

Abilità Informatiche

Luigi Catuogno

[luigi.catuogno@uniparthenope.it]

Corso di Laurea in Economia e Commercio - Anno Accademico 2022-23

1

Libro di testo

[IdB]

Dennis P. Curtin, Kim Foley, Kunal Sen, Cathleen Morin

Informatica di base

VII edizione (2016), MacGraw Hill Education

ISBN: 978-88-386-1537-5

2

Altro materiale di utile consultazione

[Sli]

Slides, appunti e altro materiale distribuito dal docente

[Misc]

Altra fonte diversamente specificata di volta in volta

3

Architettura del calcolatore

Il modello di Von Neumann

4

Il modello di Von Neumann



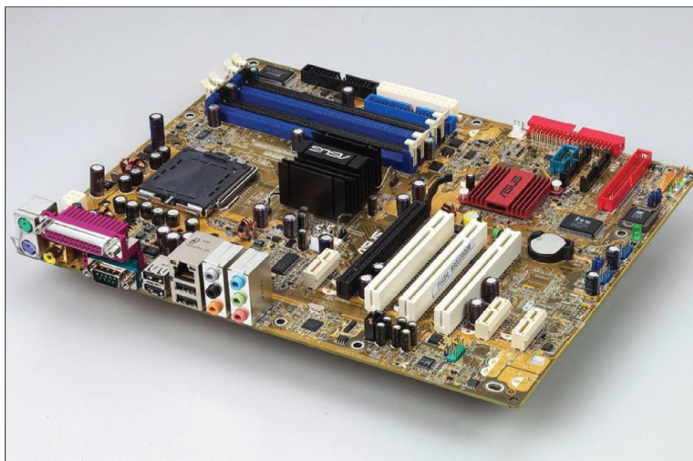
John Von Neumann 1903-1957

- Modello architetturale di uno strumento di calcolo «reale»
 - Introduce un modello per l'organizzazione e il funzionamento di un calcolatore
 - Proposto nel 1945 dal matematico e fisico di origini ungheresi John Von Neumann
 - Concetto «Derivato» dalla Macchina di Turing

5

Il modello di Von Neumann

A differenza della MdT (che è un modello astratto), Von Neumann descrive le componenti funzionali di un calcolatore, il mezzo e i modi delle loro interazioni in termini molto concreti

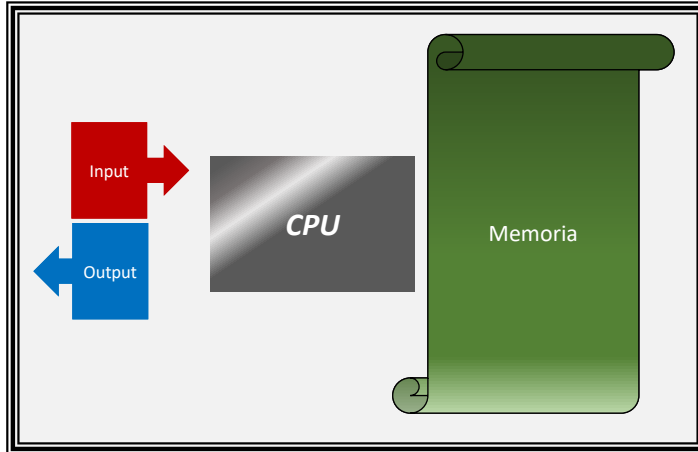


In quasi tutti i calcolatori oggi in commercio, sono attuate le indicazioni date dal modello e si possono trovare tutte le componenti che prevede: CPU, I/O, memoria, BUS...

6

Il modello di Von Neumann

A differenza della MdT (che è un modello astratto), Von Neumann descrive le componenti funzionali di un calcolatore, il mezzo e i modi delle loro interazioni in termini molto concreti

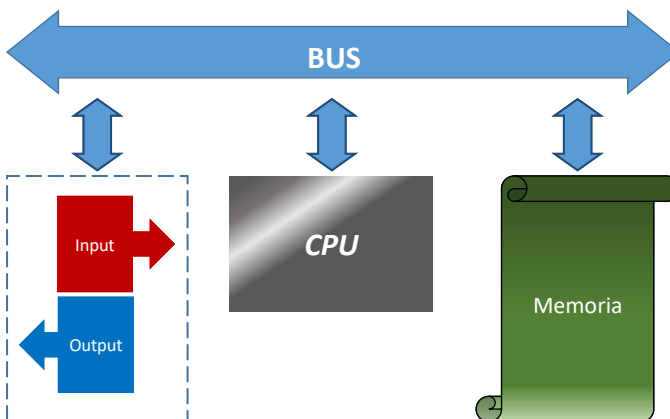


In quasi tutti i calcolatori oggi in commercio, sono attuate le indicazioni date dal modello e si possono trovare tutte le componenti che prevede: CPU, I/O, memoria, BUS...

