

Utilizarea Sistemelor de Operare



Introducere în Sisteme de Operare

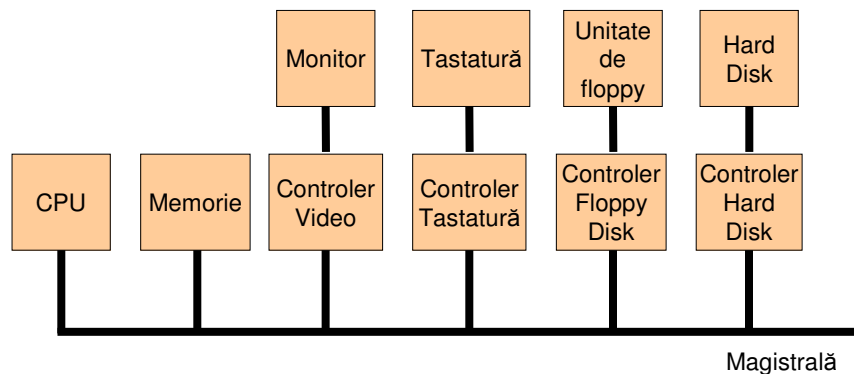
- Curs 2 -
13.10.2005

Universitatea POLITEHNICA București

1. Elemente de structură hardware



- Principial, un calculator personal simplu poate fi redus la următorul model:



1.1 Procesorul



- Procesorul are rolul de a citi instrucțiunile din memorie și de a le executa
- Ciclul de bază al CPU:
 - Citește prima instrucțiune din memorie
 - O decodifică (=> tipul ei și operanzii)
 - O execută
 - Continuă citirea, decodificarea și executarea instrucțiunilor următoare
- Fiecare CPU are un set specific de instrucțiuni pe care le poate executa

Registre



- Fiecare CPU conține câteva registre în care se rețin variabile și rezultate temporare
- Există și registre speciale:
 - Contorul de program (*program counter*)
 - Indicatorul de stivă (*stack pointer*)
 - Cuvântul de stare al programului (*PSW*)

Procesoare îmbunătățite



- Unitățile CPU moderne pot executa mai multe instrucțiuni simultan (bandă de asamblare - *pipeline*)
 - Există unități separate pentru citire, decodificare și execuție.
 - În timp ce se execută instrucțiunea n , se decodifică instrucțiunea $n+1$ și ce citește $n+2$
- O structură și mai avansată este cea a unui **CPU superscalar**. Acesta poate citi și decodifica mai multe instrucțiuni simultan, pe care le plasează într-o zonă de memorie tampon. De acolo instrucțiunile sunt preluate de mai multe unități de execuție specializate.

Moduri de lucru ale procesorului



- Un CPU poate lucra în două moduri: **modul kernel** și **modul utilizator**
 - Modul este comandat de un bit din PSW
- Pentru a obține servicii din partea SO, un program trebuie să inițieze un *apel de sistem* (*system call*).
- Comutarea între cele două moduri de lucru este făcută de instrucțiunea TRAP, care asigură și revenirea în programul utilizator la instrucțiunea ce urmează apelului de sistem

1.2 Memoria



- Cei doi parametri ce definesc o memorie sunt: timpul de acces si capacitatea.
- Sistemul de memorie este construit ca o ierarhie de niveluri:
 - Registrele procesorului
 - Memoria cache
 - Memoria principală (RAM)
 - Disc magnetic
 - Bandă magnetică

Tipuri de memorie



- **Registrele procesorului** sunt foarte rapide, dar au o capacitate $< 1\text{kB}$
- Memoria principală este împărțită în linii de **memorie cache** de 64B. Memoria cache reține liniile cele mai folosite.
- Memoria principală (**RAM**) (zeci, sute de MB)
- Discul magnetic (**hard-disk**) este un dispozitiv mecanic, timpul de acces aleator fiind cu trei ordine de mărime mai mare ca la RAM
- **Banda magnetică** este utilizată pentru arhivare
- Alte tipuri de memorie
 - Nevolatilă (ROM, EEPROM, flash)
 - Volatila (CMOS)

Problemele partajării memoriei

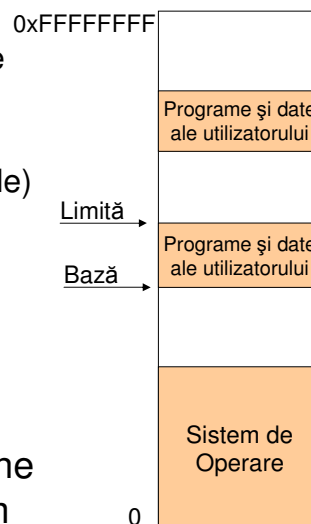


- Memoria principală poate fi partajată între mai multe programe.
- Apar însă două probleme:
 - Protejarea unui program de altul și a kernelului de toate
 - Gestiunea realocării de memorie

