

(Personal computer - From Wikipedia, the free encyclopedia)

A személyi számítógépet ára, mérete és képességei – és a használatában kialakult kultúra – teszik elérhetővé és kívánatosszá az otthoni és hivatali alkalmazásban. Általában egyszerre egy felhasználó használja őket a szokásos feladatokra: szövegszerkesztés, internetezés, e-mail-ezés, táblázat szerkesztés, ezen kívül számítógépes játék, film nézés, zenehallgatás, és sokkal ritkábban számítógép programozásra.

A PC a munkavégzés minden fázisában jelen van, abba teljesen integrálódik, alkalmazása nélkül az élet megállna.

A modern személyi számítógép használója többé-kevésbé tisztában van az operációs rendszer és az alkalmazási programok működésével, de általában nem érdeklődik, sőt nem is képes számítógép programot írni. Ezért a PC-re írt programok többségénél a nagyon egyszerű használatra, a “felhasználó barát” működésre törekednek.

A PC története

Előtörténet:

Mini számítógépek:

például [LINC](#) és [PDP-8](#). Ezek viszonylag nagy (kb. jégszekrény méretű), de az akkori idők nagy számítógépeihez képest kicsi, kisteljesítményű és viszonylag olcsó gépek voltak.

Otthoni számítógépek:

ZX Spectrum

Atari

Commodore, Amiga

Apple II

Először TV-hez lehetett csatlakoztani őket, a jobb modellekhez monitort is fejlesztettek.

Programozás gyakorlására,

Játékgépként,

táblázatkezelőként, később egyszerű szövegszerkesztőként alkalmazták őket.



PC modellek

1981.08 - IBM PC	Intel 8088, 5,5' floppy disk, MDA (80 x 25 karakter) vagy CGA monitor (640 x 200 pont, 4 szín), 64..256kbyte RAM
1983.03 – IBM XT	Intel 8088, 5,5' floppy disk, 10MByte Hard disk, 128..640kByte RAM
1984 – IBM AT	Intel 80286, CMOS RAM, max. 16MByte RAM, 5,5' floppy disk, 20MByte hard disk, EGA monitor
1987 – IBM PS/2	Intel 80386, 3,5' floppy disk, VGA monitor

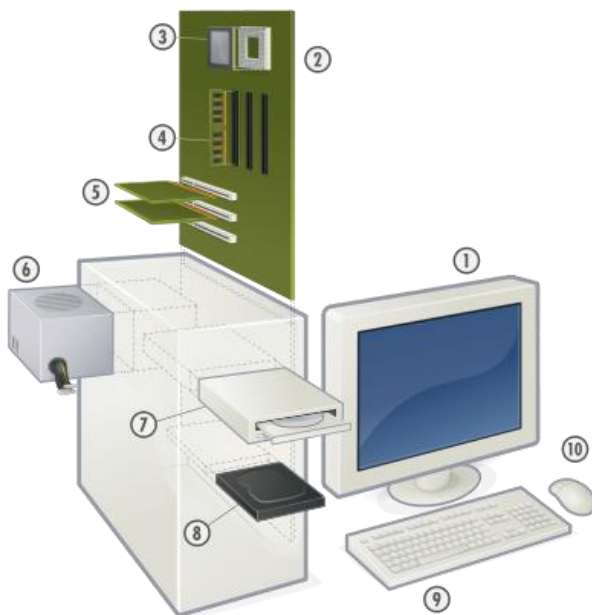
Processzorok

1982 - IBM 80286	valós és védett működési mód, 16 Mbyte RAM lehetősége
1986 - IBM 80386	32 bites, valós, védett és virtuális működési mód, flat memória modell, elvileg maximum 4GByte memória
1989 - IBM 80486	integrált lebegőpontos egység, utasítás és adat cache
2003 – Pentium	64 bites, superscalar architektúra: két adatvonal, így órajelenként több utasítást hajthat végre.

...

