Il processore - clock

- Dati due processori con lo stesso linguaggio macchina sarà più veloce quello con frequenza di clock maggiore
- Non è possibile fare un confronto tra processori con linguaggi macchina diversi

## Il processore - clock

Esempio

### Computer A

**Clock :** 500MHz

**Linguaggio macchina**  
richiede 10 operazioni  
elementari per istruzione

### Computer B

**Clock :** 400MHz

**Linguaggio macchina**  
richiede 6 operazioni  
elementari per istruzione

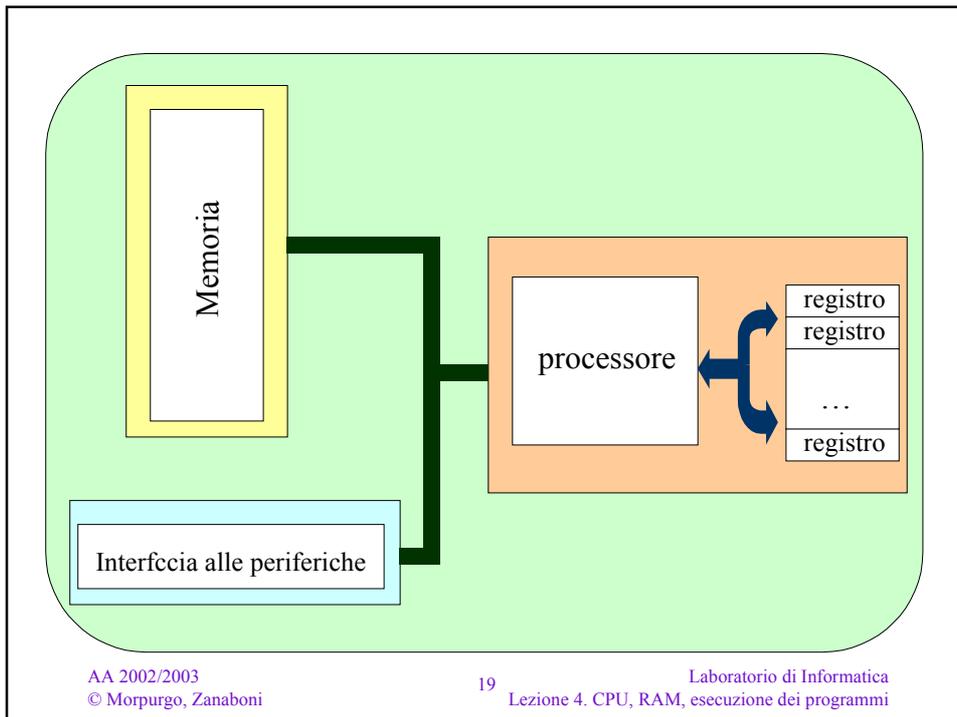
Devono eseguire un programma di 100M istruzioni macchina

