

Számítógépek Architektúrája

*segédlet a GKxB_INTM110,
az xGB_IN004_1, az xGB_IN061_1,
valamint GKxB_INTM012 kurzusokhoz*

összeállította: Paál Dávid
rs1.sze.hu/~paalda

*** Folyamatosan bővülő anyag * Célszerű ellenőrizni a verziót és a dátumot ***

Az óravázlat nem pótolja az előadás meghallgatását és megértését!

Javasolt ezen kívül – egyéni úton – további,

a témához kapcsolódó megbízható források megismerése, tanulmányozása!

Első fejezet (ALAPOK)

1. Analóg / Digitális fogalma

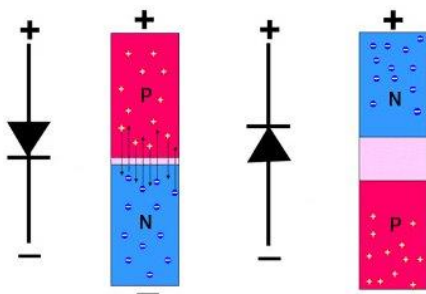
Két tetszőleges pont között végtelen vagy véges számú lépés van.

2. Analóg „számítógép” (célgép) / Digitális számítógép (univerzális)

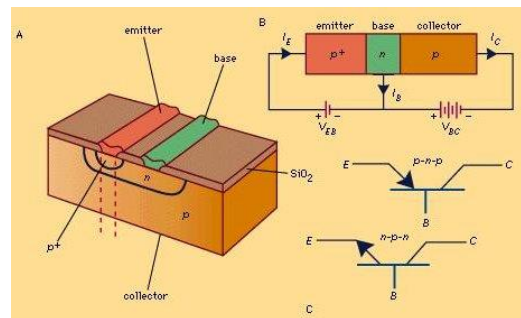
Newton majd Kelvin: apály/dagály előrejelzés, természeti fizikai, azaz analóg jellemzők a bemeneti információk, és ezek matematikai feldolgozása, fix „program/algorithmus” szerint.

3. Feszültség, áram, teljesítmény, (hővesztés → hűtés)

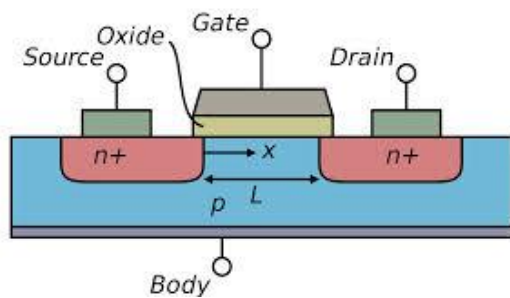
Tranzisztor, FET, MOSFET; TTL [0-0.8V, 2.0-5V], CMOS [1/3, 2/3, 3/3]; tápfeszültségek;



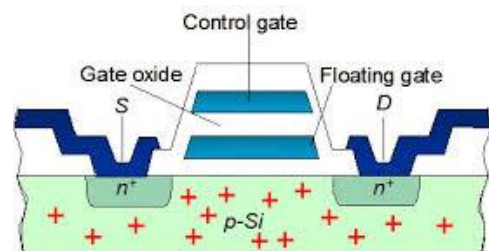
Dióda



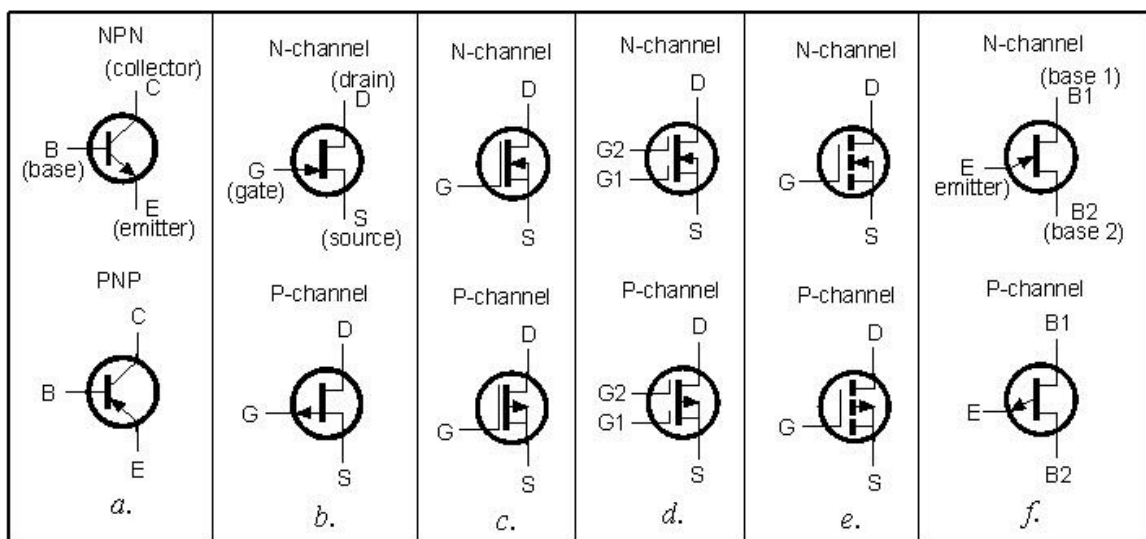
Tranzisztor



MOSFET



Floating Gate MOSFET



a: bipoláris tranzisztor, b: FET, c: MOSFET, d: dual vezérlőelektródás MOSFET,

e: szigetelt vezérlőelektródás FET (IGFET) , f: egyrétegű tranzisztor (UJT)

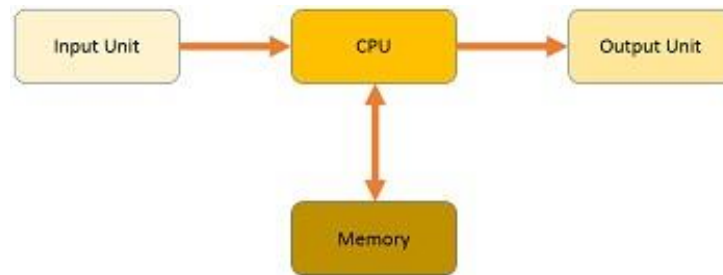
4. Előzmények, mérföldkövek

- Joseph Marie Jacquard (1752-1834)
1810 körül tervezte meg azt az automatizált szövőgépet, mely a mintákat a lyukkártyás vezérlés alapján hozta léte.
- Charles Babbage (1791-1871)
1820 körül tervezte meg a „Difference Engine” nevű teljesen mechanikus gépet, amit a kor műszaki színvonalán nem lehetett megépíteni (pl. a surlódás miatt). A gép az első 7 hatvány- és a logaritmusfüggvény kiszámítását végezte. (1991-ben az eredeti tervek alapján megépült egy tökéletesen működő modell a londoni Science Museum-ban.) 1833-ban tervezett „Analytical Engine” még kormányzati támogatással sem tudott megépülni, mivel egy kb. futballpálya méretű, gőzgéphajtású mechanikus gépről beszélünk. Egy szabadon programozható, matematikai műveletek programozott sorozatát végrehajtó gép lett volna. 1847-49 „Difference Engine2” a tervek szerint hetedfokú egyenlet megoldására lett volna képes 231 decimális értéken. Babbage életében egyetlen általa tervezett gép sem lett készen, de a XX. század végén épült replikák működőképesnek bizonyultak.
- George Boole (1815-1864)
A logikai algebra kifejezés egy 1854-es monográfiában kerül először publikálásra, de maga a Boole algebra – amivel a 4. fejezet foglalkozik részletesen – csak 4 évvel később születik meg, nem is Boole közvetlen közreműködésével, hanem William Jevons és Charles Peirce értekezésében. A Boole algebrát, a matematikai logika gyakorlati hasznosítását, azaz az a nullákkal és egyesekkel végzett logikai műveleteket az informatika a programozás alapjának tekinti.
- Herman Hollerith (1860-1929)
Az Egyesült Államokban az 1880-as népszámlálás alkalmával az 55 millió ember statisztikai adatainak rögzítésére tervezte meg lyukkártyás gépét. Ezen gép sikereire és annak továbbfejlesztéseire épülve alapította meg 1924-ben az IBM-et.
- Alan Turing (1912-1954)
1936-ban fogalmazta meg a gép elvi alapjait:
 - Algoritmus
 - Döntéshozatal: az aktuális pozíció/érték alapján
 - Végrehajtás: módosítás/lépésA Turing gép egy elvileg végtelen, cellákra osztott szalagból, az azt előre-hátra mozgó mechanizmusból, egy író-olvasó fejből és az utasításokat értelmező és végrehajtó központi logikából épül fel, és algoritmizálható matematikai feladatok végrehajtására alkalmas. Modell szinten eredetileg az egyes számrendszer használatával.
Video: [Lego Turing Machine](#)
- Konrad Zuse (1910-1995)
A Zuse által épített Z1 gép évekkel előzte meg a hasonló USA-ban épített MARK-I-et. Sajnos a II. Világháborúban megsemmisültek a gépei, viszont az újraépített működő replikák a Berlini Technikai Múzeumban megtekinthetők.
 - Z1 (1938)
A világon az első, kettes számrendszerben szabadon programozható számítógép, 1Hz (!) órajellel, ami kb. 30.000db elektromechanikus alkatrészből állt. Az adatbevitelhez billentyűzetet illetve lyukszalagot, a kimenetehz világitó mátrixtáblát használt.

- Z2 (1940)
Ez a gép már 5Hz órajellel működött és arányaiban kevesebb mechanikus és több elektronikus alkatrészt tartalmazott. A memória mérete 64 szó (word) volt, és 16 bites fixpontos aritmetikával dolgozott.
- Z3 (1941)
Az órajel itt már 10Hz-re emelkedett, a memória maradt 64 szavas, de 22 bites lebegőpontos számokkal dolgozott. Ezt a gépet tekintjük az első univerzális Turing gépnek.
- Z4 (1945)
Az órajel itt már elérte a 40Hz-et, és 32 bites lebegőpontos számokkal dolgozott.
- John Vincent Atanasoff (1903-1995) és Clifford Berry (1918-1963)
 - ABC (1942)
Az első, tisztán elektronikus alkatrészekből felépített digitális számítógép. A gép órajele 60Hz-es, memóriája 3000 bites. (Az elsőség – nem meglepő módon az USA bírósága előtt dőlt el – az ENIAC ellenében.)
- Alan Turing (1912-1954)
 - Colossus (1943)
A II. Világháború alatt 10db épült ebből az elektronikus számítógépből. A gépek célja alapvetően titkosított üzenetek dekódolása volt. A gép annak a „Turing-Bomba” nevű gépnek a továbbfejlesztése volt, amit az ENIGMA üzenetek dekódolására használtak. (Érdemes nem elfelejteni, hogy az Enigma titkosítását a lengyel Marian Rejewski már 1932-ben feltörte...)
- Howard Aiken
 - Harvard MARK I. (1944)
Hivatalosan a gép megnevezése: „Automatic Sequence Controlled Calculator” (ASCC). A számítógépet a Manhattan Projekt keretében használták, és Neumann János írta hozzá az első programot. (Nem keverendő össze a Manchester MARK I. (1949) számítógéppel, aminek a hivatalos megnevezése: „Manchester Automatic Digital Machine” (MADM).)
- ENIAC (1946)
A hivatalosan „Electronic Numerical Integrator And Computer” nevű gépet a Pennsylvania-i Egyetemen készítették, és főleg katonai ballisztikai számításokra használták. A gép órajele 100kHz volt, és tízes számrendszerben dolgozott. (A Computer elnevezés a röppályát eredetileg kézzel számító matematikusok megnevezéséből származik.)
- EDVAC (1949)
A hivatalosan „Electronic Discrete Variable Automatic Computer” nevű gépet is Pennsylvania-ban, de a Moore School of Electrical Engineering intézetben készítették el, a Neumann elvek mentén, így ez a gép már kettes számrendszert használt. A memória kapacitása 5.6kB volt, azaz képes volt a programok tárolására, nem csak a programfuttatásra.
- UNIVAC (1951) Az első kereskedelmi forgalomban kapható számítógép.
- IBM 305 RAMAC (1956) Az első HDD háttértárral rendelkező számítógép
Video: [Az IBM RAMAC számítógép 5MB merevlemezzel](#)
- IBM 360 (1964) Az első kereskedelmi forgalomban kapható általános célú számítógép.
- Apple II. (1977) Az első 8 bites, tömeggyártásra szánt mikroszámítógép.
- IBM PC (1981) A „Personal Computer”, azaz a PC megjelenése.

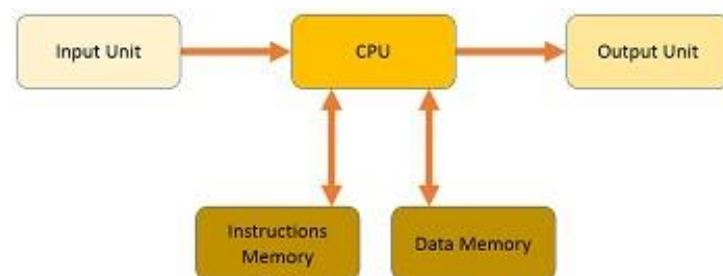
5. Neumann architektúra (1945):

Egy busz rendszer az adatokhoz és az utasításokhoz, összesen egy memória egység.