[Intel Core](#) 2-i3-i5-i7-i9, [Atom](#), [Celeron](#), [Pentium](#), [Xeon](#), [Itanium](#)

Egy modern PC robbantott képe



1. Display	5. Bővítőkártyák	9. Billentyűzet
2. Alaplap	6. Tápegység	10. Egér
3. CPU (Mikroprocesszor)	7. Optikai disk (CD vagy DVD)	
4. Operatív tároló (RAM)	8. Háttér tároló (Hard disk)	

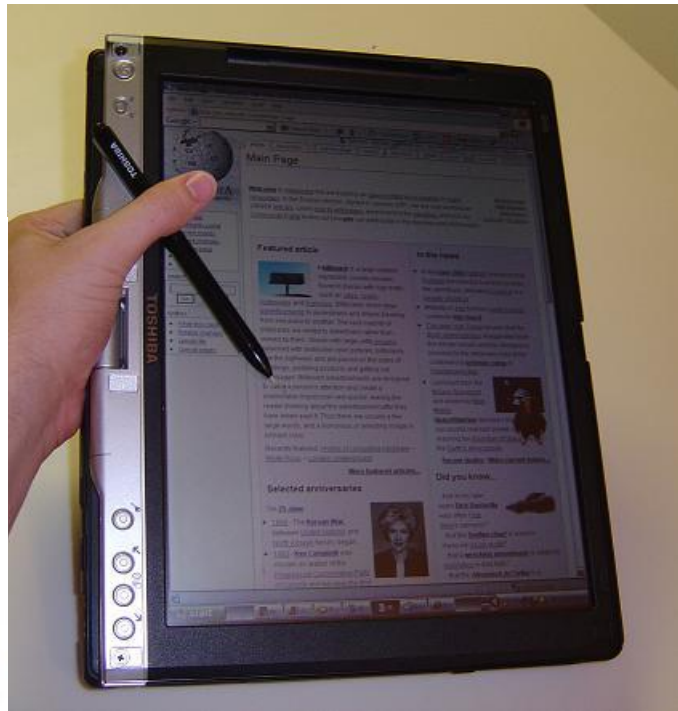
A PC-k megjelenési formái

Asztali gép

Notebook



Palmtop vagy PDA
(Personal Digital Assistant)