**A impiegherà 2 sec.**

**B impiegherà 1,5 sec.**

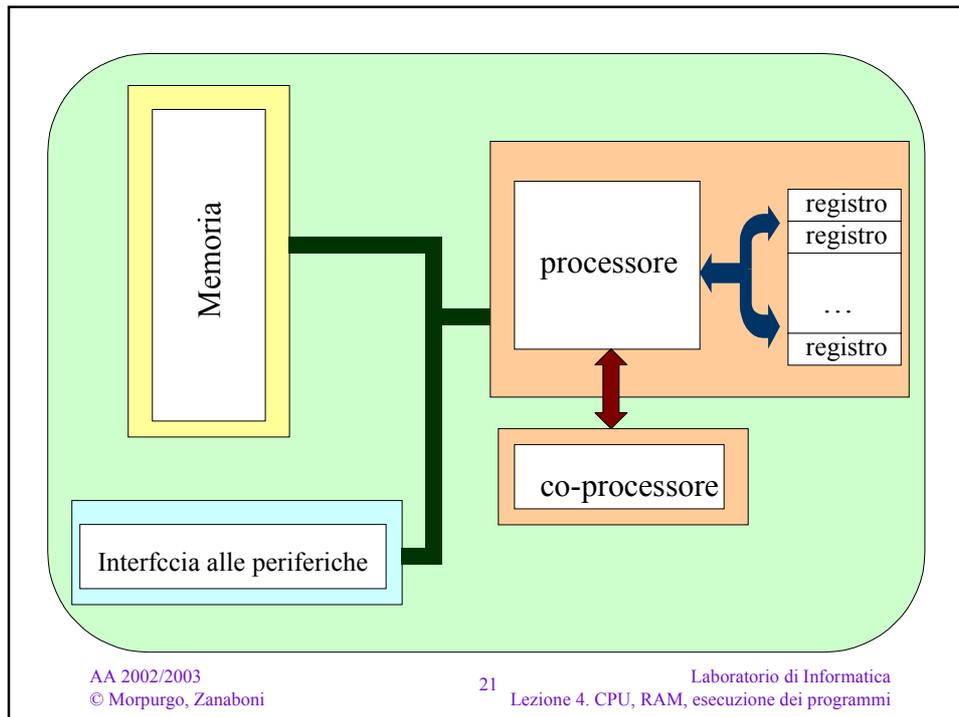
## Il processore - registri

- Per eseguire le operazioni elementari il processore ha bisogno di utilizzare delle memorie (di piccole dimensioni) alle quale accedere in modo veloce (es: una zona che contiene l'indirizzo dell'istruzione da eseguire, una per i dati su cui l'istruzione opera)
- Queste memorie, che fisicamente sono contenute nel chip, sono chiamate *registri* del processore



## Sistemi con più processori I co-processor

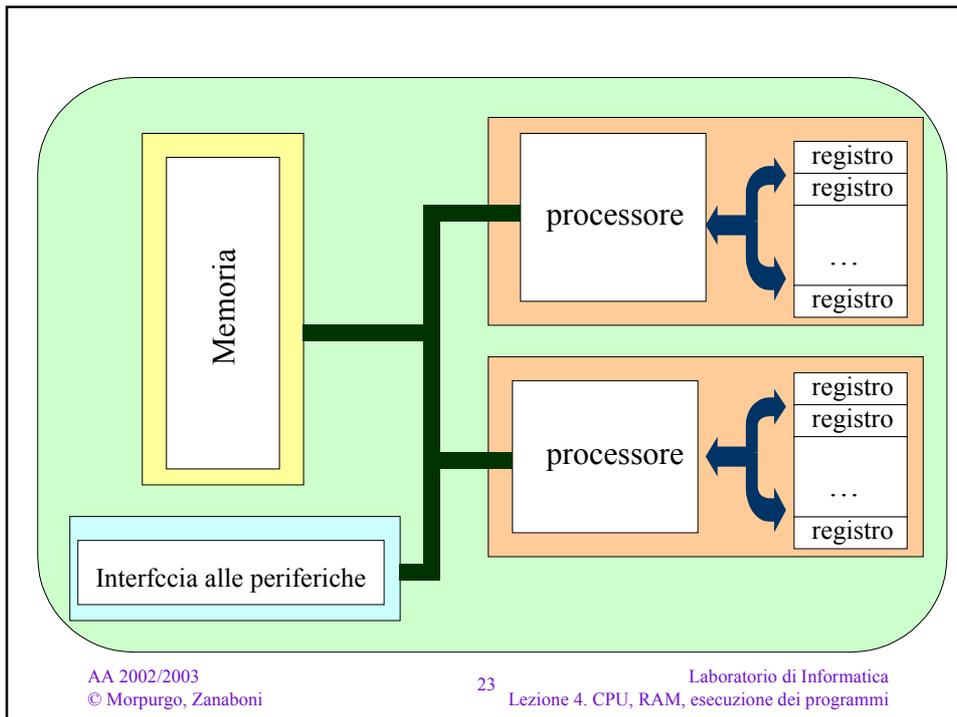
- A seconda del tipo di uso che l'utente fa del computer, per velocizzarne le prestazioni si possono aggiungere dei co-processor.
- Questi sono processori specializzati nell'esecuzione di compiti specifici (es: gestione dello schermo) che operano sotto il controllo della CPU.