7

Il modello di Von Neumann

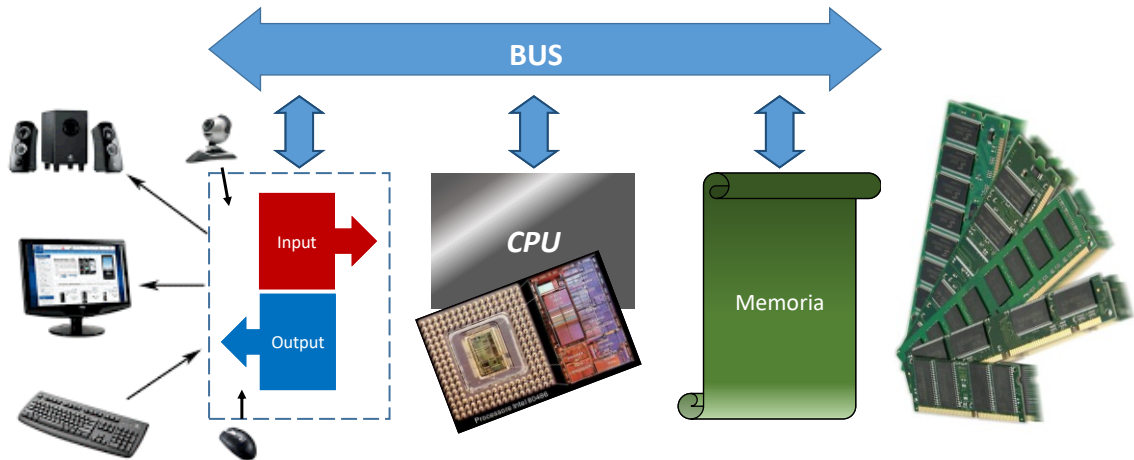
A differenza della MdT (che è un modello astratto), Von Neumann descrive le componenti funzionali di un calcolatore, il mezzo e i modi delle loro interazioni in termini molto concreti



In quasi tutti i calcolatori oggi in commercio, sono attuate le indicazioni date dal modello e si possono trovare tutte le componenti che prevede: CPU, I/O, memoria, BUS...

8

Il modello di Von Neumann

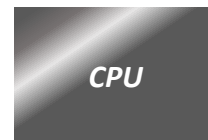


9

Il modello di Von Neumann

• Central Processing Unit (CPU)

- Esegue le istruzioni di cui è composto il programma
 - Operazioni logico aritmetiche
 - Presiede alla «movimentazione» dei dati
 - Da/per la memoria
 - Input/Output
- Coordina e controlla le varie componenti di cui è costituito il calcolatore

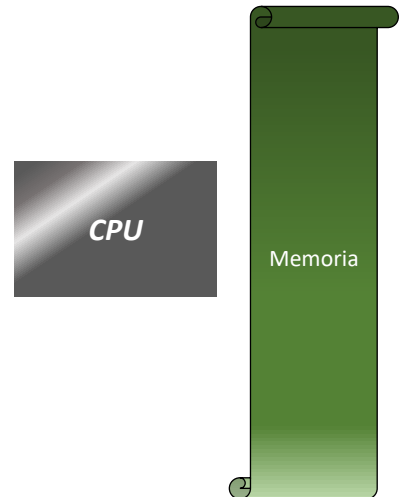


10

Il modello di Von Neumann

- **La memoria contiene:**

1. Le istruzioni che compongono il programma
2. I dati elaborati dal programma e i risultati da esso prodotti

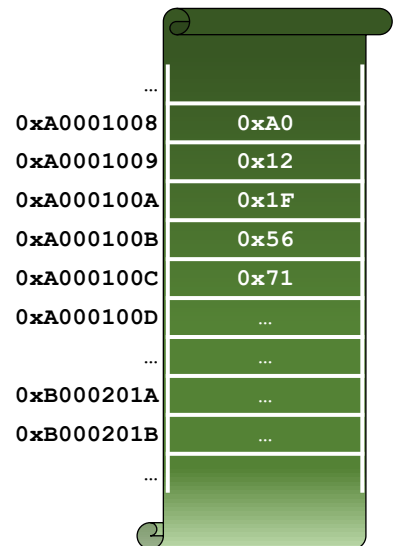


11

Il modello di Von Neumann

- **La memoria**

- è organizzata come un array di «registri» (o locazioni, o celle) in cui è possibile scrivere e leggere valori numerici
- ciascuna locazione è identificabile con un «indice» univoco: l'indirizzo
- sia le istruzioni del programma, sia i dati sono memorizzate utilizzando una codifica binaria

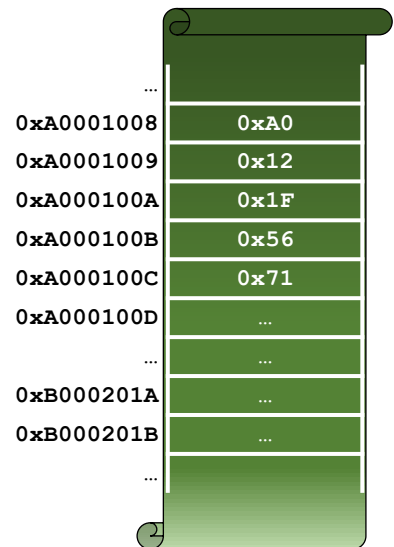


12

Il modello di Von Neumann

• La memoria

- Tutte le celle di memoria sono costituite dallo stesso numero di bit (cifre binarie).
- Il numero di bit di una cella di memoria è frutto di una scelta progettuale e non può essere cambiato.
- Se l è il numero di bit disponibili, ogni cella può contenere 2^l possibili valori distinti.