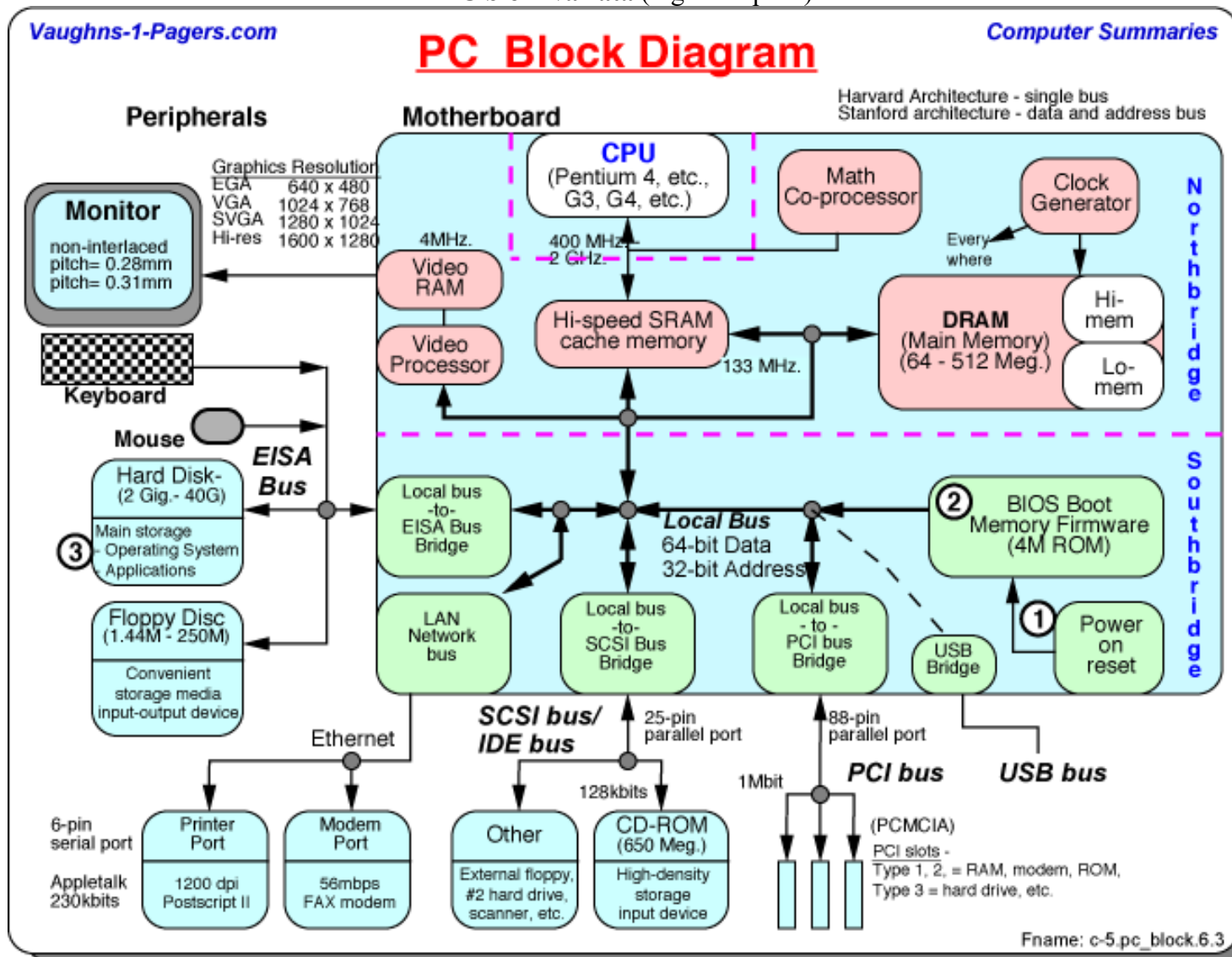


Tablet gép



Viselhető gép

PC blokkvázlata (régebbi típusú)

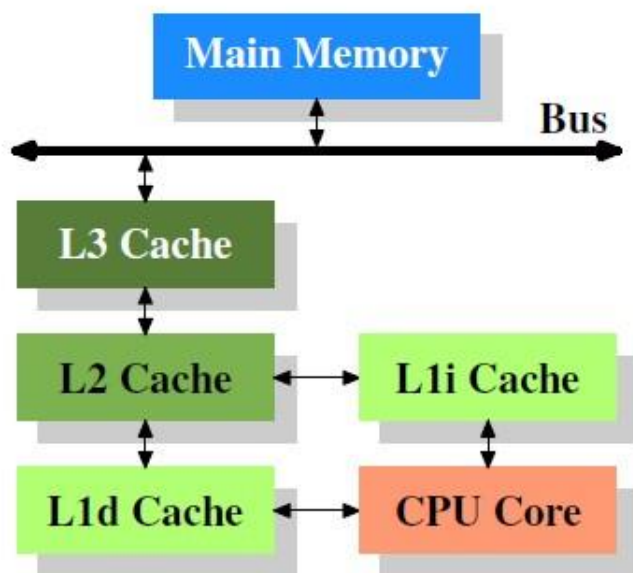


Számítógépek teljesítményének növelésére használt eszközök

Csak néhány érdekes megoldást ismertetünk, messze a teljesség igényétől.

A PC-ben a kódmemóriát is RAM valósítja meg. A nem felejtő memória kisméretű ROM. Ebben csak az operációsrendszer elindításához szükséges program és periféria kezelő programrészek vannak (BIOS, Basic Input Output System). *A ROM lassú a RAM-hoz képest, az elején a ROM tartalmát a ROM-ban levő program a RAM-ba másolja (shadow RAM), majd onnan fut a továbbiakban.*

Habár a PC processzora (CPU) Harvard architektúrájú (külön buszokon fér hozzá a kód memóriához és az adatmemóriához), azonban a kód és az adatok fizikailag egyetlen RAM memóriában, az operatív memóriában helyezkednek el. *Az operatív memória lassabb, mint a processzor, ezért az operatív memória és a CPU közé több gyors memóriát helyeznek el. Ezeket nevezik cache-nek.* Az első kettő a Level1 program cache és Level1 adat cache. Ezek újabb Level2 cache-hez csatlakoznak, ez pedig (esetleg egy Level3 cache-hez, amely pedig) az operatív memóriához. A továbbiakban egyetlen cache működési elvét mutatjuk be.