## Sistemi con più processori

### Sistemi paralleli

- I sistemi paralleli sono sistemi con più CPU che operano in modo indipendente e condividono le altre risorse del computer.
- Il coordinamento dei diversi processori è demandato al sistema operativo.
- La velocità del computer risulta aumentata ma non moltiplicata, perché non è trascurabile il tempo che il sistema operativo deve dedicare al coordinamento dei processori



# LA MEMORIA

AA 2002/2003  
© Morpurgo, Zanaboni

24

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## La memoria

- La memoria è la componente che svolge la funzione di magazzino delle informazioni sulle quali opera il processore
- L'unità di informazione minima è il *bit* (*binary unit*). Consideriamola come una casella nella quale possiamo scrivere il simbolo 0 oppure il simbolo 1.

## La memoria-il bit

- Avendo a disposizione un solo bit si possono rappresentare due elementi diversi:  
Si assegna al primo elemento la codifica 0 e al secondo la codifica 1
- Avendo a disposizione due bit si possono rappresentare quattro elementi diversi, assegnando a ciascuno una codifica diversa:  
00, 01, 10, 11

## La memoria-il bit

- Avendo a disposizione tre bit si possono rappresentare otto elementi diversi.
- ....
- Avendo a disposizione n bit si possono rappresentare  $2^n$  elementi diversi.

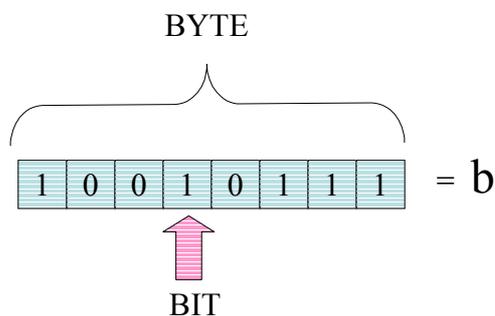
## La memoria-il byte

- Agli albori dell'informatica c'era l'esigenza di rappresentare i caratteri stampabili (le lettere dell'alfabeto, le cifre da 0 a 9, lo spazio, la virgola etc...) per un numero totale di caratteri compreso tra 90 e 120.
- Per poter rappresentare 128 elementi diversi servono almeno 7 bit ( $2^7=128$ ).
- E' bene avere a disposizione un bit supplementare poter effettuare controlli sulla sequenza

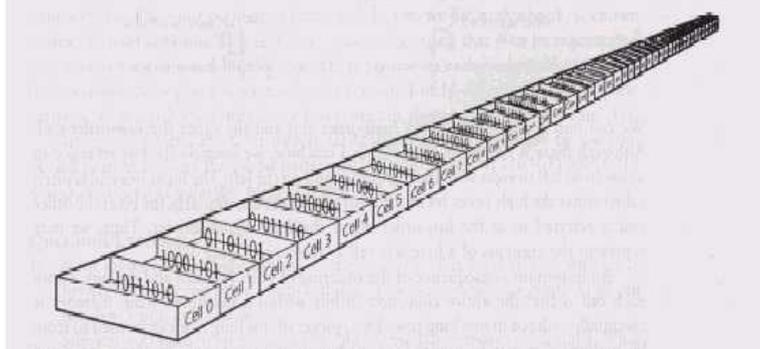
## La memoria-il byte

- Quindi per rappresentare i caratteri stampabili è necessaria una sequenza di  $7+1=8$  bit.
- Una sequenza di 8 bit è detta *byte*, ed è diventata una unità di misura della occupazione di memoria.

## La memoria-il byte



## Le celle di memoria



AA 2002/2003  
© Morpurgo, Zanaboni

31

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## Le celle di memoria

- Ogni cella di memoria contiene un byte, ed è contraddistinta da un indirizzo
- Volendo *scrivere* o *leggere* un dato dalla memoria il processore deve sempre specificare l'indirizzo della cella alla quale vuole accedere

AA 2002/2003  
© Morpurgo, Zanaboni

32

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

## La memoria-unità di misura

Le dimensioni di una memoria si misurano in:

- Kbyte (kilo byte)=  $2^{10}$  byte = 1024 byte ;  $10^3$  byte
- Mbyte (Mega byte)=  $2^{10}$  Kbyte = 1024 Kbyte=  $2^{20}$  byte ;  $10^6$  byte
- Gbyte (Giga byte)=  $2^{10}$  Mbyte =1024 Mbyte =  $2^{30}$  byte ;  $10^9$  byte
- Tbyte (Tera byte)=  $2^{10}$  Gbyte =1024 Gbyte =  $2^{40}$  byte ;  $10^{12}$  byte