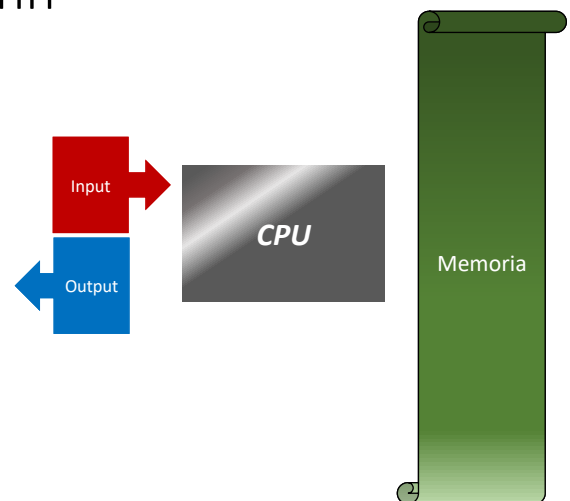


13

Il modello di Von Neumann

Dispositivi di Input/Output (I/O)

- Permettono al calcolatore di interagire con il «mondo esterno»
- **Input:** procedure e dispositivi per fornire dati al calcolatore
- **Output:** procedure e dispositivi attraverso i quali il calcolatore produce i risultati dell'elaborazione e informazioni sul suo stato

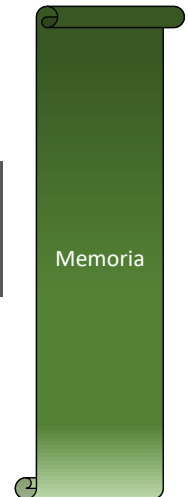
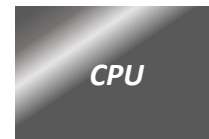
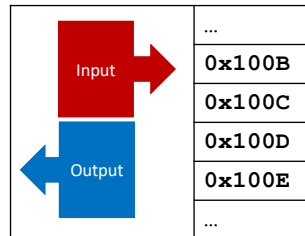


14

Il modello di Von Neumann

Dispositivi di Input/Output

- Ciascun dispositivo è identificato da un suo «indirizzo» univoco, in maniera analoga alla memoria;
- Ciascuna istruzione di I/O deve comprendere l'indirizzo di I/O del dispositivo



15

Il modello di Von Neumann



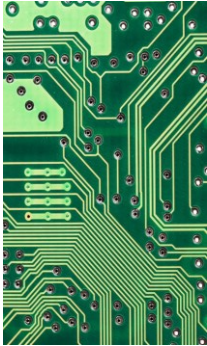
Il BUS è il canale attraverso il quale la CPU invia e riceve/dati da/verso le altre componenti.

La particolarità dei BUS è che essi sono canali *condivisi* cioè, che sono utilizzati per le comunicazioni tra tutte le componenti ad essi collegati

La metafora da cui trae origine il nome è quella degli autobus pubblici. Essi percorrono un percorso prestabilito che collega diverse località. I passeggeri che desiderano spostarsi tra qualsiasi coppia di queste, condividono lo stesso bus per il tragitto in comune.

16

Il modello di Von Neumann

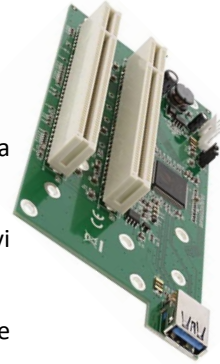


Fisicamente è composto da:

Una serie di linee (e.g. fili) che trasmettono ciascuna segnali elettrici che riproducono una codifica binaria

Una serie di interfacce (e.g. connettori) con i dispositivi ad esso collegati

Una circuiteria che regola l'accesso al bus e ne stabilisce il protocollo di comunicazione.



17

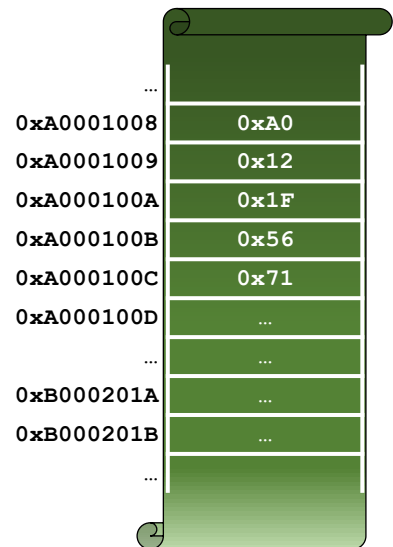
La memoria

18

La memoria

La memoria effettua due tipi di operazioni:

- **load** (lettura): Richiesta della CPU di inviarle il contenuto di una cella di memoria (dato il suo indirizzo).
 - L'operazione non ne altera il contenuto.
- **store** (scrittura): La CPU invia un dato da «stoccare» e l'indirizzo della cella di destinazione.
 - Il nuovo dato, sostituisce il precedente contenuto della cella



19

Tipi di memoria

- **RAM (Random Access Memory)** Memoria ad accesso casuale
 - Memorie in cui sono consentite entrambe le operazioni di lettura e scrittura
 - Memorie destinate a ospitare dati e programmi arbitrari.
 - il tempo di accesso non dipende dalla posizione e si può assumere (in generale) costante;
- **ROM (Read Only Memory)** Memorie a sola lettura
 - Memoria che consentono la scrittura una sola volta (generalmente con l'impiego di dispositivi ad hoc)
 - Le operazioni di lettura possono essere eseguite indefinitamente
 - Conservazione di dati e (più frequentemente) istruzioni che devono essere mantenute indefinitamente e che non devono essere modificate (Es. il BIOS del PC)