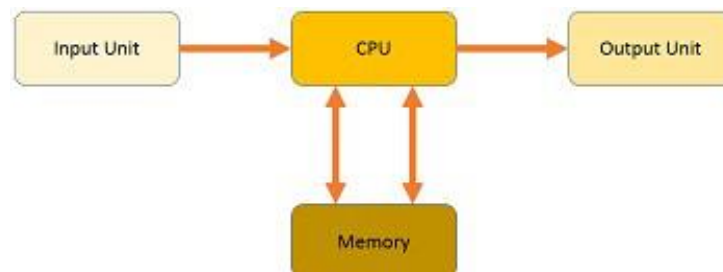
6. Harvard architektúra:

Külön buszon egy időben olvas be adatot és utasítást, két külön memóriából.



7. Módosított Harvard architektúra:

Külön buszon egy időben olvas be adatot és utasítást, azonos memóriából.



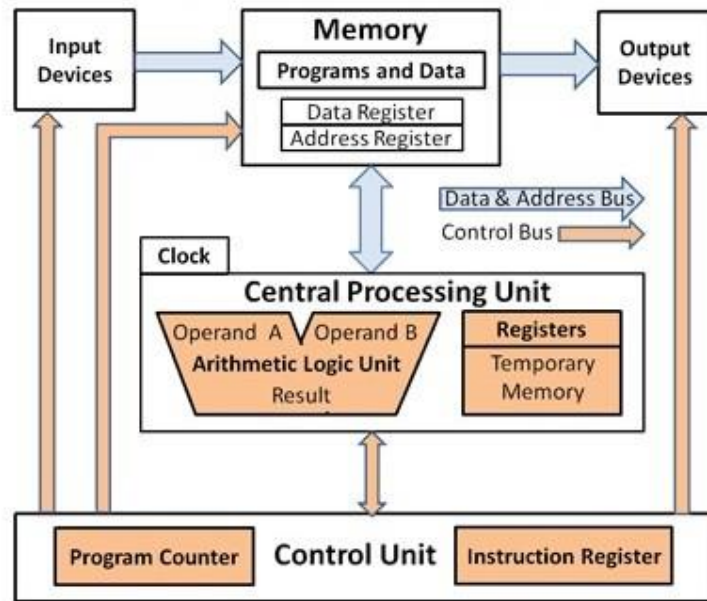
8. Neumann elv (1946) – tárolt program (Univerzális Turing gép)

- A számítógép vezérlését tárolt program végzi. (Turing)
- A vezérlést vezérlés-folyam (control-flow) segítségével lehet leírni.
- A gép belső tárolójában a program utasításai és a végrehajtásukhoz szükséges adatok egyaránt megtalálhatók (közös utasítás és adattárolás, a program felülírhatja magát illetve nyilván az adatokat is – azaz az egész memóriát).
- Az aritmetikai és logikai műveletek (a programutasítások) végrehajtását önálló részegység, az ALU végzi.
- Az adatok és programok beolvasására és az eredmények megjelenítésére önálló egységek (perifériák) szolgálnak.
- A 2-es (bináris) számrendszer alkalmazása.

Második fejezet

1. A számítógép felépítése:

- 5 alapvető egység
 - bemenet
 - memória
 - aritmetikai egység
 - vezérlőegység
 - kimenet



A mikroszámítógép elnevezés természetesen nem objektíve a gép méretét jelöli. Az elnevezés azt jelenti, hogy a gép központi egysége a mikroprocesszor, nem pedig diszkrét elektronikai (pláne nem mechanikai) elemek sokasága.

2. A Mit és a Hogyan kérdései:

• IMPERATÍV

A programozó mondja meg, hogy mit és hogyan csináljon a program.

Szekvenciális algoritmus.

Az általánosan használt programozási nyelvek imperatívák.

• DEKLARATÍV

A programozó állításokat közöl a géppel/programmal. Az állítások logikai összefüggéseket tartalmaznak, amiből a gép/program képes logikus következtetéseket levonni. (pl. Én, Apám, az Apám Apja. Kérdés: Ki a Nagyapa?)

A deklaratív programozás két fő iránya a logikai programozás (LP) és a funkcionális programozás (FP). A logikai programozás fontos kiterjesztése a korlát-logikai programozás (Constraint Logic Programming, CLP), amely egy adott speciális területen különösen hatékony következtetési módszereket nyújt.

3. Az első diáor [01_ch01_hun_v01_hefop_v02.ppt]

Általános PC, CPU, Memória, Számítógép rendszerek, Protokollok, Szabványok.

Hiba: #11-es dia → Random Access Memory = Tetszőleges Hozzáférésű Memória

A „random” azaz tetszőleges (tehát ez esetben a jelentés NEM a „véletlenszerű” szó!) kifejezés a „sequencial” azaz szekvenciális, kötött sorrendű hozzáférés ellentétéként értelmezendő!

4. A mikroprocesszor fejlődésének rövid története

4bit [1971.09.17 Texas Instruments TMS1000; majd Intel 4004 MCS-4; AMD Am2901]

8bit [Intel 8008, 8080, 8086, 8088; Fairchild F8; Zilog Z8; Motorola MC6800]

16bit [Intel 80186, 80286]

32bit [Intel 80386, 80486; AMD K5, K6, Duron, Athlon]

64bit [Intel Pentium, Core, Core2, i3, i5, i7, i9; AMD Duron, Sempron, Athlon, Phenom, Turion]

128bit-es és 256bit-es CPU-k is léteznek, de a mindennapokban ezekkel még nem találkozunk

5. Software, Hardware, Firmware, BIOS, UEFI

- SW: minden, ami nem kézzel fogható, azaz program
- HW: minden, ami kézzel fogható (hétköznapi elnevezése: „vas”)
- FW: Egy adott, jellemzően nem rugalmasan bővíthető HW-ra kreált működtető, vezérlő SW, amit felhasználó (elvileg) módosítani nem tud, csak lecserélni frissebbre, javítottra, más célra átalakítottra (pl.: router, HDD, I/O eszközök, autó, sütő, stb.)
- BIOS: Alapvető Be- és Kimeneti Rendszer (Basic Input Output System) Egy adott, jellemzően rugalmasan bővíthető HW-ra kreált működtető, vezérlő SW, amit felhasználó (elvileg) módosítani nem tud, csak lecserélni frissebbre, javítottra, más célra átalakítottra, viszont a kapcsolódó I/O eszközöket a felhasználó paraméterezheti. A BIOS biztosítja a kapcsolatot a HW-ek és a HW-ekre telepített operációs rendszer között.
- UEFI: Univerzálisan Bővíthető Firmware (Universal Extensible Firmware Interface). A BIOS azon korlátait hivatott javítani, melyek a BIOS elmúlt kb. 30 évéből és az ebből fakadó, egyre többször jelentkező kompatibilitási problémáiból ered. Fő jellemzői:
 - Gyorsabb sebesség, grafikus felület a karakteres felület helyett.
 - Modulárisan frissíthető.
 - Közvetlen natív x86, x64, Itanium és ARM architektúra támogatás.
 - GPT (GUID – Globally Unique Identifiers Partition Table) partíció támogatás, diszkenként akár 4db 2.2TB-os elsődleges partícióval, maximális diszk méret 16.8EB (Exa = 10^{18}), akár több mint 100 partícióval diszkenként.
 - Az x64 architektúrán akár 17.2EB memória megcímezhető.
 - Rootkit védelem.

Alaplapok esetében hasznos dolog a: „POST” → Power On Self Test, a „csipogó BIOS kódok”.

<http://www.computerhope.com/beep.htm>

Harmadik fejezet

1. Számrendszerek (második diasor [02_ch02_hun_zs1.ppt] végig)

A digitális technika alapjai. A kettes számrendszer, tizenhatos számrendszer.

Magyarázat: #27-es dia: $13 \times 5 = 65$; #28-as dia: $109 \times 38 = 4142$; #29-es dia: $761 \times 7 = 5327$

Hiba: #41-es dia $\rightarrow 9 \times 1 / 10000$ helyesen a 0,0009 !!!

2. Számábrázolás I. (harmadik diasor [03_ch4_hun_zs2.ppt]:

Az egész számok ábrázolása, a #26-as diáig)

A „számrendszer” szó, mint fogalom azt jelenti, hogy az egyes helyiértékeken, a számrendszer alapjának a helyiérték sorszáma mínusz egység kitevőjű hatványai találhatóak.

Kettes számrendszerben, az előjel nélküli egész számokat 0-tól 2^{n-1} -ig tudjuk ábrázolni.

3. Számábrázolások összehasonlítása, Fixpontos ábrázolás

Fixpontos bináris kódok (4 biten)							
Egyenes (Abszolútértékes)		Kettes komplement		Többletes		Egyes komplement (Negációs)	
-7	1111	-8	1000	-8	0000	-7	1000
-6	1110	-7	1001	-7	0001	-6	1001
-5	1101	-6	1010	-6	0010	-5	1010
-4	1100	-5	1011	-5	0011	-4	1011
-3	1011	-4	1100	-4	0100	-3	1100
-2	1010	-3	1101	-3	0101	-2	1101
-1	1001	-2	1110	-2	0110	-1	1110
-0	1000	-1	1111	-1	0111	-0	1111
+0	0000	0	0000	0	1000	+0	0000
+1	0001	+1	0001	+1	1001	+1	0001
+2	0010	+2	0010	+2	1010	+2	0010
+3	0011	+3	0011	+3	1011	+3	0011
+4	0100	+4	0100	+4	1100	+4	0100
+5	0101	+5	0101	+5	1101	+5	0101
+6	0110	+6	0110	+6	1110	+6	0110
+7	0111	+7	0111	+7	1111	+7	0111

- Egyenes kód (abszolútértékes kód)
A pozitív számokat és a nullát egy vezető zérussal, e számok ellentettjét egy vezető egyessel írjuk. Például: $+1 = 0001$, és $-1 = 1001$. E kód jellegzetessége a negatív nulla, ami formálisan képezhető a nulla szám 0000 kódjából az előjelbit átváltásával: 1000.
- Komplement kód (kettes komplement kód)
A legáltalánosabb fixpontos bináris kód a nullát kódoló 0000-ból előre ($+1 = 0001$) és vissza ($-1 = 1111$) számlálással adódik (ahogyan a decimális műszereken a 0000-től visszaforgatással a 9999).
- Többletes kód
Ebben a kódban minden számot 2^{n-1} többlettel ábrázolunk. Az $n = 4$ bit esetében ez a többlet $2^3 = 8_{10} = 1000_2$. Ennél a kódnál a negatív előjelbit 0, és nincs ábrázolva negatív 0 érték.

- Negációs-kód (egyes komplement kód)

A pozitív számokat és a nullát ugyanúgy írjuk, mint a természetes számírásban. A pozitív számok ellentettjét a jegyek mindegyikének átfordításával (negáció) kapjuk.

Például: $+1 = 0001$ és ebből $-1 = 1110$. Ebben a rendszerben is létezik a negatív nulla: 1111

A negatív nulla az érintett számábrázolási módokban műveletvégzés eredményeként is keletkezhet. Negatív nulla keletkezik például a nulla szám és egy negatív szám szorzásakor az egyenes kódban, illetve két azonos abszolút értékű, de ellentétes előjelű szám összeadásakor negációs kódban.

4. Számábrázolás II. (harmadik diáor [03_ch4_hun_zs2.ppt] a #27-es diától a végéig)

Magyarázat: #30-as dia: Átvitel bit (Carry) és Túlcsordulás (Overflow) / Alulcsordulás (Underflow)

- Átvitel bit: Jelzi, ha egy adott helyiértéken elvégzett művelet eredménye meghaladja az ott ábrázolható értéket.
- Túlcsordulás (illetve alulcsordulás): ha egy összeadás vagy kivonás művelet eredménye kívül esik az aktuális számábrázolási tartományon.

5. Lebegőpontos számok (harmadik diáor [04_ch05_hun_zs3.ppt] végig)

A lebegőpontos szám lényege, hogy az ábrázolásánál a tizedespont „lebeg”, vagyis az ábrázolható értékes számjegyeken belül bárhova kerülhet. (Példa erre az 1,23; 12,3; 123 számok, melyek mindegyike 3 értékes számjegyet tartalmaz.) A lebegőpontos ábrázolás előnye a fixpontos számábrázolással szemben az, hogy sokkal szélesebb tartományban képes értékeket felvenni.

Magyarázatok, kiegészítések a diáorhoz:

Az #5-ös dián a „8 bit” helyett pontosabb a „8 digit” vagy „8 számjegy” kifejezés használata, mert nem bináris számábrázolásról van szó!

A #7-es dián az alulcsordulás és a túlcsordulás bemutatásához az ábrázolható legnagyobb, illetve legkisebb számot használjuk fel pozitív és negatív előjellel. A legnagyobb szám a $0,99999 \times 10^{49}$, a legkisebb szám a $0,00001 \times 10^{50}$, ami átrendezés után így néz ki: 1×10^{55} . Ez az oka, hogy itt a kitevőben 55, és nem 50 szerepel!

6. Excess N alak – Eltolt ábrázolás

Az Excess N ábrázolás lényege az, hogy azt számtartományt, mely a kitevőt (karakterisztikát) jelzi két részre osztjuk. Az „N” a felezőérték, melyhez ez után (valós értékétől függetlenül) a nulla kitevőt rendeljük. Az „N”-nél kisebb értékek a negatív, az „N”-nél nagyobb értékek a pozitív kitevőket jelentik.

Excess 50 esetében a kitevő a következő táblázat szerint alakul.