## L'ESECUZIONE DEI PROGRAMMI

## Quando chiediamo al computer di eseguire un programma accade che:

- Il programma (già scritto in linguaggio macchina) viene *trasferito* nella memoria di lavoro (la *memoria centrale*), alla quale il processore può accedere velocemente
- Per ogni istruzione del programma la CPU esegue tre tipi di operazioni, che costituiscono il *ciclo della macchina*:

## Ciclo della macchina

- Lettura (*fetch*)– il processore reperisce dalla memoria l'istruzione da eseguire
- Decodifica (*decode*) – il processore identifica l'istruzione da eseguire
- Esecuzione (*exec*) – il processore esegue le operazioni corrispondenti all'istruzione

## L'esecuzione dei programmi

La **memoria centrale** è l'area di lavoro:

- contiene i programmi in esecuzione in quel momento ed i relativi dati
- in particolare, contiene sempre la parte del sistema operativo che governa l'utilizzo delle risorse di calcolo

L'unità fisica che esegue i programmi in memoria centrale è la **CPU**

## L'esecuzione dei programmi

- Si dice che un programma ha il controllo della CPU se la CPU sta eseguendo quel programma;
- impropriamente, diremo che:
  - un programma **P** svolge un compito
    - per dire che
  - la **CPU** svolge quel compito sotto il controllo di **P**

## **L'esecuzione dei programmi del Sistema Operativo (SO)**

- Quando il calcolatore è in attesa di comandi, il controllo della CPU è detenuto dal SO, che:
  - gestisce l'interfaccia utente
  - accetta i comandi dell'utente
  - avvia (lancia) l'esecuzione dei programmi applicativi

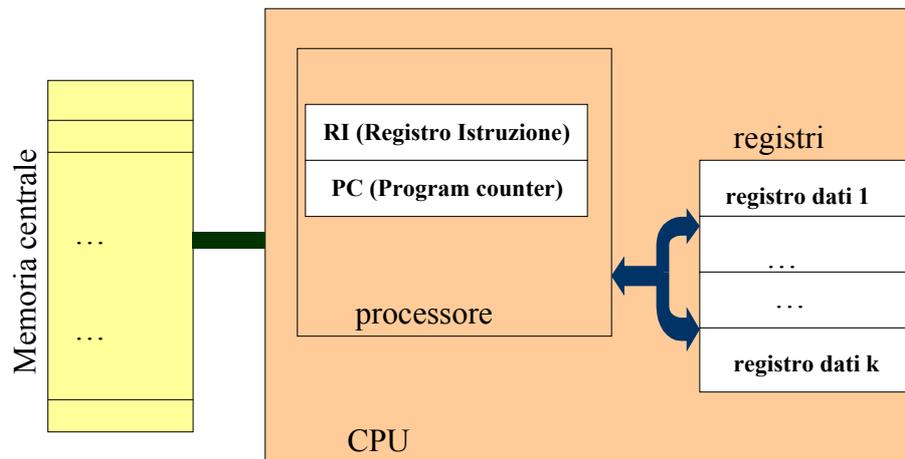
## **L'esecuzione di un programma applicativo**

- La CPU è sotto il controllo del SO, che gestisce l'interfaccia utente
- L'utente chiede, tramite l'interfaccia, di eseguire, ad esempio, WORD
- La CPU, sotto il controllo del SO, riconosce la richiesta e
  - carica (trasferisce) WORD in memoria centrale
  - inizia ad eseguire WORD
- L'esecuzione di WORD termina e la CPU riprende ad eseguire il SO;
- nuovamente sotto il controllo del SO, la CPU è pronta ad accettare nuovi comandi

## Ma la situazione è più complessa

- In realtà il SO gestisce l'esecuzione di più programmi e più utenti, per cui
  - si hanno più processi e il SO concede la CPU a ciascuno di essi, per piccoli intervalli di tempo, a rotazione o secondo altre politiche
  - tutto ciò è trasparente all'utente e non ce ne occuperemo in questo corso