Cache

(Angolul ugyanúgy mondják, mint a Cash-t [kassza, készpénz], de jelentése rejtékhely.)

A cache egy a főmemóriához képest **kisméretű, nagyon gyors**, ún. **asszociatív (tartalom címzésű) memória**.

*Amikor a CPU egy memória címről először olvas, akkor az olvasott **memóriacím**, az olvasott **adat** és egy minden Cache műveletnél eggyel növekvő **sorszám** a Cache-ba is bekerül.*

*Ha legközelebb a CPU **ugyanerről a címről olvas**, akkor az adatot nem a memóriából, hanem a **Cache-ból** kapja meg. A Cache sokkal gyorsabb, mint az operatív memória, ezért a működés felgyorsul. A módszer azért eredményes, mert a számítógép programok gyakran hajtják végre ismételten ugyanazt a programrészt (program ciklus), illetve veszik elő többször ugyanazt az adatot.*

A Cache működéséhez a következő feladatokat kell megoldani:

- rá kell jönni, hogy egy memóriacím tartalma benne van-e a Cache-ben. Ezt úgynevezett tartalom szerinti kiválasztással valósítják meg: a Cache azt a rekeszt választja ki, melyben a cím megegyezik a kívánt címmel. Az így működő memóriát asszociatív memóriának nevezik.
- *ha a Cache tele van, akkor egy új adat behelyezésekor egy bentlévőt felül kell írni.* Általában *a legrégebben használtat írják felül*. Ezt segíti a használati sorszám: *a legkisebb sorszámút kell felülírni*. (az ábrán a sorszámok azt mutatják, hogy a Cache 1., 3., 4., 5 sorszámú helyére került tartalom. Most 2. es sorszám azért nincs, mert legutoljára ugyanarra a 201h címre írt a CPU, így a sorszámát és az adatot felülírta. Ha most a CPU új memória címre ír tartalmat, a Cache-be az 1. es sorszámú helyet fogja felülírni és a használati sorszáma 6 lesz.)

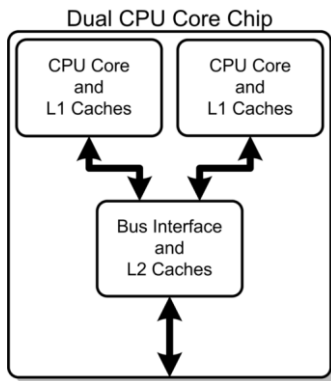
CPU	Cache			Operatív memória
	Használati sorszám	Memória cím	Adat	
	1	120h	A7h	
	5	201h	23h	
	3	070h	15h	
	4	121h	7Ch	

- Ha a CPU a memóriába ír, akkor ezt egyrészt a Cache-be, másrészt a memóriába is be kell írni, de a CPU-nak nem kell megvárnia a memóriába történő írás végét.

Ha az adat benne van a Cache-ben, azt Hit-nek (találat), ha nincs, Miss-nek (céltévesztés) nevezik.

A PC-ben használt processzorokban nagyméretű Cache-ek vannak, az elmondottnál sokkal bonyolultabb szervezésűek.