Partajarea memoriei



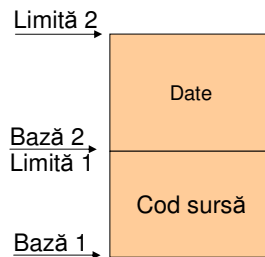
- Se folosesc două registre:
 - **Registru de bază** (reține adresă de început a textului programului)
 - **Registru limită** (precizează cât de mari sunt textul programului și datele)
- Adresa generată de program se numește **adresă virtuală** și este mai mică decât valoarea din registrul limită
- Adresa utilizată de memorie se numește **adresă fizică** și se obține prin adunarea contorului program la registrul de bază



MMU



- Punerea în corespondență a celor două tipuri de adrese este făcută de MMU (Memory Management Unit)
- Un MMU mai avansat folosește două perechi de registre baza-limită, una pentru a delimita codul programului și alta pentru a delimita datele



1.3 Dispozitivele de I/E



- Principial sunt alcătuite din două blocuri:
 - Controler (comandă nivelul fizic al dispozitivului)
 - Dispozitivul propriu-zis (cu interfață standard pentru asigurarea compatibilității cu controlerul)
- Deoarece există mai multe tipuri de controlere, sunt necesare programe diferite (drivere) care să facă legătura între controler și SO.

Adăugarea unui driver



- Pentru a putea fi folosit, un driver trebuie adăugat SO astfel încât să poată rula în mod kernel.
- Acest lucru se poate realiza în trei moduri:
 - Se recompilază kernel-ul cu noul driver și se restartează sistemul (specifică UNIX-ului)
 - Se realizează o intrare într-un fișier al SO specificându-i că are nevoie de driver și se restartează sistemul, acesta încărcând driverul la repornire (specifică Windows-ului)
 - Driverul se poate instala “din mers”, fără ca sistemul să fie reinițializat (de ex: dispozitivele ce folosesc interfața USB)

Operații I/E



- Operațiile de intrare-ieșire se pot realiza în trei moduri:
 - Busy waiting:
 - Apelul de sistem → apel de procedură pentru driver
 - Driverul pornește dispozitivul și rămâne într-o buclă de interogare până când se realizează operațiile cerute
 - Driverul transferă datele și se întoarce în program
 - Folosind întreruperi
 - Driverul pornește dispozitivul și solicită o întrerupere la terminarea activității.
 - SO îl blochează pe apelant și execută alte operații.
 - La sfârșitul activității controlerul generează întreruperea și SO revine în programul apelant
 - Se utilizează Direct Memory Access:
 - CPU precizează numărul de octeți de transferat, dispozitivul și adresa implicate și sensul
 - Transferul propriu-zis este controlat de către chipul de DMA, fără intervenția CPU
 - La finalizarea transferului chipul de DMA generează o întrerupere

Gestionarea întreruperilor

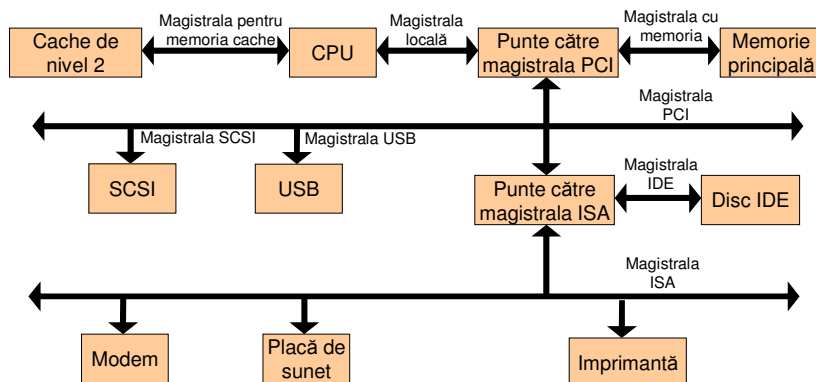


- CPU poate dezactiva întreruperile pentru un timp. Dacă mai multe dispozitive generează întreruperi în acest timp, controlerul de întreruperi le va trata în funcție de o ierarhie statică de priorități

1.4 Magistrale



- Pe varianta originală de IBM PC exista o singură magistrală. Cu timpul aceasta a devenit insuficientă și au fost adăugate magistrale suplimentare:



Tipuri de magistrale



- Acest sistem are 8 magistrale:
 - Magistrala pentru memoria cache
 - Magistrala locală
 - Magistrala pentru memorie
 - Magistrala PCI (Peripheral Component Interconnect)
 - Magistrala SCSI (Small Computer System Interface)
 - Magistrala USB (Universal Serial Bus)
 - Magistrala IDE (Integrated Drive Electronics)
 - Magistrala ISA (Industry Standard Architecture)
- Cele mai importante sunt magistralele ISA și PCI
- Conceptul de Plug and Play definește un PC ce nu necesită instalare (managementul nivelurilor de întrerupere și al adreselor de I/E este făcut de sistem)