Számtartomány	0	1	...	47	48	49	50	51	52	53	...	98	99
Ábrázolt kitevő	-50	-49	...	-3	-2	-1	0	1	2	3	...	48	49

Amennyiben a kitevő ábrázolására két helyiérték áll rendelkezésünkre, – tízes számrendszerben értelmezve – akkor a 0 és a 99 között vehetünk fel értékeket. Excess 50 esetében ezt a 100db értéket olyan módon osztjuk ketté, hogy az 50-es érték a 0 értéken ábrázolt kitevőt jelezze. A kitevő értéke az 50-től jobbra haladva, illetve balra haladva eggyel növekszik, illetve eggyel csökken.

A rendelkezésre álló 100db értéket így (amennyiben a 0 értéket külön kezeljük, végeredményben nem kettő, hanem három részre) szétosztva, $(1 \times 50) + 1 + (1 \times 49)$ érték ábrázolhatóságát hoztuk létre.

Az ábrázolt kitevő negatív tartományában a -50 a szélsőérték (minimum), míg a pozitív tartományában a +49 a szélsőérték (maximum). A jelenség oka természetesen a 0 érték, ami kihat egyet a pozitív értékek közül.

Az „eltolt ábrázolás” kifejezés arra utal, hogy a számtartományon belül az ábrázolt kitevő (akár) egy eltolás művelettel (is) kiszámítható.

A #12-es dián szöveges leírás van arról a folyamatról, amit a #8-10-es dián csináltunk.

A #13-as dián az első (azaz a nulladik) bit jelzi, hogy a mantissza pozitív, vagy negatív.

A következő 8 bit ábrázolja a kitevőt Excess-128 formában, azaz a kitevő $2^8=256$ féle érték lehet, legkisebb értéke -128, legnagyobb értéke +127. A maradék 23 bit fejezi ki a mantisszát.

A #14-es dián logikailag a fekete sorokkal kell kezdeni, mert azokat alakítjuk át.

7. IEEE 754 (I tripple E) szabvány

Az IEEE 754/1985 szabvány szerint 4 féle – csak kettes számrendszerben értelmezendő – lebegőpontos szám ábrázolási formátum létezik, melyek csak az adott elem bitszámában térnek el egymástól. Az 1985-86-os évben került elfogadásra az IEEE 754-es (854) számú lebegőpontos számaábrázolási szabvány, melyet a legfontosabb processzorgyártók is elfogadtak.

Tulajdonságai:

- a mantissza előjele 0, ha a szám pozitív; és 1, ha negatív;
- a mantisszában levő fixpontos szám 1-re normalizáltan értendő, azaz 1. a formájú;

A szabvány a lebegőpontos művelet végrehajtáshoz többfajta pontosságot definiál:

Pontosság	Hossz	Előjel	Mantissza	Karakterisztika	Számtartomány
egyszeres	32 bit	1 bit	23 bit	8 bit	$8,43 \cdot 10^{-37} < N < 3,37 \cdot 10^{38}$
dupla	64 bit	1 bit	52 bit	11 bit	$4,19 \cdot 10^{-307} < N < 1,67 \cdot 10^{308}$
kiterjesztett	80 bit	1 bit	64 bit	15 bit	$3,4 \cdot 10^{-4932} < N < 1,2 \cdot 10^{4932}$
négyszeres	128 bit	1 bit	112 bit	15 bit	

(Vegyük észre, hogy a #14-es dián a karakterisztika és a mantissza előjele nem azonos módon van jelölve. A #15-ös dia után a #16-os nem kell, logikailag a #19, #20-as dia következik.)

8. Excess 128-as alak és az Excess 127-es alak

Az Excess 128 esetében a mantissza „klasszikus” alakú, azaz a bázispont (tízes számrendszer esetében a tizedesvessző) az első nem nulla karakter elé kerül.

Az Excess 127 esetében a mantissza kvázi „törtként szerepel”, de így nyerünk egy helyiértéket, ami növeli az ábrázolható pontosságot. A visszaállításkor erre külön ügyelni kell.

9. Átalakítások

Magyarázat: #18-as dia: a „vagy” után a kék színű szám egy sorba írandó.

A kitevő (10001101_2) értéke 141_{10} , ami az Excess 127-es alak miatt $127_{10}+14_{10}$ értelmű, (ábrázolva $128_{10}+13_{10}$ van) azaz a kitevő valós értéke 14_{10} .

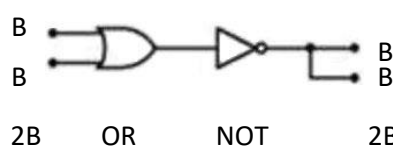
Az Excess 127 miatt a mantissza az utolsó sorban csak a bázispont utáni részével szerepel!

Negyedik fejezet

1. Matematikai logika alapjai, kapcsolat a kettes számrendszerrel, elektronikai megvalósítások.

ÉS	konjunkció	AND	$A \times B$	A és B	soros kapcsolás
VAGY	diszjunkció	OR	$A + B$	A vagy B	párhuzamos kapcsolás
KIZÁRÓ VAGY	antivalencia	XOR	$A \oplus B$	vagy A vagy B	
TAGADÁS	negáció	NOT	$\neg A$	nem A	
ÖSSZEFONÓDÁS	implikáció		$A \Rightarrow B$	ha A akkor B	
EGYENÉRTÉKŰSÉG	ekvivalencia	XNOR	$A \Leftrightarrow B$	ha A akkor B, és ha B akkor A	
értéktáblázatok:	lásd az XLS két munkalapja				
áramköri megvalósítások (megemlíteni: NAND, NOR, szekvenciális hálózatok, RS tárolók, stb.)					

Logikai kapcsolat		Igazságtáblázat			Rajzjel	
művelet	elnevezés	B	A	Y	angol	nemzetközi
$A \cdot B$	ÉS (AND)	0 0 1 1	0 1 0 1	0 0 0 1		
$A + B$	VAGY (OR)	0 0 1 1	0 1 0 1	0 1 1 1		
\bar{A}	NEM (NOT)		0 1	1 0		
$\overline{A \cdot B}$	NEM-ÉS (NAND)	0 0 1 1	0 1 0 1	1 1 1 0		
$\overline{A + B}$	NEM-VAGY (NOR)	0 0 1 1	0 1 0 1	1 0 0 0		
$A \oplus B$	ANTIVALENCIA (XOR)	0 0 1 1	0 1 0 1	0 1 1 0		
$\overline{A \oplus B}$	EKVIVALENCIA (XNOR)	0 0 1 1	0 1 0 1	1 0 0 1		



[To be, or not to be. – Lenni, vagy nem lenni.]

A 2. és a 3. pont részletes tárgyalása főleg a GAZDINFO-s hallgatók részére kiemelt fontosságú, akik nem vettek részt a Somogyi Miklós által tartott „Digitális logikai rendszerek” kurzuson.

2. Együtemű vezérlésű tárolók (flip-flop)

Az együtemű tároló elnevezés arra vezethető vissza, hogy ezen eszközök jellemzője az, hogy a bemenetére érkező logikai szintek megjelenésével gyakorlatilag egy időben (azaz csak kapunyi késleltetéssel) létrehozható a kimenetek megváltozása.

A logikai szintek megfeleltethetők a kettes számrendszer elemeinek:

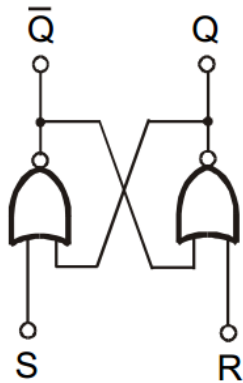
„H” (high/magas) = 1

„L” (low/alacsony) = 0

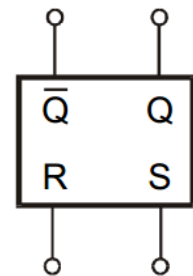
A bemenetek jelölése: „S” (set / beállítás) illetve „R” (reset / törlés)

A kimenetek jelölése: „Q” illetve „ \bar{Q} ”

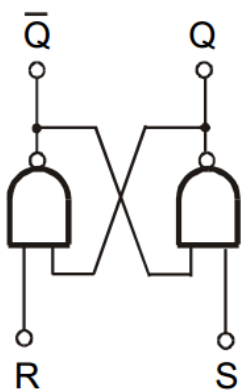
A NOR kapukból épített tároló az „S”-re érkező „H” bemeneti szintre „H” kimeneti szinttel válaszol (természetesen „L” szintű „R” esetén). Metastabil (azaz bizonytalan) helyzet akkor áll elő, amikor mindkét bemenetre „H” szintű jel érkezik.



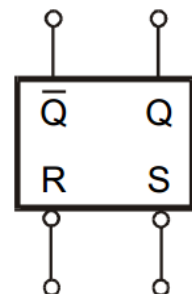
S	R	Q^{T+1}
L	L	Q^T
L	H	L
H	L	H
H	H	tiltott



A NAND kapukból épített tároló az „S”-re érkező „L” bemeneti szintre „H” kimeneti szinttel válaszol (természetesen „H” szintű „R” esetén). Metastabil (azaz bizonytalan) helyzet akkor áll elő, amikor mindkét bemenetre „L” szintű jel érkezik.



S	R	Q^{T+1}
L	L	tiltott
L	H	H
H	L	L
H	H	Q^T

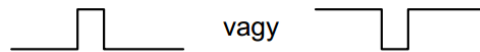


3. Hazárd Jelenségek

- Statikus hazárd

A kombinációs hálózat egyik bemenetének megváltozásakor jön létre a statikus hazárd. Kiváltó oka az az áramköri megvalósítás, amikor egy logikai kapu két bemenetére a bemenő jelek nem egy időben, hanem kis késéssel (pl. egy „kapunyi” késleltetéssel) érkeznek egymáshoz képest. Mivel tervezési hiba okozza, célszerű áttervezni, vagy átalakítani a hálózatot.

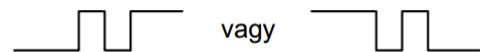
A kimeneten ezt tapasztaljuk:



- Dinamikus hazárd

Dinamikus hazárd csak 2-nél többszintű hálózatban jöhet létre. Feltétele, hogy egy jel legalább 3 úton terjedjen a kimenetre. Kivédni a statikus hazárdok megszüntetésével lehet.

A kimeneten ezt tapasztaljuk:



- Funkcionális hazárd

Akkor jelentkezhet funkcionális hazárd, ha több bemeneti változó változik egyszerre. Az aszinkron sorrendi hálózatokban ez nem kívánt állapotátmeneteket okozhat, a kimeneten vagy a specifikációtól eltérő szintváltás, vagy többszörös szintváltás jelentkezik. A funkcionális hazárd kivédésének a legbiztosabb módja a szinkronizáció.

- Lényeges hazárd

Az úgynevezett lényeges hazárd csak aszinkron sorrendi hálózatokban jelentkezik. Az első integrált áramkörök megjelenése keltette életre ezt a jelenséget, mert itt a szekunder változó és a bemeneti változó sebessége már összemérhető versenyhelyzetet teremtett. Kiküszöböléséhez szándékos késleltetést kell alkalmaznunk vagy módosítanunk kell az állapottáblát, ami jellemzően állapotszám növekedést is jelent.

- Rendszerhazárd

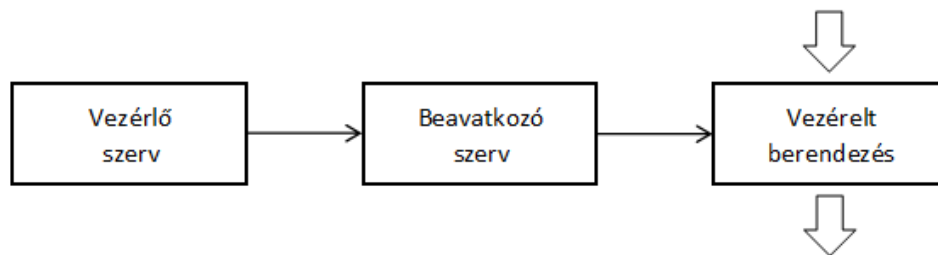
Több szekunder változó esetén fordulhat elő a rendszerhazárd. A jelenség oka, hogy a visszacsatoló ágban lévő flip-flopok bemenetén jellemzően eltérő időkésleltetések fordulnak elő. Ez a gyakorlatban azt jelenti, hogy az egyik flip-flop jóval gyorsabb a többinél és a gyors kimenete már visszahat, így a többi flip-flop már nem az eredeti állapotból vesz mintát, hanem ebből a módosítottból.

Ötödik fejezet

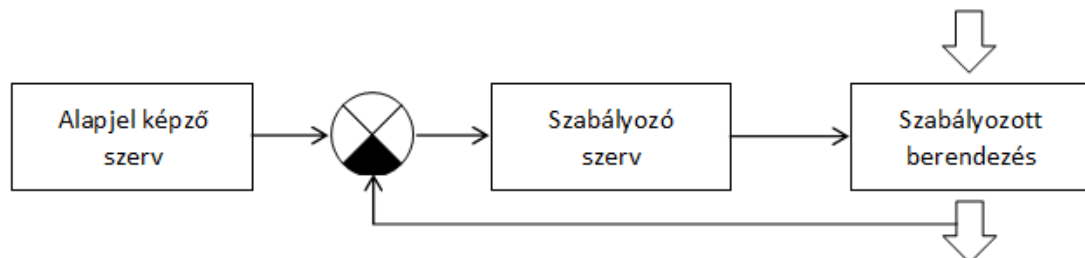
1. Irányítás, Vezérlés, Szabályozás, Automatizálás

Az irányítás olyan művelet, mely beavatkozik valamely műszaki folyamatba annak létrehozása (elindítása), fenntartása, tervszerű lefolyása, megváltoztatása vagy megszüntetése céljából. A legfontosabb részműveletek: a működési paraméterek érzékelése illetve mérése, ezek alapján az ítéletalkotás, majd a beavatkozás. Az irányítás jellegét tekintve lehet kézi- illetve önműködő irányítás.