20

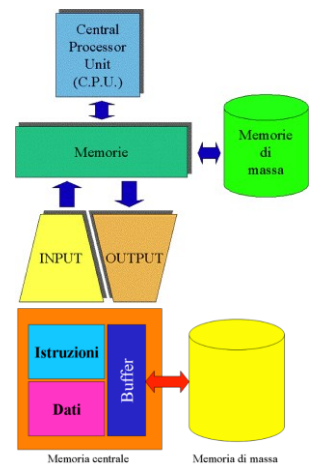
Tipi di memoria

- **Volatile:** il contenuto dei registri di memoria è perduto quando si interrompe l'alimentazione.
 - Memoria centrale di un calcolatore (RAM)
- **Non volatile:** i dati immagazzinati sono preservati anche quando i circuiti di memoria non sono alimentati
 - ROM/EPROM (= Erasable ROM)
 - NVRAM (Non Volatile RAM)

21

Memoria «di massa» (secondaria)

- Le memorie di massa sono memorie ausiliarie caratterizzate da una elevata capacità.
 - Le informazioni contenute nella memoria di massa devono essere dapprima trasferite nella memoria centrale e successivamente elaborate
 - le informazioni prodotte dalla CPU, viceversa, devono essere depositate in memoria centrale per poi essere conservate nelle memorie di massa.



22

Memoria «di massa» (secondaria)

- Le memorie di massa sono memorie ausiliarie caratterizzate da una elevata capacità.
 - Le informazioni contenute nella memoria di massa devono essere dapprima trasferite nella memoria centrale e successivamente elaborate
 - le informazioni prodotte dalla CPU, viceversa, devono essere depositate in memoria centrale per poi essere conservate nelle memorie di massa.



23

Comunicazione tra le componenti: I BUS

24

Comunicazione tra le componenti: I BUS

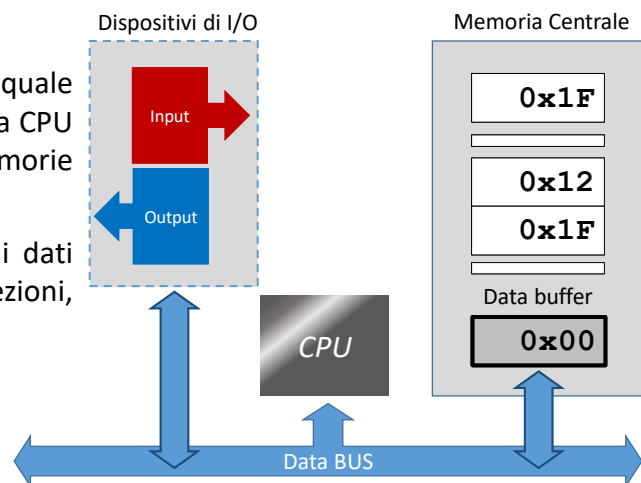
- Il BUS è un canale di comunicazione condiviso da più utilizzatori.
 - E' fisicamente costituito da uno o più «fili» su cui possono transitare uno o più bit contemporaneamente.
- A seconda delle informazioni trasportate si hanno:
 - data bus
 - address bus
 - control (o command) bus
- Un BUS può essere *monodirezionale* o *bidirezionale*;

25

Data BUS

Il **data bus** è il canale attraverso il quale sono trasferiti i dati scambiati tra la CPU e la memoria centrale, le memorie secondarie e i dispositivi di I/O.

È un canale *bidirezionale* poiché i dati possono fluire in entrambe le direzioni, a seconda dell'operazione in corso;



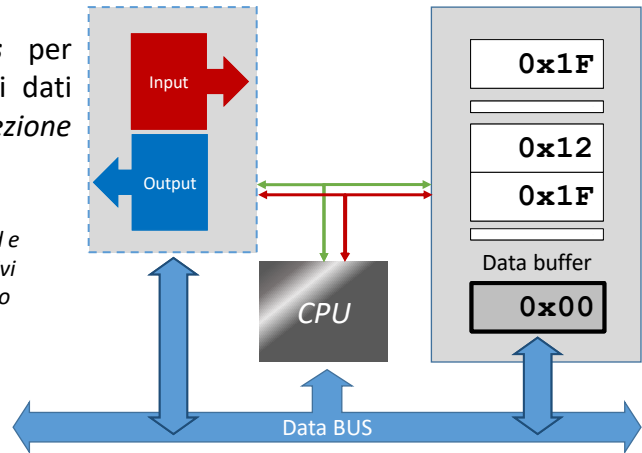
26

Control BUS

La CPU utilizza il **control bus** per regolare l'accesso e l'impiego dei dati presenti sul data bus (i.e. la direzione dei dati in transito da/per la CPU)

Tipici segnali del control bus sono quelli di read e write mediante i quali la CPU indica ai dispositivi se devono leggere un dato dal data bus (read) o scriverlo su di esso (write).

(per la memoria: load e store)



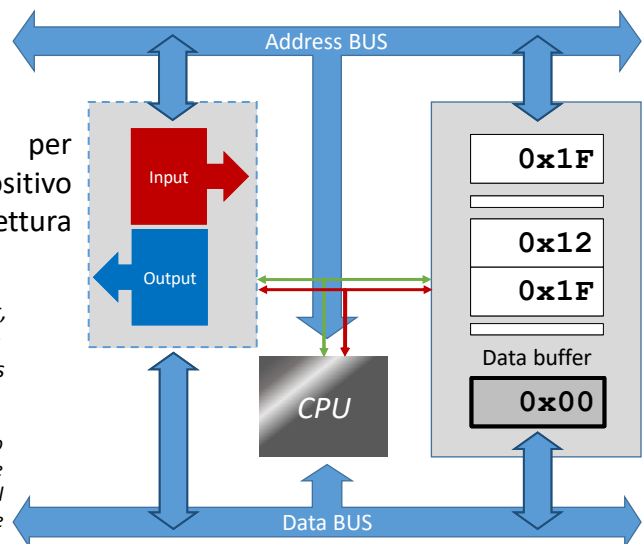
27

Address BUS

La CPU utilizza l'**address bus** per comunicare l'indirizzo del dispositivo interessato da una operazione di lettura o scrittura.