Többmagos CPU

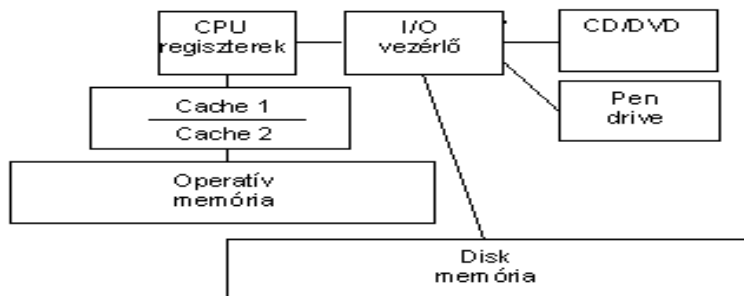


2005. májusban jelent meg a kétmagos AMD Opteron 875, 2005. novemberben a kétmagos Intel Xeon MP.

A kétmagos processzorok teljesítménye elvileg közel dupla akkora lehet, mint az egymagosé, de ennek feltétele az, hogy a gépben ténylegesen két szál fusson párhuzamosan.

(Szálaknak nevezik egy programon belül, vagy több programban a párhuzamosan futó feladatokat, például egy játékprogram és vele párhuzamosan egy internet letöltés fut.)

Memória hierarchia



Típus	Kapacitás	Hozzáférési sebesség
Regiszter	10-100byte	CPU órajel
Cache L1	128kbyte...	Néhány CPU ciklus
Cache L2	1Mbyte...	$L2 < L1$
Operatív memória	2-32...Gbyte	$\dots < 10 \text{ GBit/s}$
Disk memória (SSD)	128G...8TByte	$\dots - 600 \text{ MBit/s}$
Disk memória (HDD)	512G...Tbyte	$\dots - 50 \text{ MBit/s} \dots$
Pen drive	1Gbyte...256Gbyte	Írás: $\dots - 200 \text{ Mbyte/s}$ Olvasás: $\dots - 400 \text{ Mbyte/s}$
CD/DVD	650Mbyte...8Gbyte	10...168 MBit/s

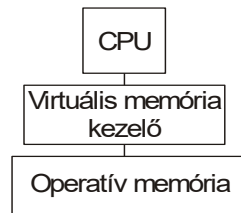
Virtuális memória az operatív memóriában

Több (sok) program párhuzamos futtatásakor alkalmazzák, amikor az összes futtatandó program a memóriában van.

Párhuzamos futtatás: *egyetlen CPU váltogatva hajtja végre a programokat* (egyszerre csak egyet képes végrehajtani!), úgy hogy a váltogatást a felhasználó ne vegye észre, vagy ne kelljen kényelmetlenül sokat várnia (néha kell).

Ha egy programot leállítanak, azt ki kell venni a memóriából, hogy hely legyen a később betöltendő programnak. A probléma az, hogy *a betöltendő program kisebb, vagy nagyobb, mint a kivett, így üres hely marad, vagy az új program nem fér el. Ráadásul a processzor-nak egy programot egymást követő szegmenseken (folyamatos címtartományban) kell látnia.*

A megoldás:



A processzor és a memória közé egy úgynevezett Virtuális memória kezelő modul kerül.

A memóriát szegmensekre (kis, egyforma méretű területekre) osztják és a programot szegmensenként töltik be. Nem biztos, hogy egy program szegmensei a memóriában folytonosan helyezkednek el. Például:

Fizikai elrendezés:

Szegmens	0	1	2	3	4	5	6	7	8	9
P1, P2,P3 betöltése	P1	P1	P1	P2	P2	P3	P3	P2	P2	üres
P2 leállítása, törlése	P1	P1	P1	üres	üres	P3	P3	üres	üres	üres
P4 betöltése	P1	P1	P1	P4	P4	P3	P3	P4	P4	P4

(Az utolsó sor úgy keletkezett, hogy először egymás után betöltődött a P1, P2, P3 program, majd P2-t leállították, és annak helyére töltődött volna a nagyobb P4 program, melynek csak egy része fért be, a maradék a végére került). **A Virtuális memória kezelő a program nem egymás melletti fizikai címen levő szegmenseit egymás melletti címen látszó logikai szegmensekre transzformálja.**

Logikai elrendezés (amit a processzor lát, P2 leállítása előtt):

Program	P1			P2				P3	
Logikai szegmens	0	1	2	0	1	2	3	0	1
Fizikai szegmens	0	1	2	3	4	7	8	5	6