A vezérlés nyitott hatásláncú folyamat, visszacsatolás nélkül. A beavatkozás után (illetve közben) nem értesülünk a beavatkozás hatásairól és következményeiről. A cél egy előre meghatározott műveletsor végrehajtása. [Például a gépi megmunkálás is jellemzően vezérlések sorozata.]



A szabályozás zárt hatásláncú folyamat, visszacsatolással. A beavatkozás után (illetve közben) folyamatosan értesülünk a beavatkozás hatásairól és következményeiről. A szabályzással irányított jellemző – például a hőmérséklet – pillanatnyi értékét össze tudjuk hasonlítani a megkívánt értékkel (alapjel). A cél az alapjel elérése és megtartása. [Például termosztátos fűtés-szabályozás, ahol a termosztát ki- és bekapcsolja a kazánt a fűtendő szoba hőmérséklete alapján.]



Az automatizálás olyan műszaki-gazdasági tevékenység, melynek eredményeképpen az ember közvetlen termelésirányító tevékenységét készülékek, berendezések, gépek veszik át. Az automatizálás önműködő készülékek, berendezések, gépek alkalmazásával az ember (monoton) irányító tevékenységét kiküszöbölve a termelés folyamatából, lényegében önműködő irányítást hoz létre. Az automatizálás a termelés ma ismert legfejlettebb formája.

Az irányítás, a vezérlés, a szabályozás és az automatizálás legmagasabb szinten informatikai, azaz számítógépes támogatással valósul meg.

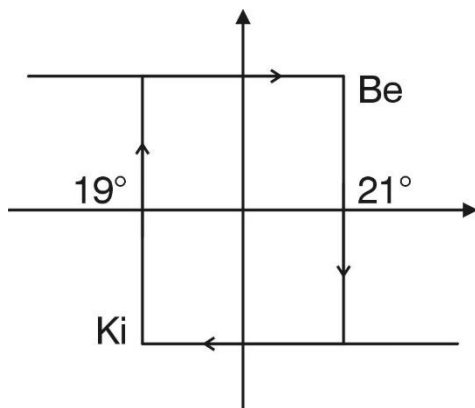
2. Hiszterézis

Egy hiszterézissel rendelkező rendszer pillanatnyi állapota függ a korábbi állapotától, a rendszer előéletétől. Azt a jelenséget is hiszterézisnek nevezzük, amikor a rendszer időkésséssel reagál a bemenő jelre. Egy hiszterézises rendszerben, ha nem ismerjük a rendszer pillanatnyi állapotát, akkor egy adott bemenő jel esetén nem lehet előre meghatározni a rendszer választát, kimenetét, mert a kiinduló állapot ismeretlen.

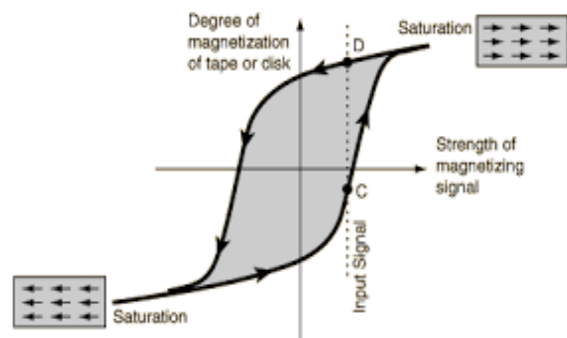
A hiszterézis szó a görög nyelvből ered, és hiányosságot, késést jelent. A hiszterézis szinte minden tudományágban megjelenik. Létezik: rugalmas, elektromos (pl. ferroelektromos anyagoknál), mágneses (ferromágneses anyagoknál), szilárd-folyadék fázisátalakulásnál, elektronikai, gazdasági és biológiai rendszerekben fellépő hiszterézis.

A legáltalánosabb példa a hiszterézisre egy termosztát, amiben egy elektronikus szabályozás elvén működő kapcsoló (hiszterézises Schmitt trigger) működik.

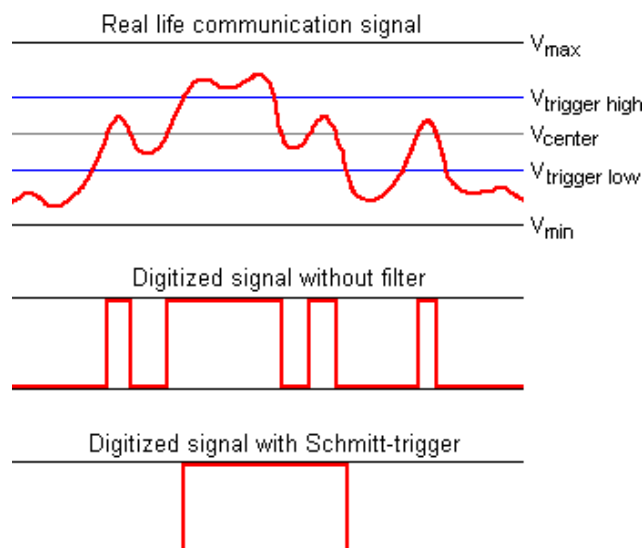
A termosztát hiszterézise



Ferromágneses anyag hiszterézise

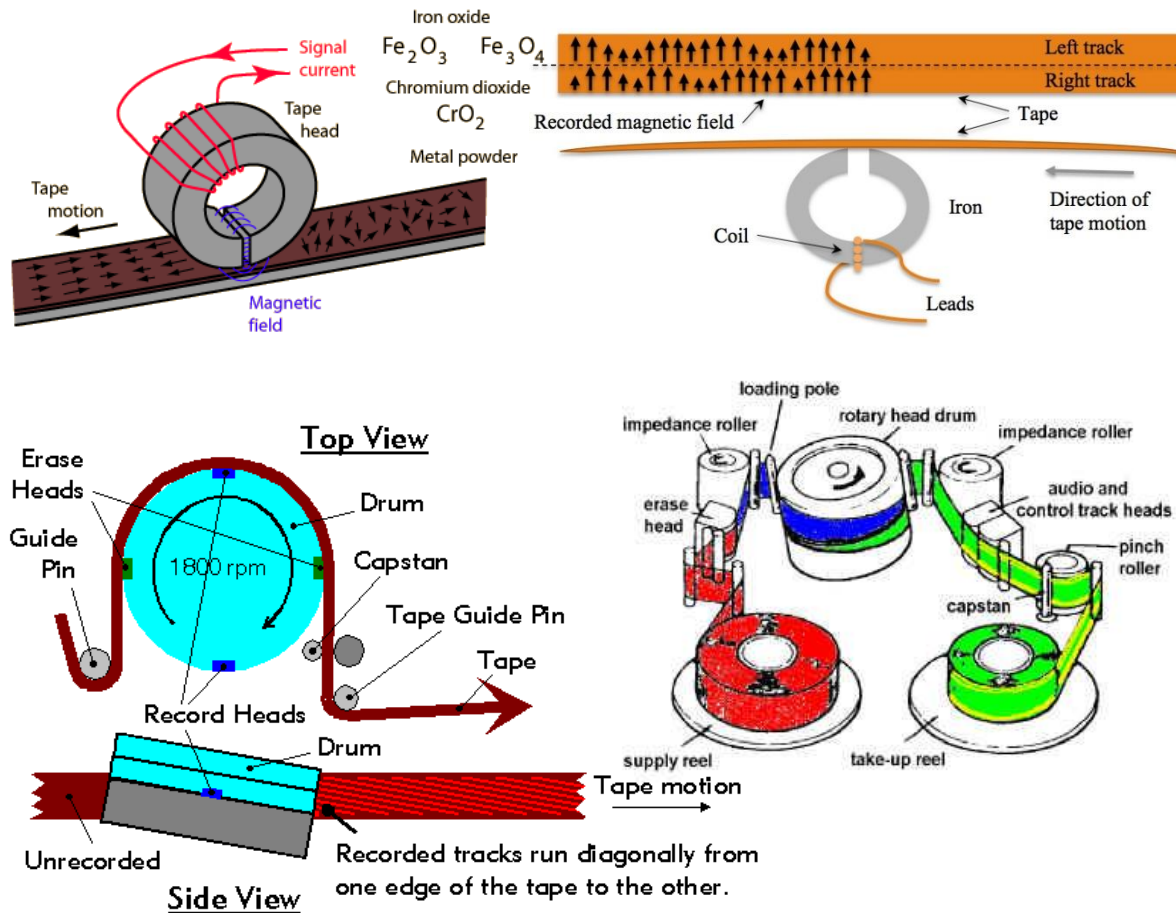


Analóg jel digitalizálás Schmitt triggerrel



3. A mágneses adattárolás alapelvei (magnószalag, videoszalag, adatszalag, adatlemez)

A mágneses adattárolás analóg alapjait képező berendezések, (a magnetofon és a videomagnó) időben változó elektromos jelek mágneses rögzítésére, és visszajátszására alkalmas készülékek.



Video: [A mágnesszalag története](#)

Video: [A VHS és a BETA videórendszerek összehasonlítása](#)

A készülék a bemenetére juttatott villamos jelet mágnesezhető anyagra rögzíti. Felvétel során a hordozó elhalad a törlőfej előtt, ez a hordozón lévő esetleges korábbi felvételt váltakozó áramú demagnetizálással megszünteti, illetve előmágnesezi (AC BIAS). Ezt követően hordozó a felvevőfej elé jut, amelynek tekercsbe a bemenetre adott, majd felerősített villamos jelet vezetik. A tekercsben a rögzíteni kívánt jelnek megfelelő összetett jelalakú és frekvenciájú, változó erősségű áram folyik. A fej kialakítása olyan, hogy a bemeneti változó jel által létrehozott mágneses tér, egy kis légréven keresztül záródjon. Ez előtt a légréves fej előtt halad el a hordozó, mely a változó mágneses térnek megfelelően felmágneseződik. A felvétel minősége szempontjából lényeges a hordozónak a fej előtti elhaladásának sebessége, valamint a réstől való távolsága. Mechanikai kényszer biztosítja a hordozónak a fejre való jó felfekvését.

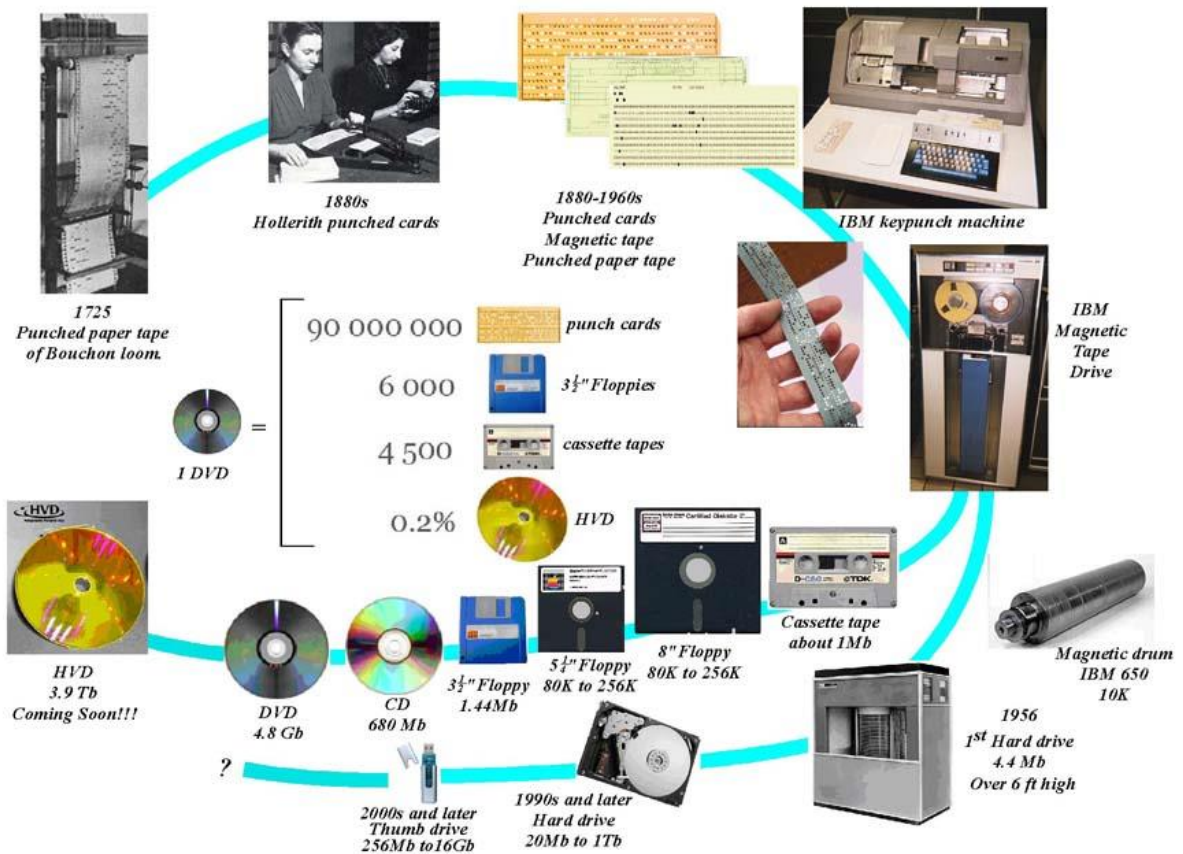
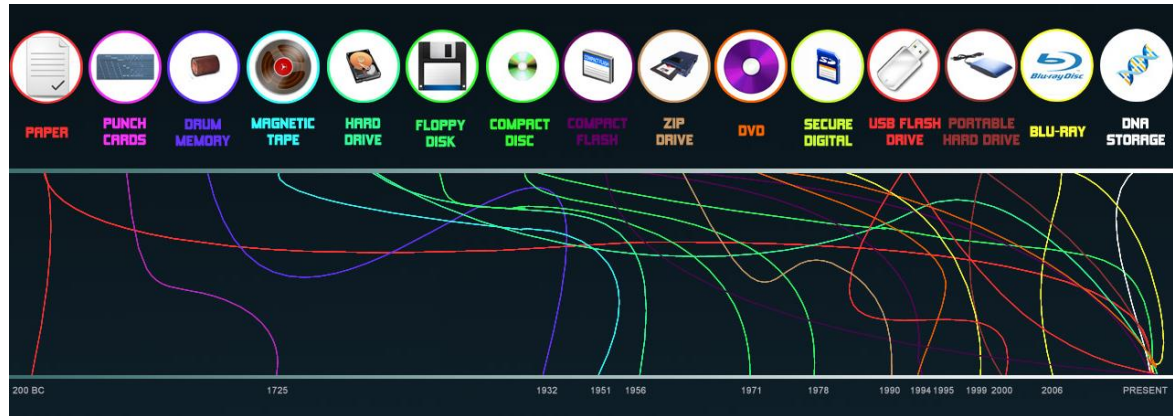
Lejátszáskor a törlőfejnek nincs funkciója. A hordozó elhalad a lejátszófej előtt, és a tárolón lévő mágneses térerősség változása a felvett jelnek megfelelő mértékben felmágnesezi a lejátszófej vasmagját. A vasmagban jelentkező erővonalasűrűség-változás hatására a rajta kialakított tekercsben feszültség indukálódik. Mivel a tekercs egy zárt áramkör része, ezért áram folyik benne. Ez az átfolyó áram arányos a korábban rögzített jellel. Így gyakorlatilag (eltekintve a zavaroktól és a torzításoktól) az eredetileg rögzített jelet kapjuk vissza.