Tutti i componenti del sistema (memoria, input, output, memoria di massa, etc.) devono essere dotati della capacità di riconoscere sull'address bus il proprio indirizzo.

Originariamente anche i dispositivi di I/O erano identificati da un indirizzo analogo a quello delle celle di memoria. Tuttavia, nei calcolatori odierni, il l'address bus per i dispositivi di I/O è fisicamente separato dall'address bus per la memoria.



28

Address BUS

- La CPU è l'unico elemento che fornisce l'indirizzo all'address bus;
- Memorie e dispositivi di input ed output devono ascoltare l'address bus per attivarsi quando su di esso compare il proprio indirizzo identificativo;
 - Per la memoria l'attivazione avviene quando l'indirizzo in *transito* corrispondente ad uno dei registri di cui essa è composta;
 - le memorie prelevano dati dal data bus o immettono dati in esso in funzione del comando impartito dalla CPU;
 - Negli altri casi:
 - i dispositivi di input possono solo immettere dati sul data bus;
 - i dispositivi di output possono solo prelevare dati dal data bus.

29

Address Bus e capacità di indirizzamento

- La *capacità di indirizzamento* del calcolatore è data dal numero di bit che compongono gli indirizzi di memoria (e quindi le dimensioni del suo *address bus*).
 - Con n bit a disposizione, la CPU teoricamente può *indirizzare* fino a 2^n celle i cui indirizzi vanno da 0 a $2^n - 1$
 - Con un address bus di 16 bit, il range di indirizzi va da $0x0000$ a $0xffff$ (cioè da 0 a 65535)

30

I dati in memoria

- Per ragioni storiche e pratiche, la dimensione delle celle di memoria è di un *byte* (otto bit) mentre le dimensioni del bus e dei registri sono piuttosto variabili a seconda delle caratteristiche dell'architettura
- La dimensione del data bus indica la quantità di dati trasferibili tra memoria e CPU con una singola operazione di Load/Store
- Per esempio:
 - nel microprocessore Z80 aveva un data bus di 8 bit e registri interni da 8 o 16 bit.
 - Quindi: operandi di 16 bit potevano essere trasferiti dalla memoria ai registri interni (o vice versa) 8 bit per volta.

31

I dati in memoria

- Le dimensioni della cella non rappresentano una grave limitazione.
- Le operazioni di Load/Store su byte risultano molto veloci
- Tipi di dato caratterizzati da dimensioni maggiori *e.g.* word (16 bit), double word (32 bit) quad(*ruple*) word (64 bit) possono essere immagazzinate in celle contigue:
 - *2 celle per le word*
 - *4 celle per le double word, etc.*
 - In questo caso, possono essere fatte diverse scelte per l'ordine di memorizzazione dei byte (endianness)

32

I dati in memoria

Memorizzazione
di una word

Byte più significativo (contiene il MSB)

Byte meno significativo (contiene il LSB)

0	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

0x0110

0	1	1	0	0	1	1	1
---	---	---	---	---	---	---	---

0x0111

0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---

Big-endian:

I byte più significativi sono memorizzati all'indirizzo più basso

Usata da: processori Motorola, codifica nei protocolli di rete

33

I dati in memoria

Memorizzazione
di una word

Byte più significativo (contiene il MSB)

Byte meno significativo (contiene il LSB)

0	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

0x0110

0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---

0x0111

0	1	1	0	0	1	1	1
---	---	---	---	---	---	---	---

La endianness è una caratteristica progettuale del processore

Little-endian:

I byte meno significativi sono memorizzati all'indirizzo più basso

Usata da: processori Intel

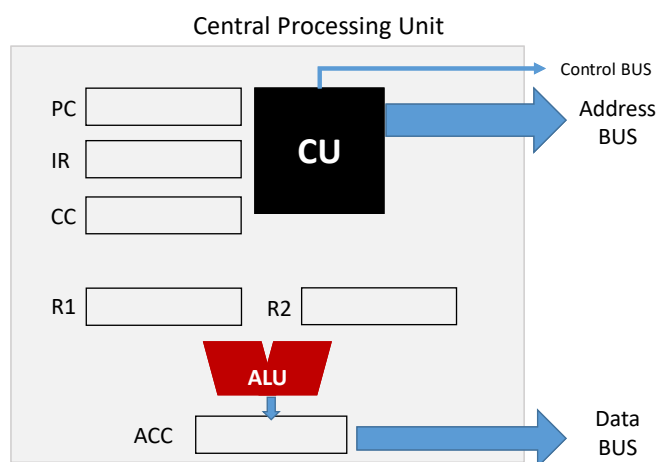
34

Central Processing Unit (CPU)

35

La CPU

- **Control Unit**
 - Decodifica ed esecuzione delle istruzioni
- **Unità Logico-Aritmetica**
- **Registri interni**
 - Operandi delle istruzioni
 - Informazioni di stato
 - Puntatore all'istruzione corrente