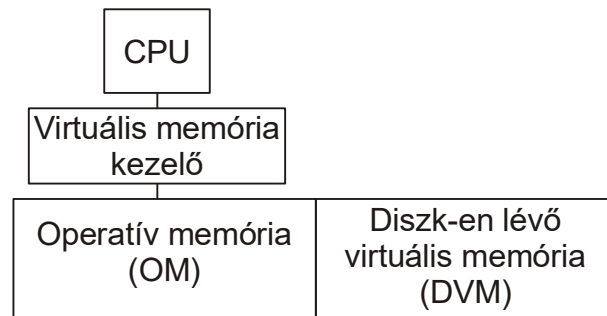
A logikai szegmens (futó program címe) $\longleftrightarrow$ fizikai szegmens (memória cím) összerendelés a virtuális memória kezelő végzi. Ha a egyik programról a másikra váltunk,

megváltozik az összerendelés. (Megjegyezzük, hogy a programok betöltését, és program váltáskor az összerendelést az Operációs rendszer végzi.)

Virtuális memória az operatív memóriában és diszken

Tételezzük fel, hogy több programot akarunk párhuzamosan futtatni, mint amennyi elfér az operatív memóriában. Ekkor **a programok egy részét ki kell tenni a háttértárolóra, amikor éppen nem működnek.**

Ezt automatizálja a PC-ben alkalmazott virtuális memória kezelés:



A Diszk memória terület egy részét, mint Disk Virtuális Memóriát (DVM) alkalmazzuk, és a virtuális memória kezelő nem csak az operatív memóriát (OM), hanem ezt is kezeli:

$$\text{Virtuális memória} = \text{OM(RAM)} + \text{DVM(háttértárban)}$$

A diszken (háttértárban) lévő virtuális memória a címzés szempontjából az operatív memória folytatása, ugyanúgy szegmentálva van..

A program, vagy annak egyes szegmensei vagy az operatív memóriában, vagy a DVM-ben vannak. **A CPU csak az operatív memóriában lévő programot tudja futtatni. Ha olyan pontra érünk, ahol a DVM-ben lévő programot kellene futtatni – ez előfordulhat egy program indításakor, vagy futás közben – a virtuális memória kezelő megállítja a program futását, és az operációs rendszer átrendezi a programokat:**

- **a DVM-ben lévő program szegmenst be kell tölteni.**
- **ha nincs az OM-ben hely, akkor előzőleg helyet kell csinálni, azaz az OM-ből annyi szegmenst kell a DVM-be kiírni, amennyi helyre szükség van.**

Nem kell a DVM-be menteni azt a szegmenst, amely már megvan a DVM-ben és az utolsó DVM-be mentés óta nem módosult. Ennek támogatására a virtuális memória kezelő megjegyzi, hogy egy szegmensbe annak betöltése óta írtak-e.

Az OM – DVM cserét swap-pelésnek (csere-bere) nevezik. A swappelés időigényes művelet, mert a diszk nagyságrendekkel lassúbb, mint az operatív memória. Ha túl sok a swappelés, akkor a rendszer megengedhetetlenül lelassulhat. Ezért a DVM maximális méretét az operatív memória méretének 1..4-szeresére szokták beállítani (a Windows eredetileg 3x-ra állítja), így az egyszerre a virtuális memóriában lévő programok összhossza az OM kapacitásánál 2..5-ször nagyobb lehet. Ennél több program esetén megengedhetetlenül sok lenne a swappelés.