4. Háttértárak

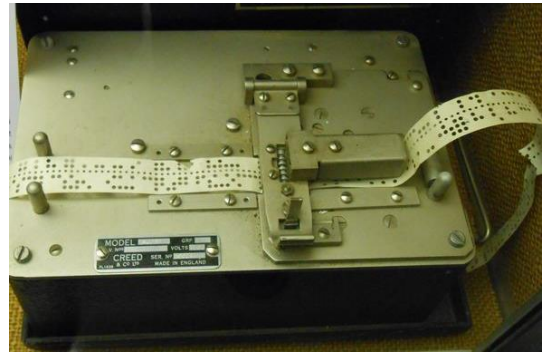
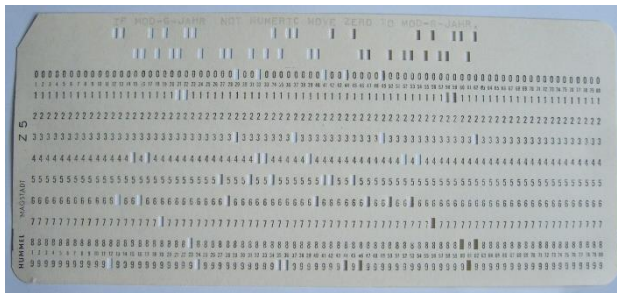
régi: lyukkártya, lyukszalag, ferritgyűrű

modern: mágneses, optikai, FLASH alapú; Streamer, Floppy, HDD, CD, DVD, Blu-Ray, SSD, SSHD, memóriakártyák (SD, CF, MS, stb.)

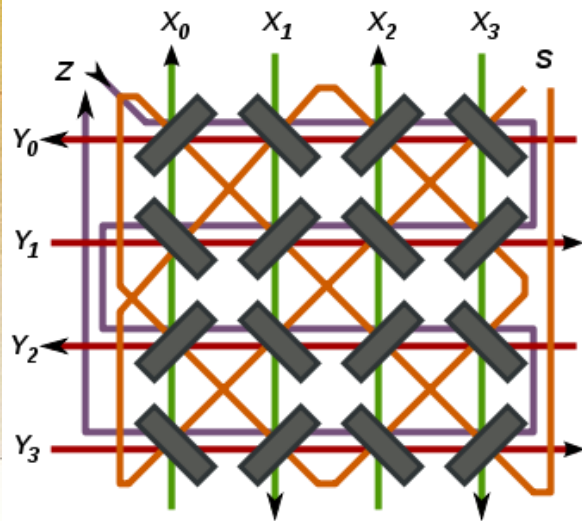
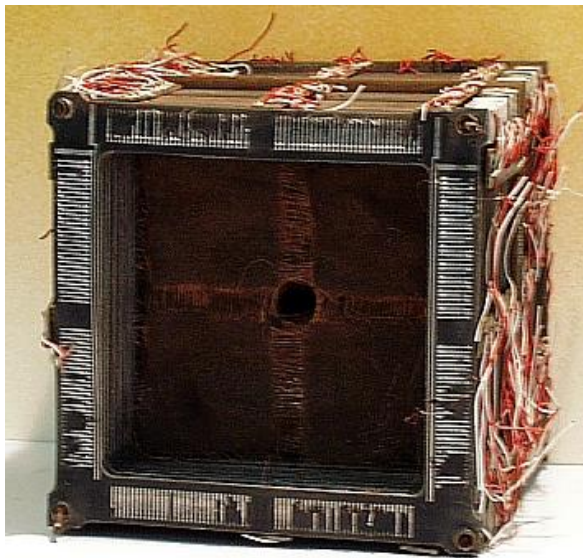
Kapcsolódó fogalmak: Biztonsági Mentés [adatvesztés kiküszöbölésére] illetve az Archiválás [cél a közvetlen kereshetőség] elvi különbségei; RAID technológia.



5. Lyukkártya, Lyukszalag



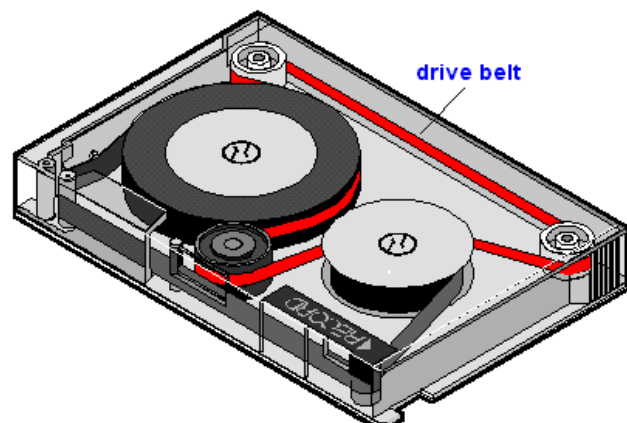
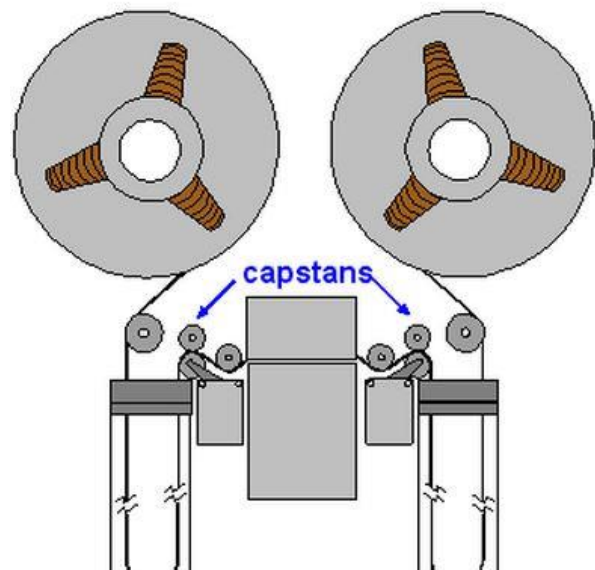
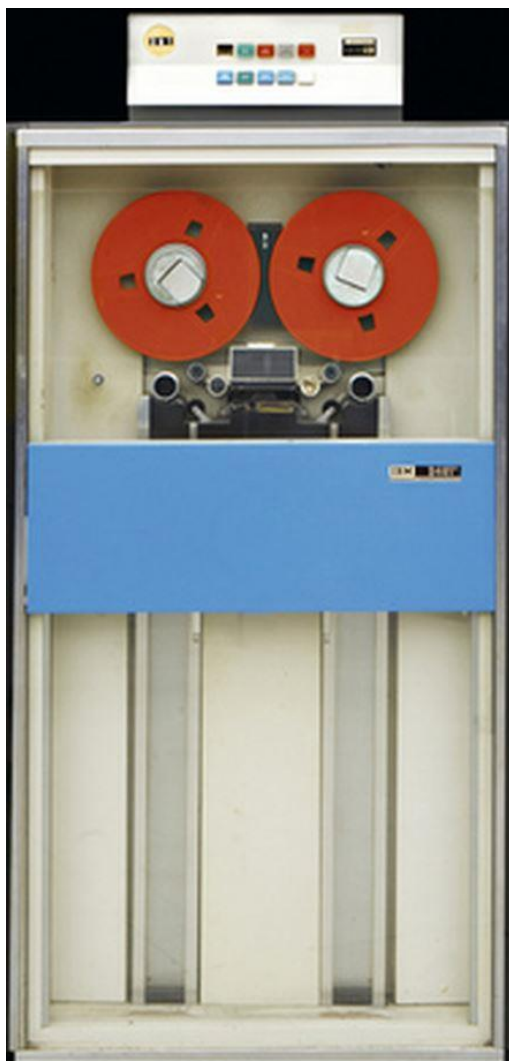
6. Ferritgyűrűs tárcák



Egy tárolóelem (azaz egy gyűrű) írása vagy törlése egy vezetékkel is megoldható, több bit esetén viszont külön vonal lenne szükséges minden egyes elem írásához. A megoldás a tárolóelemek mátrixba kötése: a mágnesezhető gyűrűk a függőleges, és vízszintes vezetővonalak metszéspontjaira vannak felfűzve. Ha az egy bit írásához (ferritgyűrű átmágnesezéséhez) szükséges áramerősség felét engedjük meg minden vonalon, akkor tárolási művelet csak a metszéspontban fog végbemenni, az adott vezetéken lévő többi gyűrű nem mágneseződik át.

A statikus mágneses állapot egyszerűen nem olvasható, mivel csak a mágneses tér változása indukál feszültséget a vezetőkben. A ferritgyűrű állapota úgy olvasható, ha azt vizsgálja az olvasó rutin, hogy 0 értékre írási kísérletre változik-e a tároló cella állapota, vagyis az olvasó (S / Sense) vezetéken jelenik-e meg indukált feszültség. Ha az adott bit 0 értékű volt, akkor nem jelenik meg indukált feszültség. Ha az adott bit értéke 1 volt, akkor a 0-ra írás után megjelenik a feszültség. Olvasás után az eredeti értéket – amennyiben az olvasás során megváltozott – nyilván vissza is kell írni.

7. Szalagos tárák



8. Modern Tape Library



Video: [A Tape Library működése I.](#)

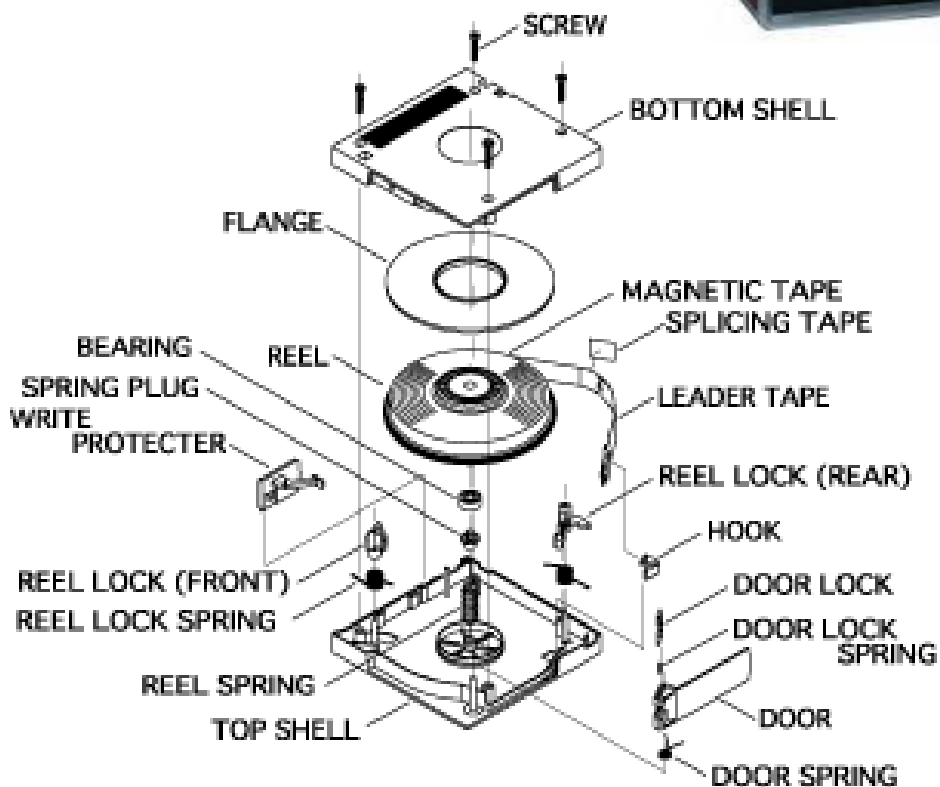
Video: [A Tape Library működése II.](#)

