

Sistemi Operativi 1

Riccardo Cara

November 10, 2023

Contents

| | | |
|----------|---------------------------------------|----------|
| 1 | Introduzione | 2 |
| 2 | Un sistema computerizzato | 2 |
| 2.1 | Architettura di un computer | 3 |
| 2.2 | Bus di sistema | 4 |
| 2.3 | dispositivi I/O | 4 |

1 Introduzione

Un *sistema operativo* è un software che si occupa di gestire le risorse hardware e fornire software di base che cambiano in base allo scopo del dispositivo. il sistema operativo si occupa di:

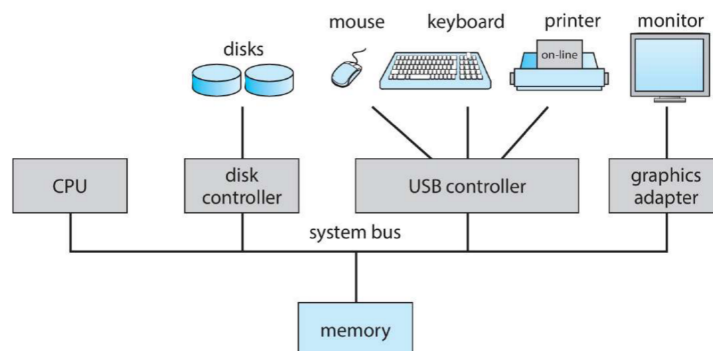
- **gestire le risorse hardware** gestire le risorse fisiche condivise per raggiungere equità e efficienza
- **virtualizzare delle risorse** per dare alle applicazioni l'illusione di avere risorse infinite, altrimenti.
- **interfacciare hardware e software** permette agli utenti di interagire con le risorse hardware senza controllarle direttamente.

e viene diviso in due parti:

- **software di sistema:** tutti i software che non sono il kernel
- **kernel:** il nucleo dell'OS e si occupa di:
 - **interfaccia hardware** interfacciare i software con l'hardware
 - **gestione della memoria** si occupa di distribuire la ram (Random Access Memory)
 - **gestione dei processi** si occupa della gestione del tempo e quindi permette il multitasking
 - **gestione dispositivi**

un software effettua delle system calls ("syscall" per chi ha studiato assembly), il kernel traduce le chiamate in serie di comandi e le invia alla cpu.

2 Un sistema computerizzato

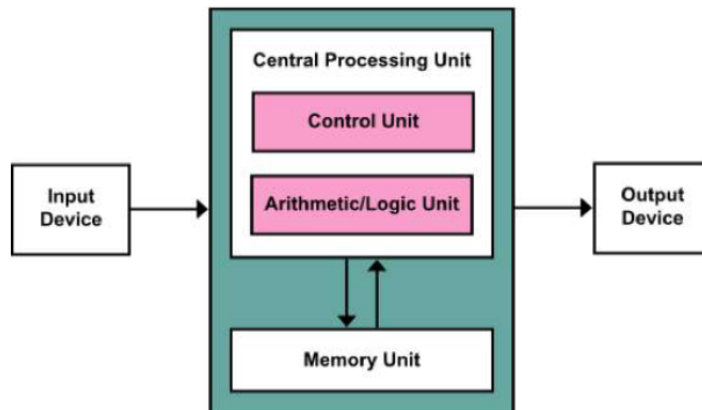


questo è lo schema di un computer, si può osservare che esso è composto da più componenti connessi tra loro tramite un bus di sistema che permette ai componenti del computer di comunicare tra loro. I componenti sono:

- *CPU* la parte che esegue le computazioni
- *memoria principale* immagazzina dati e istruzioni usati dalla CPU, è condivisa tra CPU e dispositivi I/O
- *dispositivi I/O* sono le periferiche, ovvero i dispositivi che permettono al computer di comunicare (monitor, stampante, mouse, tastiera,...)

2.1 Architettura di un computer

L'architettura di un computer è concettualmente identica tra Personal Computers, servers, smartphones ecc... basato sul concetto di programma memorizzato, ovvero un computer che immagazzina i dati e le istruzioni dei programmi sullo stesso spazio di memoria, prima di questa architettura, le istruzioni dei programmi e i dati erano salvati su spazi di memoria distinti, basti pensare a quando per cambiare programma si doveva cambiare scheda fisica. (non ne sono sicuro, forse era in un film, comunque il concetto è quello :D).