Merev lemez

Mai merev lemezes tárolók adatai:

Kapacitás: 128Gbyte-...Tbyte

Méret: 3,5' vagy 2,5'

Sebesség:

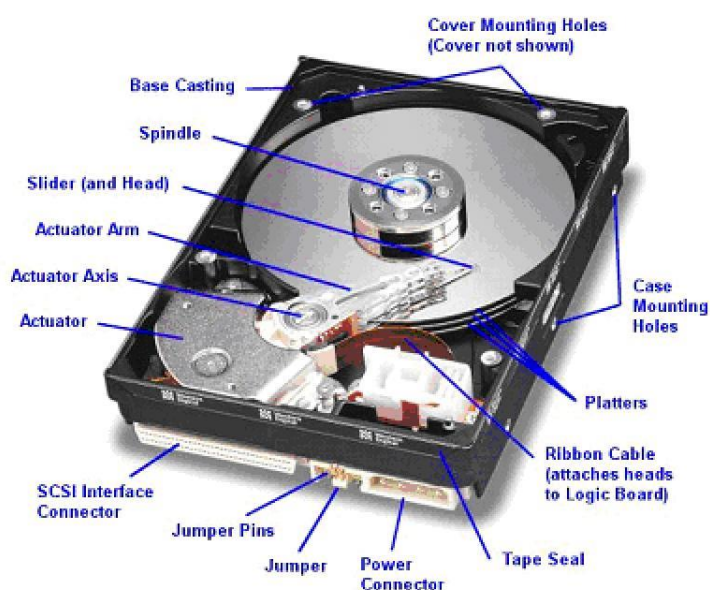
Véletlen hozzáférési idő: 5ms ... 15ms

Adatátviteli sebesség : 50...120MByte/s

Western Digital WD400 Hard disk felül és alulnézetben:

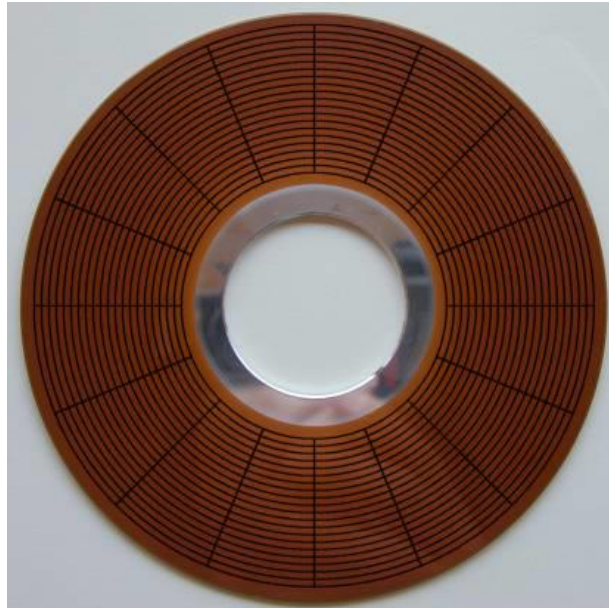


Hard disk felépítése:



A Hard disk a kazettás magnóval azonos elven rögzíti az információt, de a mágneses anyag kétoldalasan borított tárcsákon van, ezek előtt mozog az író-olvasó fej.

Sávok és szektorok



Az ábra szemléletesen mutatja a sávok és szektorok elhelyezkedését. A valóságban a nyom sűrűség sokkal-sokkal nagyobb! Az írás-olvasás logikai egysége a szektor, ami például 1024 byte-nyi információ.

Slolid state disk

Forgómágneses tárolás helyett flash-ROM-ot alkalmazó nagykapacitású tároló.

Alkalmazás: Notebook-ban

Előnyök a hard drive-hoz képest:

kisebb fogyasztás

rezgésállóság (max. 10 G; hard disk < 1G)

gyorsabb működés

nagyobb élettartam?...

Hátrány: még drágább, de csökken az ára...



Monitor és grafikus kártya

A **monitor** TFT (Thin film transistor) képernyős megjelenítő eszköz. A PC által használt felbontás:

- 640 x 480 (VGA)
- 800 x 600 (SVGA)
- 1024 x 768 (XGA)
- 1280 x 1024
- vagy még nagyobb [valamikor 320 x 200 volt].

TFT monitor

A TFT képernyőn pontosan a felbontásnak megfelelő **sor x oszlop** számú (17'-os vagy 19'-os 1280 x 1024) RGB vezérelhető színszűrőket vezérlő TFT tranzisztor hármass van, amely az oldaltól (a képernyő alja felől) jövő megvilágítást háromféle színben a vezérléstől függő erősséggel a felület felé engedi. A PC grafikus kártyája ezeket a tranzisztorokat vezérli, ezért a TFT monitor ábrája borotvaéles.