Video: [A Tape Library működése III. \(NCAR\)](#)

Video: [A Tape Library működése IV. \(Spectra T950\)](#)

Video: [A Tape Library működése V. \(HP ESL G3\)](#)

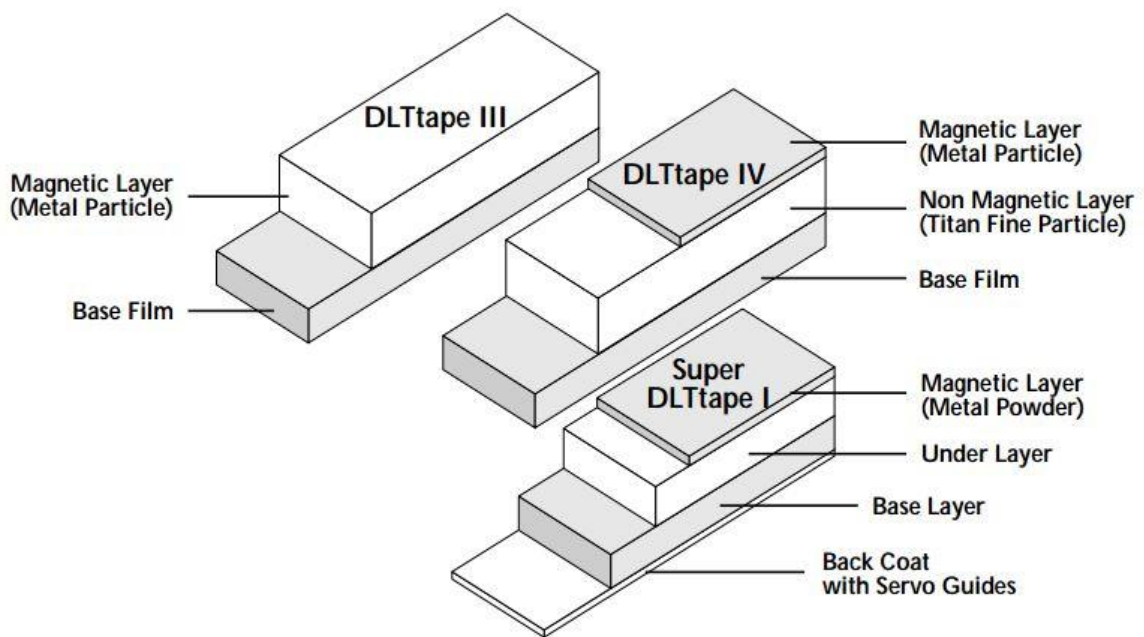
Video: [Az LTO drive belső felépítése + hibaelhárítás. \(HP\)](#)



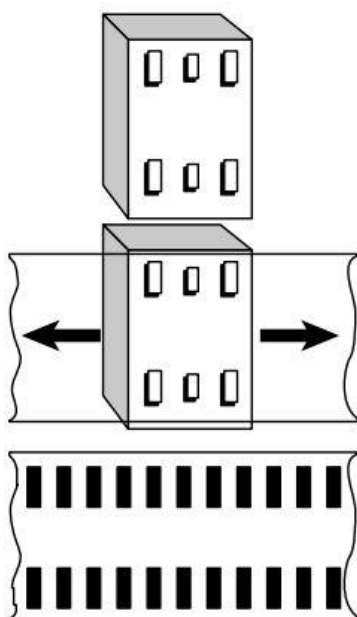
DLT – Digital Linear Tape

LTO – Linear Tape Open

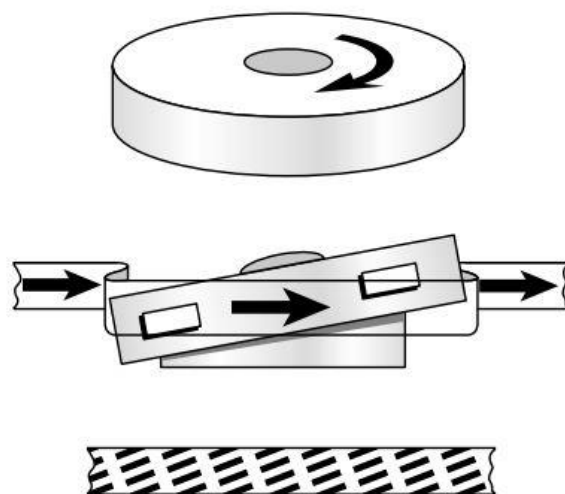
DLT III	1993	10GB	LTO-1	2000	100GB
DLT IV	1999	40GB	LTO-2	2003	200GB
DLT V4	2002	160GB	LTO-3	2005	400GB
SuperDLT I	2002	160GB	LTO-4	2007	800GB
SuperDLT II	2004	300GB	LTO-5	2010	1.5TB
DLT S4	2006	800GB	LTO-6	2012	2.5TB
			LTO-7	2015	6.0TB

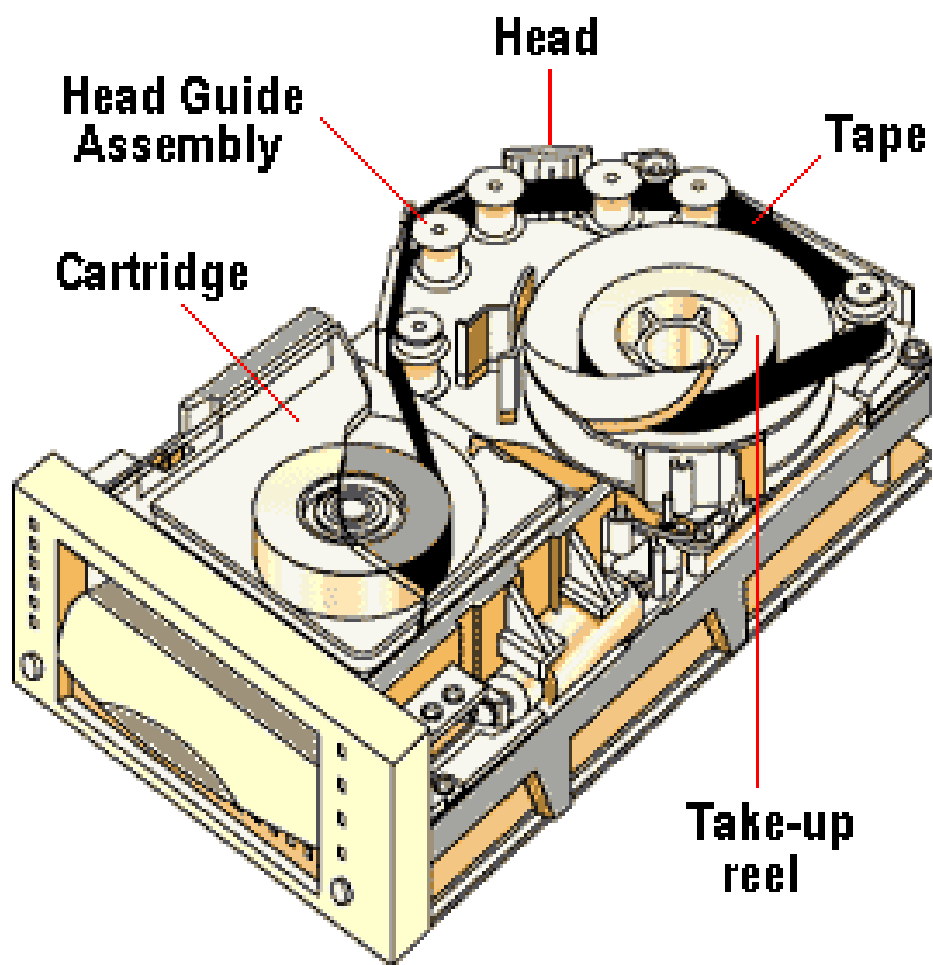
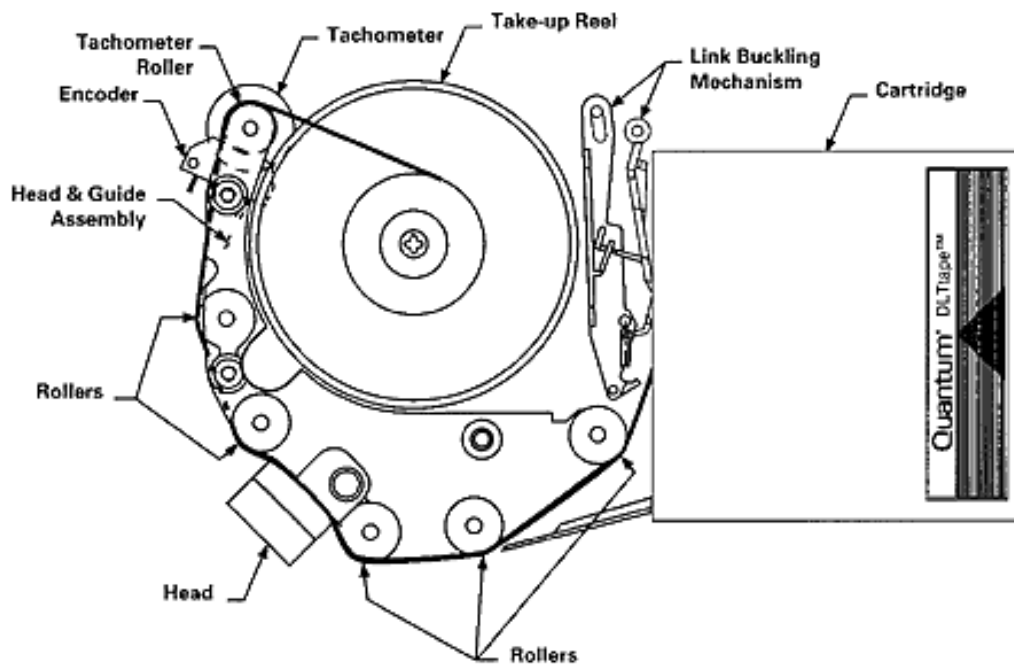


DLTtape drive recording



Helical scan recording





Hatodik fejezet

1. Floppy disk és drive

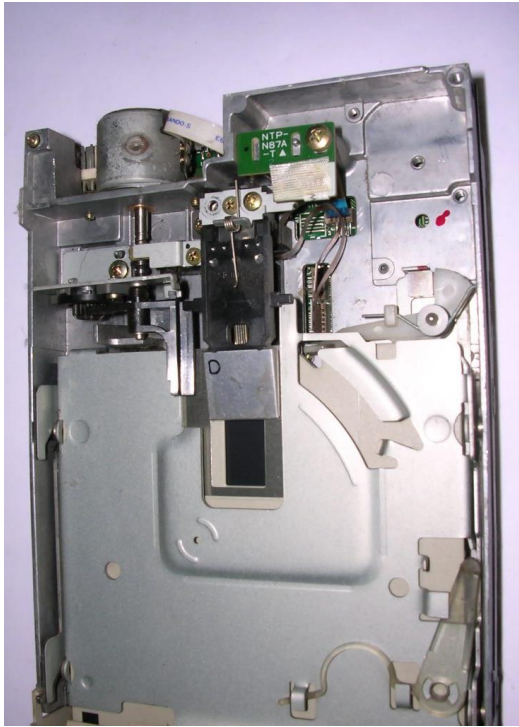
Az első 8"-os puhatokos „Floppy”, azaz hajlékony lemezt az IBM fejlesztette ki a 1960-as évek második felében. Jánosi Marcell (BRG, 1974) alkotta meg az első merevtokos 3.0"-os hajlékonylemezt, de a szabadalom levédésére sem pénz sem hajlandóság nem volt. A Shugart rendszerű 5.25"-es puhatokos lemez 1976-ban, a SONY rendszerű 3.5"-os merevtokos lemez 1982-ben jelent meg a piacon.



A számítástechnikában (anno) használt floppy lemezek (8", 5.25", 3.5"):



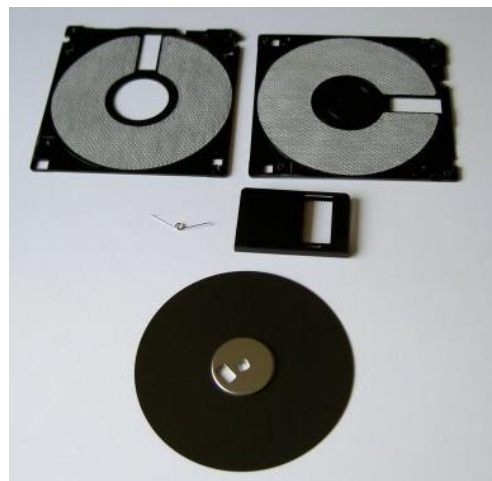
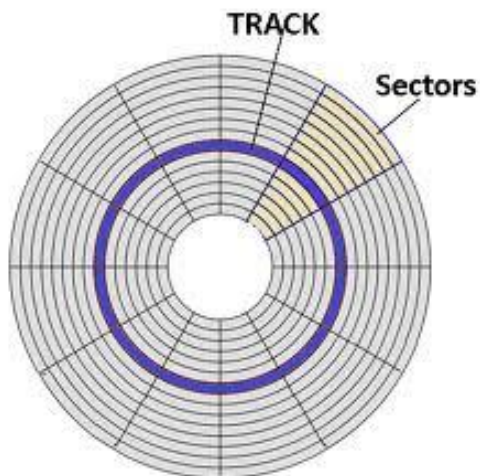
Video: [A 3.5" Floppy Disk Drive működése](#)



DS/DD "Double Density"
"Low Density" 720K
No hole at bottom right



DS/HD "High Density"
1.44 MB
Note hole at bottom right



2. ZIP Drive

100MB kapacitású, 1994-ben jelent meg.