## Funzionamento della CPU



## La nostra CPU finta

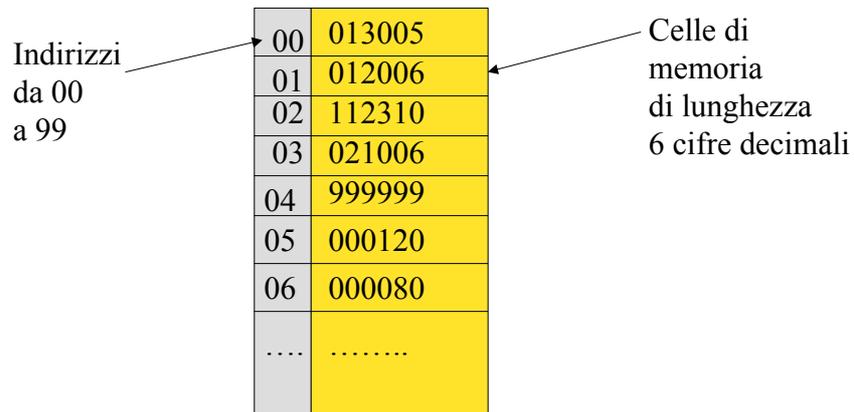
- Una CPU può leggere/scrivere il contenuto di una sola cella di memoria alla volta, specificandone l'indirizzo;
- contiene inoltre dei **registri**, come **area di lavoro veloce**
- la nostra CPU finta contiene i seguenti registri
  - il registro **PC** (Program counter) che contiene l'indirizzo dell'istruzione da reperire
  - il registro **RI**, (Registro istruzioni) che contiene l'istruzione da decodificare ed eseguire
  - **4 registri dati**
- il ciclo di esecuzione, nelle sue fasi, avviene come segue

## Ciclo della macchina

- La CPU esegue le istruzioni in memoria centrale ripetendo i seguenti passi (**ciclo della macchina**):
  - **1) Lettura (*fetch*)**
    - reperisce dalla memoria centrale l'istruzione da eseguire e la carica nel registro istruzioni **RI**
  - **2) Decodifica (*decode*)**
    - riconosce l'istruzione presente nel **RI**;
  - **3) Esecuzione (*exec*)**
    - esegue le operazioni dettate da tale istruzione
  - **4) Torna al passo 1)**
    - per la lettura della prossima istruzione

## Esempio con una CPU finta, che usa la numerazione decimale

La nostra CPU finta usa una **memoria centrale** indirizzata da 2 cifre decimali, cioè di 100 elementi



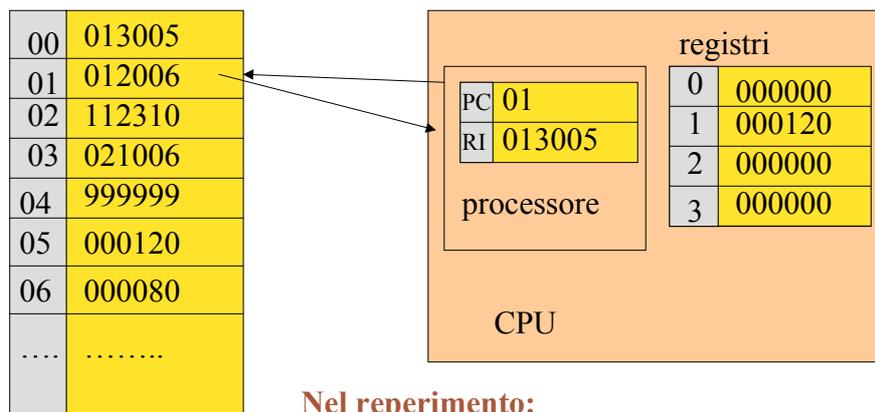
AA 2002/2003  
© Morpurgo, Zanaboni

45

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

### Quando inizio il **reperimento** il **PC** contiene l'**indirizzo** dell'istruzione **da eseguire in quel momento**

In questo esempio è l'istruzione di indirizzo 01



### Nel **reperimento**:

la CPU trasferisce l'istruzione da eseguire (di indirizzo 01) nel RI e si ottiene:

AA 2002/2003  
© Morpurgo, Zanaboni

46

Laboratorio di Informatica  
Lezione 4. CPU, RAM, esecuzione dei programmi