36

Il ciclo della CPU

Durante l'arco di tempo in cui è in funzione, la CPU esegue le istruzioni che compongono il suo programma seguendo un ciclo: «*il ciclo della CPU*» (anche detto *ciclo Fetch-Execute*) che traduce il ciclo di esecuzione degli algoritmi (il ciclo dell'esecutore)

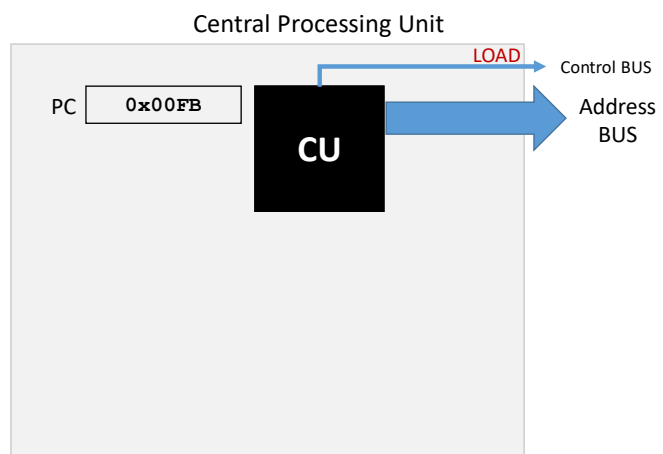
37

Il ciclo della CPU: Fetch

Control Unit (CU)

È quella che: preleva le istruzioni dalla memoria.

Tiene traccia dell'istruzione da eseguire annotandone l'indirizzo di memoria in uno dei suoi *registri interni*: il Program Counter (PC)



38

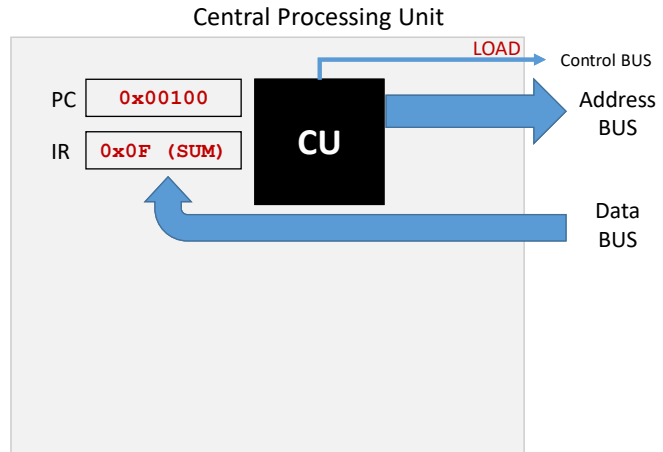
Il ciclo della CPU: Fetch

Control Unit (CU)

È quella che: preleva le istruzioni dalla memoria.

Immagazzina l'istruzione prelevata temporaneamente nel registro IR (Instruction Register)

Ogni volta che un'istruzione viene prelevata, si aggiorna il PC con l'indirizzo della istruzione seguente



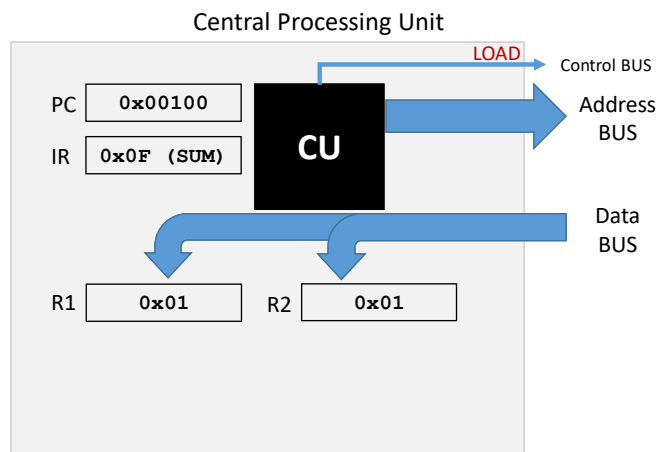
39

Il ciclo della CPU: Operand Assembly

Control Unit (CU)

Immagazzina l'istruzione prelevata temporaneamente nel registro IR (Instruction Register)

Interpreta l'istruzione e se questa prevede operandi, li preleva nella memoria e li memorizza nei *registri operando*



40

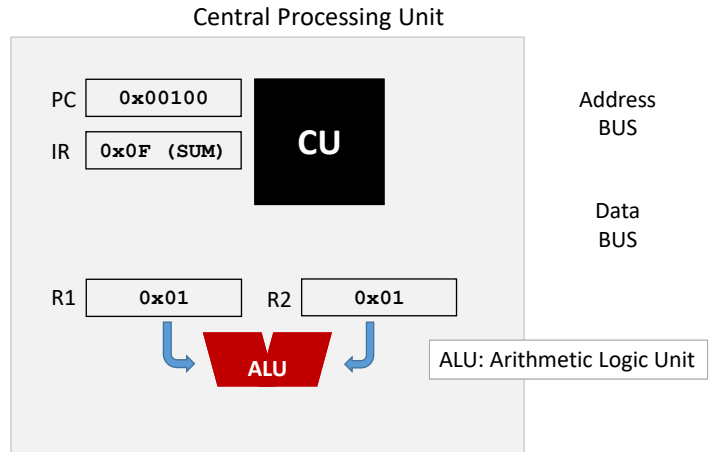
Il ciclo della CPU: Execute

Control Unit (CU)