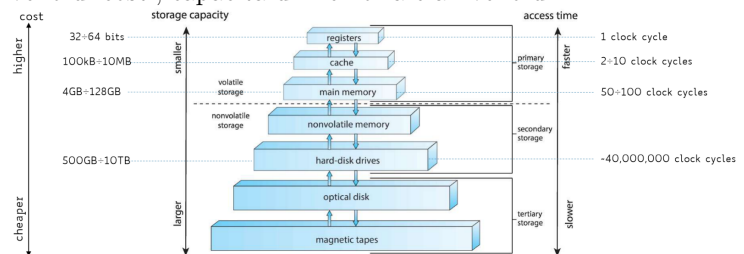
In questo tipo di architettura, la CPU esegue le operazioni in maniera sequenziale usando registri interni mentre la memoria contiene le istruzioni e i dati.

CPU esegue tre passi in maniera ciclica:

1. **Fetch**: raccoglie l'istruzione dal Program Counter (un registro di memoria speciale della cpu)
2. **Decode**: interpreta l'istruzione fetchata
3. **Execute**: esegue l'istruzione fetchata

la CPU può essere a processore singolo o a processori multipli, la differenza più grande è che la cpu a processori multipli incrementa il throughput, ovvero, nello stesso periodo di tempo, la cpu a processori multipli da in output una mole di dati maggiore rispetto al processore singolo in questo corso ci concentreremo sui sistemi a processore singolo. La CPU possiede delle istruzioni, l'insieme di istruzioni viene definito dal linguaggio macchina. Le istruzioni in linguaggio macchina sono composte da un operatore (op code), zero o più operandi rappresentanti registri di memoria o indirizzi di memoria.

Memoria in un Sistema computerizzato ci sono molte memorie, registri di memoria, cache, RAM, spazi di archiviazione(HDD, SSD) ecc... queste memorie hanno un sistema gerarchico a livello di costi, capacità di memoria e a livello di



tempo impiegato per accedervi:

le memorie sono divise in due sottogruppi:

1. **registri e cache:** sono gestiti dall'architettura
2. **memoria principale, memoria non volatile, spazi di archiviazione:** sono gestiti dal sistema operativo

2.2 Bus di sistema

i bus di sistema sono 3:

- **Data Bus:** porta l'informazione
- **Address Bus:** determina dove l'informazione va inviata
- **Control Bus:** determina quale operazione viene effettuata

Se il control bus è condiviso tra memoria e dispositivi di I/O di cui si parlerà nella prossima sezione, si utilizzerà una linea speciale chiamata "M/#io" ed indica se la CPU vuole comunicare con la memoria o con un dispositivo I/O.

2.3 dispositivi I/O

i dispositivi input output, sono divisi in 2 parti, il dispositivo in se e il device controller, ovvero un chip che permette di controllare una famiglia di device controller:

- **SATA controller:** controlla i dischi SATA

- IDE controller: controlla i dischi IDE
- usb controller: controlla i dispositivi USB
- PCI bus controller: controlla i dispositivi connessi al bus PCI
- ...

il sistema computerizzato, comunica con gli elementi I/O tramite dei software chiamati driver: SATA driver, IDE driver, USB driver, PCI bus driver ecc... I dispositivi I/O hanno dei registri dedicati con cui comunicare con il sistema computerizzato.

- registri di stato: registro in cui viene salvato lo stato del dispositivo (attende l'input, occupato, errore, transazione completata, idle, ecc...)
- registri di controllo/configurazione: usato dalla CPU per configurare e controllare il dispositivo
- registri dei dati: usato per leggere o inviare dati dal/al dispositivo I/O

La CPU può comunicare con i dispositivi I/O in due modi:

- Port-mapped I/O si riferisce ai registri dei controller tramite spazio di indirizzi I/O separato.
 - il registro di ogni device controller è mappato ad una porta (indirizzo) specifica durante il boot
 - necessita di istruzioni della CPU speciali (IN per leggere dal dispositivo I/O, OUT per scriverci)
 - non necessita di interpellare M/#IO per le istruzioni IN, OUT poiché sono solo per i dispositivi I/O e non confondibili con altre istruzioni per la memoria.
- memory mapped I/O i registri dei controller vengono mappati sugli stessi indirizzi usati per i registri di memoria.
 - non necessita di istruzioni speciali
 - le porte dei dispositivi I/O sono viste come normali indirizzi di memoria mappati nella RAM
 - per accedere ai registri dei dispositivi I/O vengono usate istruzioni simili a MOV
 - M/#IO indica sempre che gli indirizzi richiesti dalla CPU appartengono alla memoria