Bernoulli elven működik, a lemez 3.000-et fordul percenként, a fej a forgás következtében létrejövő légpárnán – a lemeztől kb. 1µm-re – helyezkedik el. Így a fej nem ér hozzá a lemezhez. (létezik 250MB és 750MB verziója is, de mire ezek megjelentek, addigra a PenDrive is...)

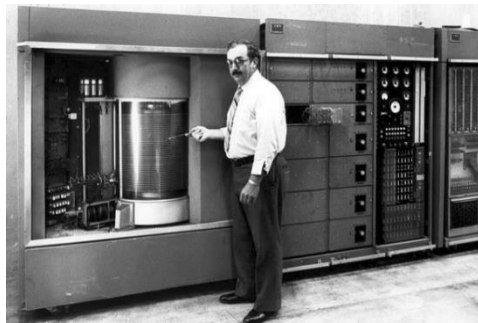


3. HDD

Rövid áttekintés a merevlemez alapú adattárolási technika fejlődéséről:



5MB merevlemez háttértár IBM RAMAC 1956



28MB IBM 1301 1961



2.6MB IBM 1311 1962



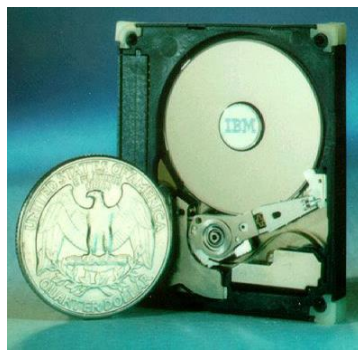
30MB IBM 3340 1973



2.5GB IBM 3380 1980



5MB ST506 1980



170MB CF-II size 1999

Purpose built for applications 容量重視のアプリケーション向けに開発

Capacity Enterprise

6TB

ULTRASTAR® 7K6000

7th generation, legendary reliability, last in-air capacity enterprise

High Scale Cloud

8TB

ULTRASTAR® He8

2nd generation, new mainstream capacity with breakthrough TCO

Online Cool

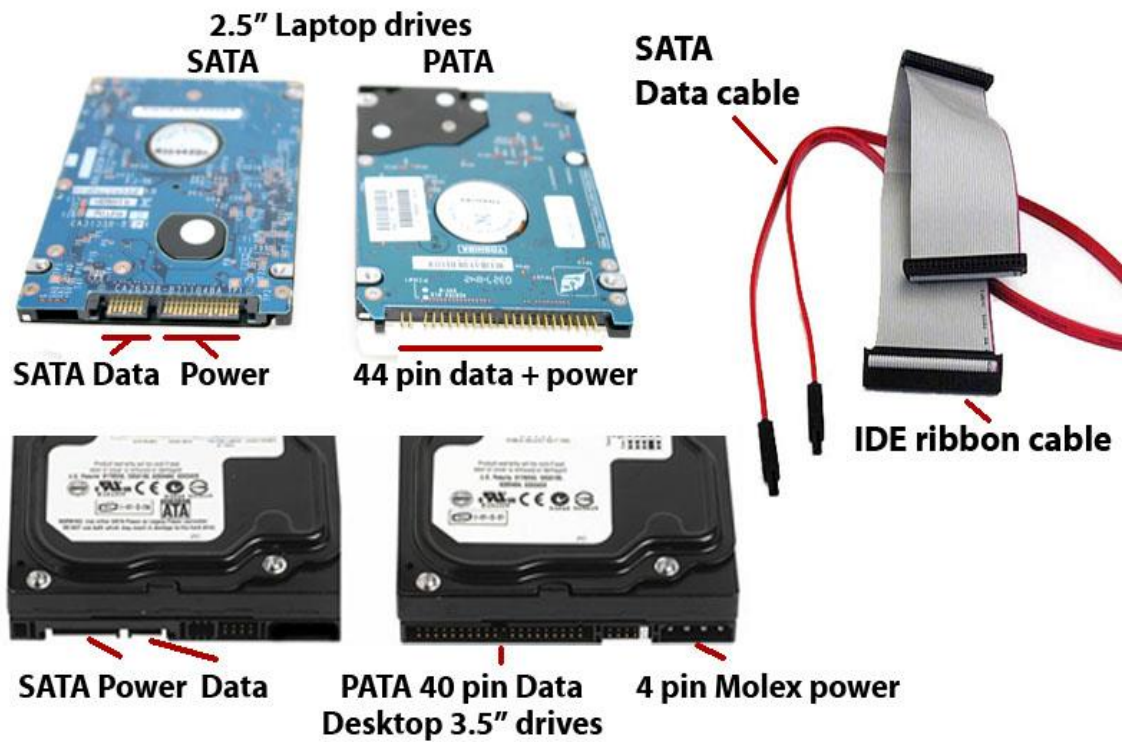
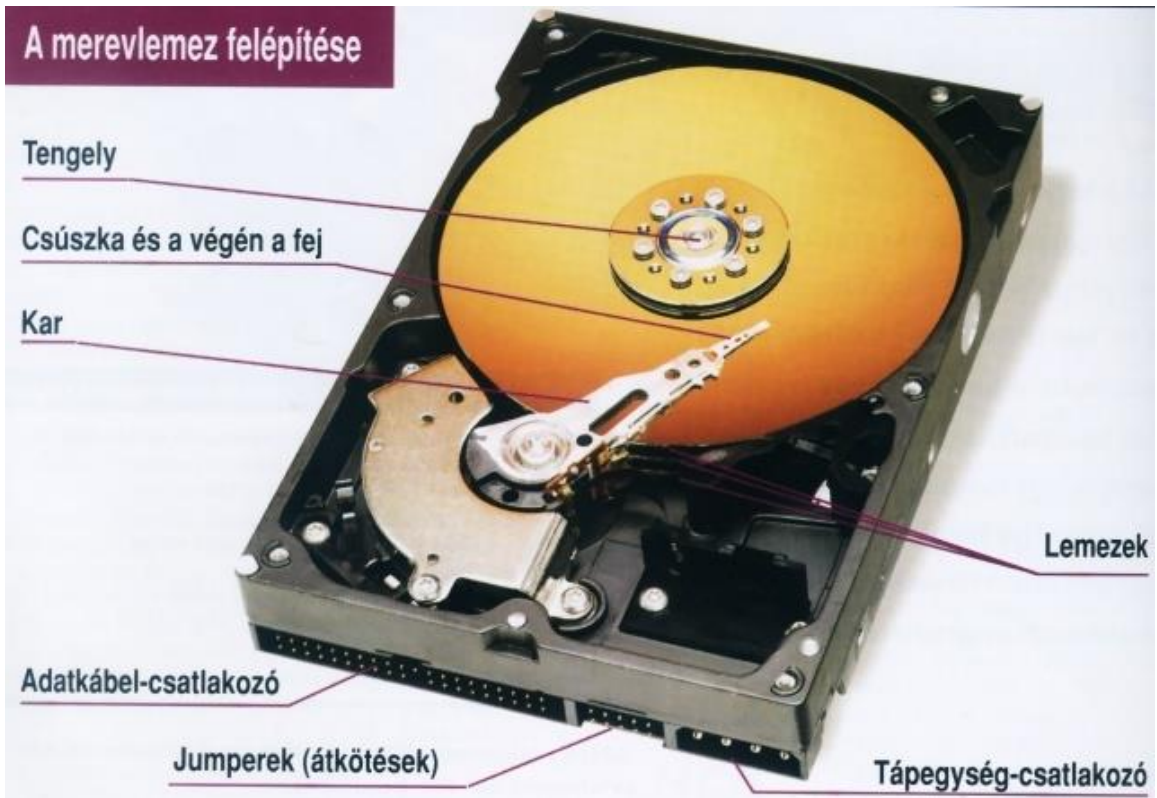
10TB

WORLD'S FIRST 10TB HDD

Value and innovation for Active Archive workloads

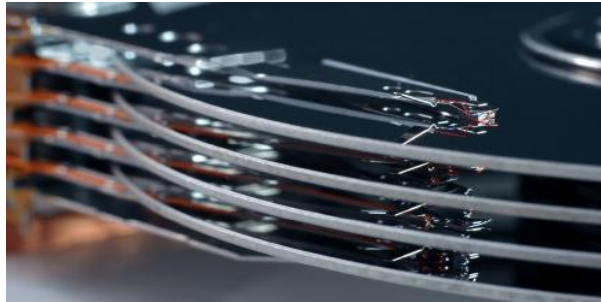
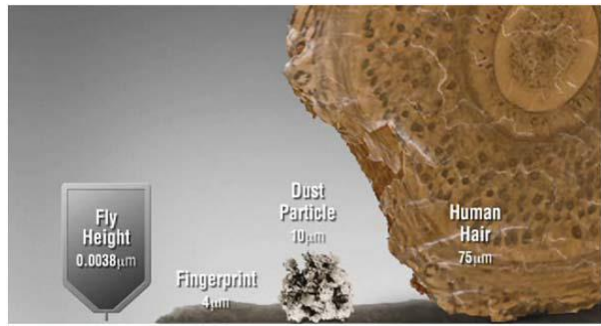
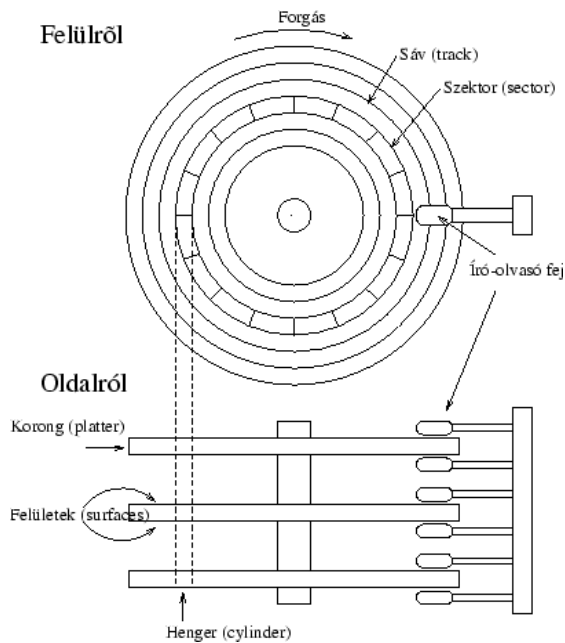
HelioSeal™ HELIUM TECHNOLOGY

10TB Hitachi SMR HelioSeal 2014



Video: [A Hard Disk Drive működése I.](#)

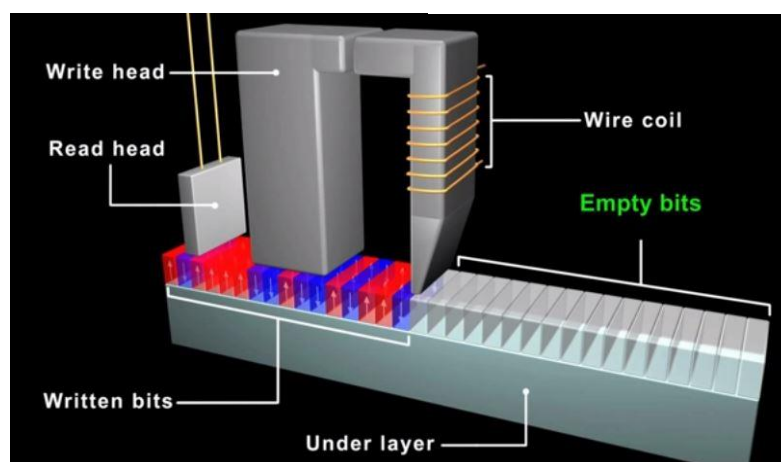
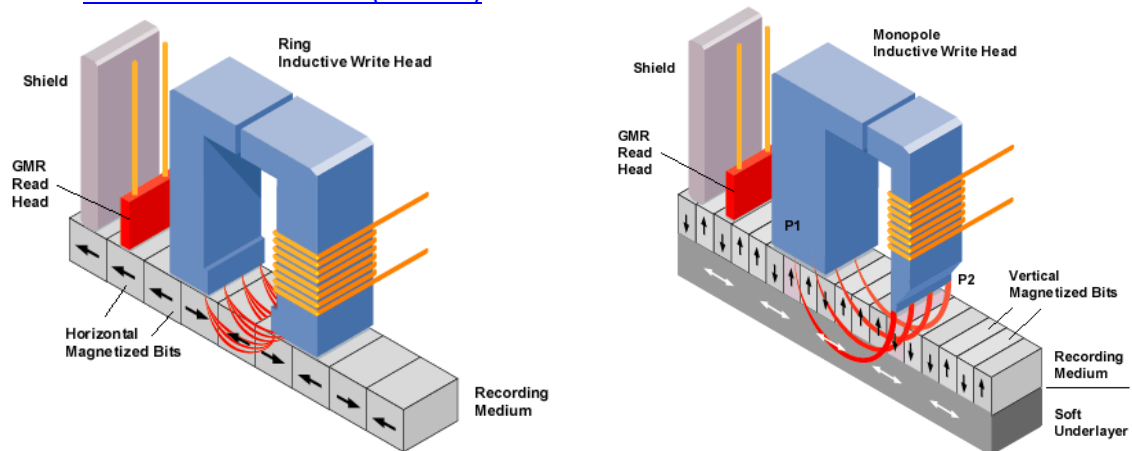
Video: [A Hard Disk Drive működése II.](#)



Az adatsűrűség akár tízszeresére is növelhető, ha a horizontális (longitudinális) mágneses információátvitel helyett vertikális (transzverzális) mágneses információátvitelt használunk. Ez a technológia az olvasást hagyományos geometriájú olvasófejjel olvassa, de az íráshoz viszont szimmetrikus írófej helyett aszimmetrikus írófejet használ. A vertikális adattárolás technológiájának a kifejlesztése Hitachi nevéhez fűződik.