Immagazzina l'istruzione prelevata temporaneamente nel registro IR (Instruction Register)

Interpreta l'istruzione e se questa prevede operandi, li preleva nella memoria e li memorizza nei *registri operando*

Attiva la ALU se si tratta di una operazione logico-aritmetica



41

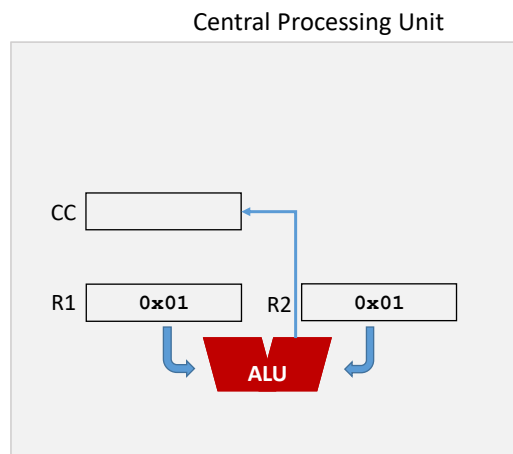
L'Unità Logico-Aritmetica

Arithmetic-Logic Unit

Su richiesta della CU, effettua le operazioni logico-aritmetiche sugli operandi caricati negli appositi registri.

Se durante l'operazione si verificano degli errori, questi sono codificati nel registro CC (Condition Code)

Errori: overflow (superamento della capacità di rappresentazione), divisione per zero, ...



42

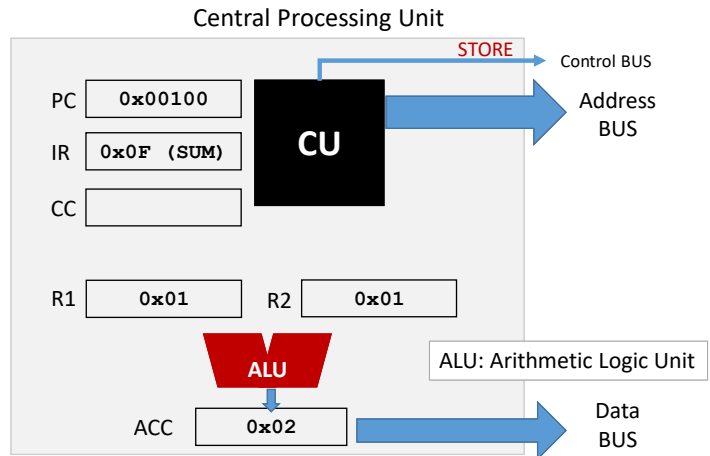
Il ciclo della CPU: Storage

Control Unit (CU)

Immagazzina l'istruzione prelevata temporaneamente nel registro IR (Instruction Register)

Attiva la ALU se si tratta di una operazione logico-aritmetica

Pone il risultato in un altro registro interno (e.g. l'Accumulatore) e da questi lo trasferisce in memoria

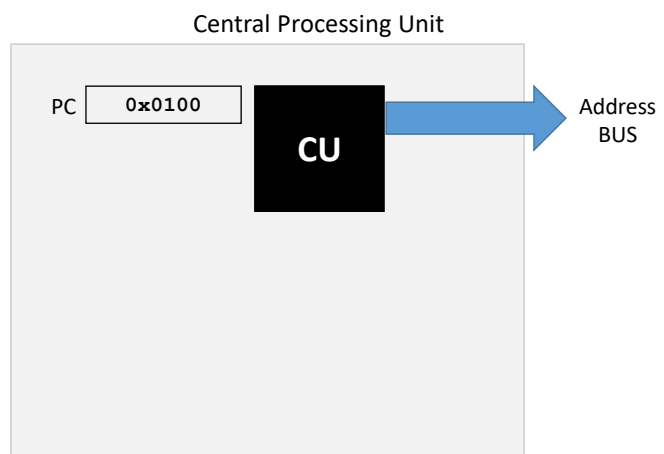


43

Il ciclo della CPU: Fetch...

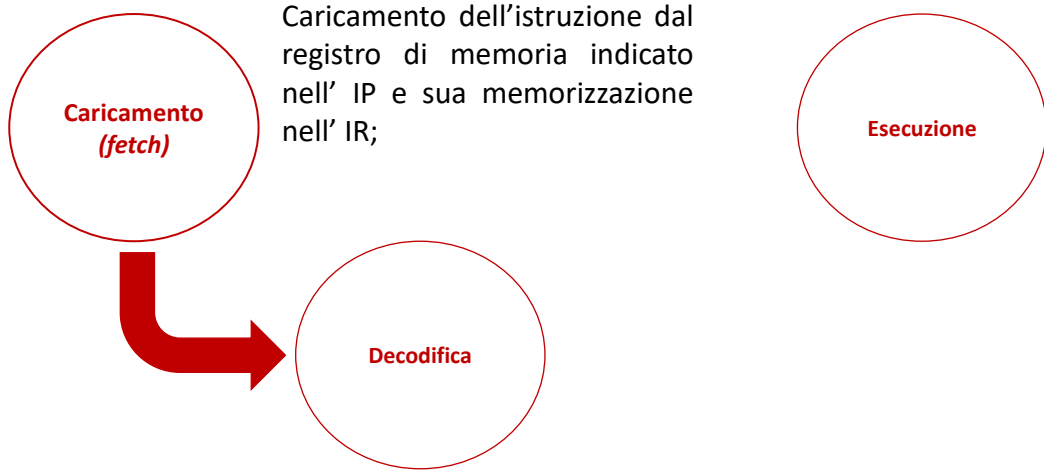
Control Unit (CU)

È quella che: preleva l'istruzione successiva e riprende il ciclo.