Figyelem! TFT monitor alkalmazásakor a PC kép felbontását pontosan a monitor felbontására kell állítani!!!

A TFT monitoroknak 3 fő paramétere van:

- **látószög**, a TFT kép viszonylag széles látószögben színtorzulás nélkül látszik.
- **fehér-fekete arány**. Ez kb. 400:1, rosszabb, mint CRT monitornál, de gyakorlatilag megfelel (sötét szobában a „fekete” monitorkép halványan világít).
- **utánvilágítás**. Ez gyors mozgásoknál okozhat problémát, például az egérmozgatás „farkot” húz maga után. A korábbi 20-25ms értéket sikerült, 6-10ms-ra csökkenteni, a probléma nem lényeges.

A processzort és a monitort a **grafikus kártya** kapcsolja össze. A grafikus kártya alapfeladata a például 1280 x 1024 képpontból álló kép megfelelő frekvenciájú kiküldése a monitornak. Alapesetben ezt a képet a PC hozza létre, a grafikus kártya a rasterképet tárolja, és ismételten (periodikusan, a képfrissítési frekvenciával) kiküldi.

A színek száma: Az eredeti VGA-nál 16 szín volt, az eredeti SVGA-nál 256 szín, a mai grafikus kártyáknál RGB komponensenként 8, 10, vagy 12 bit. Ez azt jelenti, **hogyan egy képpont szín és fényesség információját a kártya 3 x 8, 3 x 10 vagy 3 x 12 biten tárolja, és ennek megfelelő számú színárnyalatot tud kiküldeni a monitornak**. Ez például 24 bitnél $2^{24} = 16777216$ -féle színt jelent.

A 2D, illetve 3D grafikus kártyák grafikus objektumokat tudnak tárolni, és azokon síkbeli, vagy térbeli transzformációkat – eltolás, nagyítás, forgatás – tudnak végezni. A kártyán – a Pentium CPU-val összevethető – nagy teljesítményű speciális processzor és nagy memória van, ára nagyon magas lehet. A 3D működésnek főleg – vagy csak – videójátékokban van szerepe az alakzatok mozgásakor.

USB busz

Újabban kifejlesztett nagysebességű Univerzális Soros Busz. Csatlakozója:



Működés: 1 gazda (host), sok eszköz

4 vezeték:

Pin	Host funkció	Eszköz funkció
1	V _{BUS} (4.75–5.25 V)	V _{BUS} (4.4–5.25 V)
2	D [−]	D [−]
3	D ⁺	D ⁺
4	Föld (GND)	Föld (GND)

A Host tápfeszültséget adhat az eszköznek. Feszültség alatt lehet csatlakoztatni, bontani.

- Eszközök:
- Billentyűzet
- Egér
- Nyomtató
- Pen-drive
- Fényképezőgép, stb.

3 adatátvitel sebesség tartomány:

- Kis sebesség (low): 183 Kbyte/s humán interface eszközökhöz (egér, billentyűzet)
- Teljes sebesség (full): 1.4 Mbyte/s. USB 1.x-ben ez a legnagyobb
- Nagy sebesség (high): **57 MByte/s**. Csak **USB 2.0**-ban használható, de egy USB 2.0 szerint működő eszköz is lehet lassú.

Nagyon feltűnő a teljes (full) sebesség és a nagy (high) sebesség különbsége. A Nagy sebességhez USB 2.0 PC port és USB 2.0 nagy sebességű eszköz kell.

- Szuper sebesség (super): **626 Mbyte/s** csak **USB 3.0**-ban használható. Az USB 3-al több mint 10x-es sebesség érhető el az USB 2-höz képest!
- **USB 3.2 2x2 20 Gb/s**
- **2019-ben** jelent meg az **USB 4** szabvány, az elérhető sebesség: **40 Gb/s**