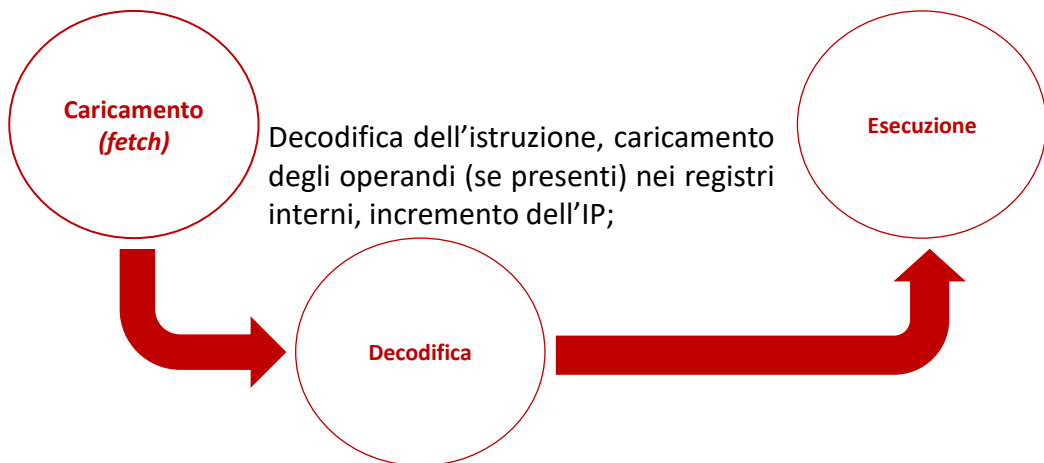
44

Il ciclo della CPU



45

Il ciclo della CPU



46

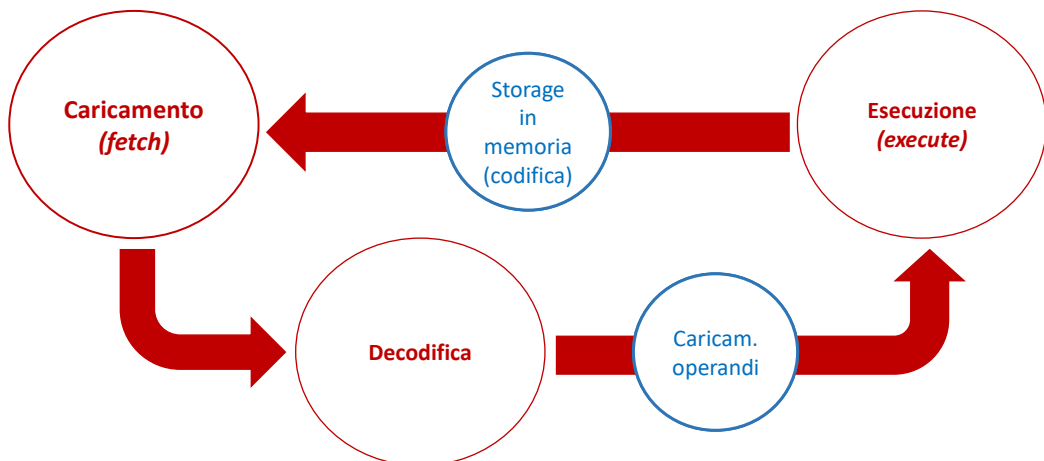
Il ciclo della CPU

Esecuzione dell'istruzione e memorizzazione dei risultati nei registri interni



47

Il ciclo della CPU



48

I programmi in memoria

49

Il «tipo di dato» istruzione

Nell'architettura di Von Neumann, le istruzioni che compongono un programma, sono immagazzinate nella memoria centrale come i dati.

Il set di istruzioni del processore e le sue regole sintattiche costituiscono quello che viene definito il «linguaggio macchina». Ciascun processore ha un suo linguaggio macchina.

Le istruzioni hanno anch'esse una codifica binaria costituita da una sequenza di byte (opcode), così organizzata:

Per esempio

Opcode	Codifica degli operandi
0F	01 01

0x00FF	...
0x0100	0F
0x0101	01
0x0102	01
0x0103	...
0x0104	...
0x0105	...

50

Il «tipo di dato» istruzione

Le istruzioni del linguaggio macchina sono molto elementari (spostamento di byte, operazioni logico aritmetiche,...)

Il linguaggio macchina risulta piuttosto astruso. Nel tempo si è affermato l'uso di sostituire i codici operativi con dei sinonimi più facili da ricordare per l'uomo (i «codici mnemonici»)

Per esempio

Opcode	Mnemonico	Codifica degli operandi
0F	SUM	01 01

0x00FF	...
0x0100	0F
0x0101	01
0x0102	01
0x0103	...
0x0104	...
0x0105	...

51

Mappa

Per lo studio e l'approfondimento degli argomenti trattati

52

Mappa

[Idb] Questo testo fornisce, un'ampia panoramica sulle tecnologie impiegate per la costruzione di un calcolatore moderno. In particolare, il Capitolo 2 introduce l'architettura della CPU (Sezione 2.8 e seguenti). Il Capitolo 3 tratta i dispositivi di Input/Output e il Capitolo 4 introduce le memorie secondarie.

[Sli] Slide della lezione