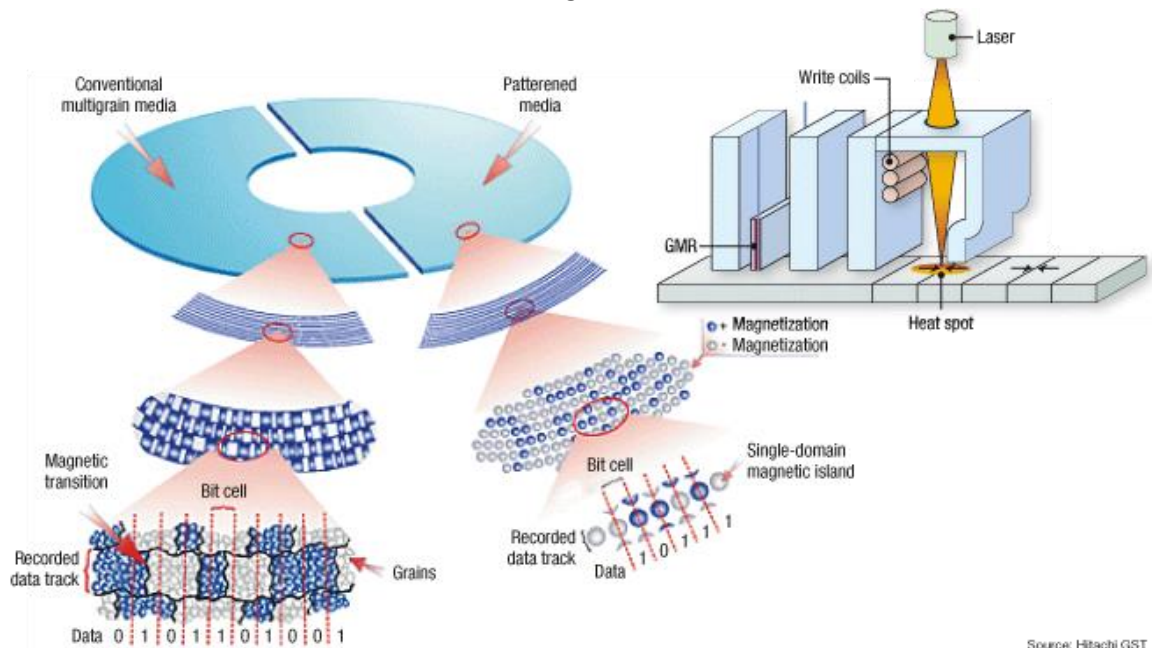
Video: [Vertikális adattárolás I. \(Hitachi\)](#)

Video: [Vertikális adattárolás II. \(Toshiba\)](#)

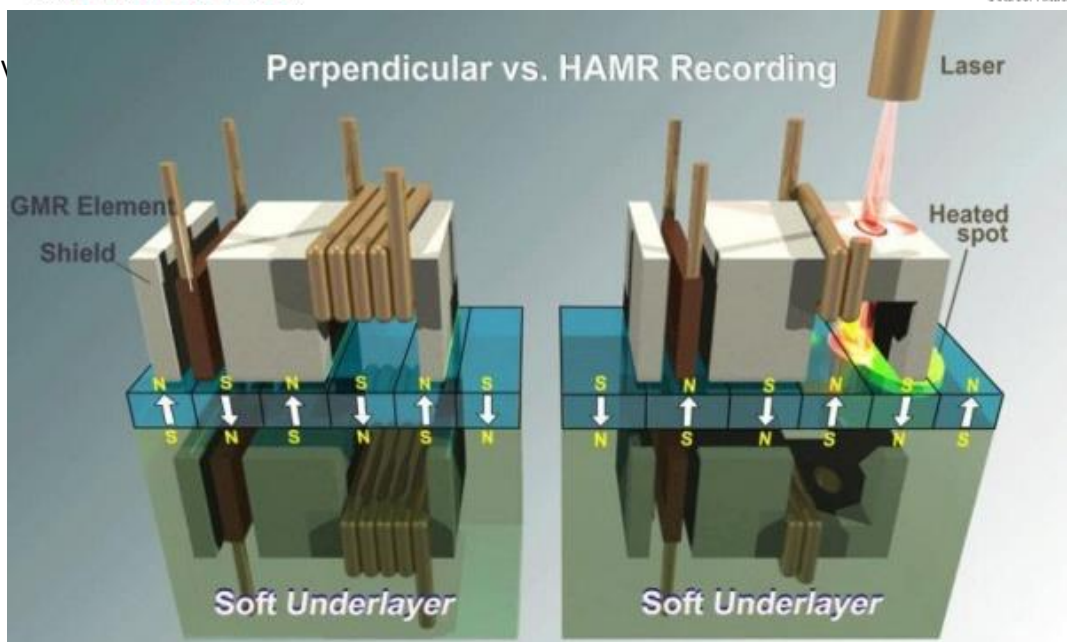


Az adatsűrűség még az eddigieknél is tovább – egy 3.5" HDD esetében akár 60TB-ig is – növelhető a HAMR (Heat-assisted Magnetic Recording / Hővel Támogatott Mágneses Adatrögzítés) technológia segítségével. Magát az effektust, miszerint hő hatására néhány mágneses tulajdonság megváltozik – például a koercivitás, azaz az átmágnesezéshez szükséges mágneses tér erősségének az értéke jelentősen csökken – az úgynevezett magneto-optikai drive rendszer már az 1980-as években is felhasználta, de természetesen még nem a vertikális adattárolással kombinálva. Ugyancsak ezt a megoldást használta az 1992-ben megjelent Sony MiniDisc is. Mai értelemben vett HDD-ben először a Fujitsu mutatta be a HAMR technológiát 2006-ban.

A lenti ábrákon az írófej aránytalanul széles, a valóságban a HAMR technológiával a mágneses információ közelítőleg kör alakban tárolódik. Az írófej geometriai szempontból is a fókuszált lézer által felmelegített lemezterületet kezeli egy tárolási egységként. A bal alsó ábrán az írófej kisebb menetszáma a kevesebb árammal történő átmágnesezésre utal.

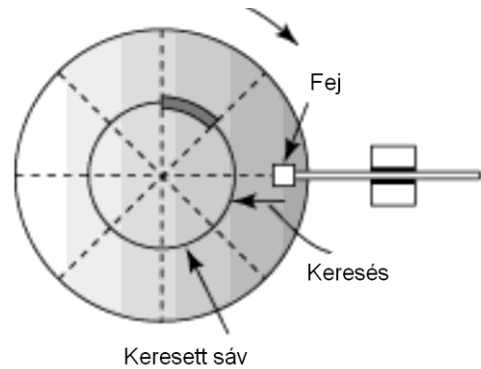


Source: Hitachi GST

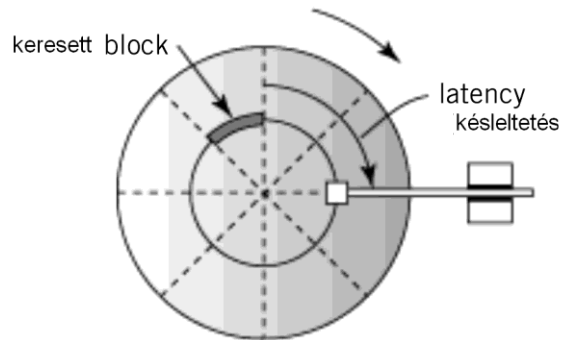


Video: [HAMR \(TDK\)](#) [A TDK HDD-t nem gyárt, de pl. az egyik legnagyobb HDD fej beszállító.]

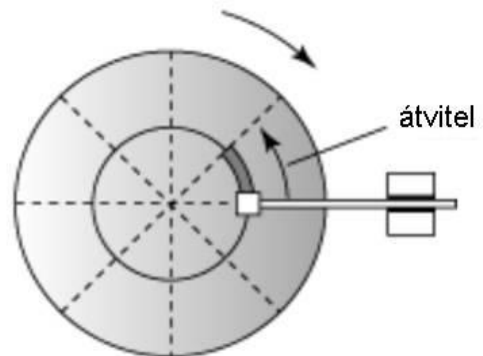
- **Átlagos fejmozgatási idő:**
A fej egyik sávról a másikra történő mozgásához szükséges idő átlaga. A fej mozgatása más algoritmus szerint valósul meg attól függően, hogy csak néhány, vagy esetleg több száz sávot kell átlépni. A fejet fel kell gyorsítani, le kell fékezni, végül pedig pontosan pozícionálni kell.



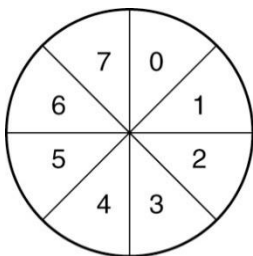
- **Latency / az elfordulás miatti késleltetés:**
Az az átlagos idő, amely ahhoz szükséges, hogy a lemez a keresett szektor kezdetéhez forduljon.



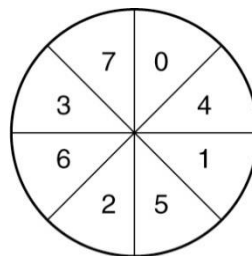
- **Átviteli idő:**
Az az idő, amely ahhoz szükséges, hogy egy adatblokkot a lemezkezelő a pufferbe juttasson. Az adatok először a HDD pufferébe kerülnek, majd onnan az operációs rendszerhez. Az átvitel remélt gyorsításának érdekében, ha puffer kapacitása lehetővé teszi, a fej a kívánt szektor minél több szomszédját is beolvassa, mert az esetek jelentős százalékában ezek tartalmára valóban szükség is lesz.



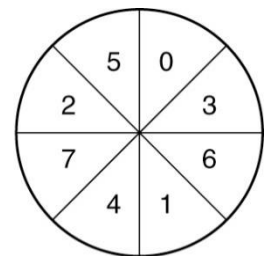
- **Disc Interleaving:**
A gyorsabb adateléréshez optimalizált (logikai) szektor sorrend. A szektorok logikai elrendezése lehetővé teszi, hogy a legkisebb Latency idővel lehessen elérni a beolvasandó adatblokkot.



(a)

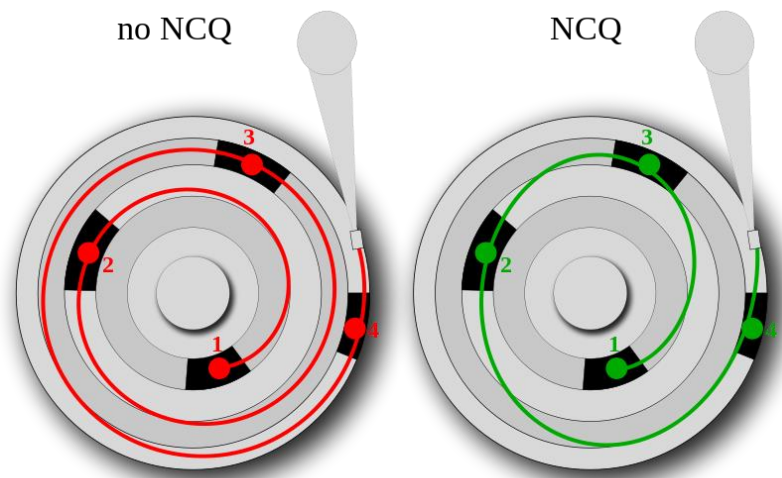


(b)

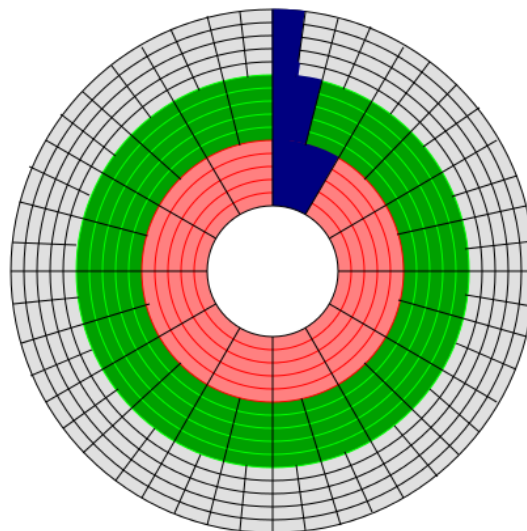


(c)

NCQ (Native Command Queuing) rendszer lehetővé teszi, hogy a HDD saját – legkevesebb fejmozgásra és időre optimalizált – egyedi sorrendben olvassa be egymás után a szektorokat. A pufferből a szektorok adatai más a valós sorrendben kerülnek az operációs rendszerhez.



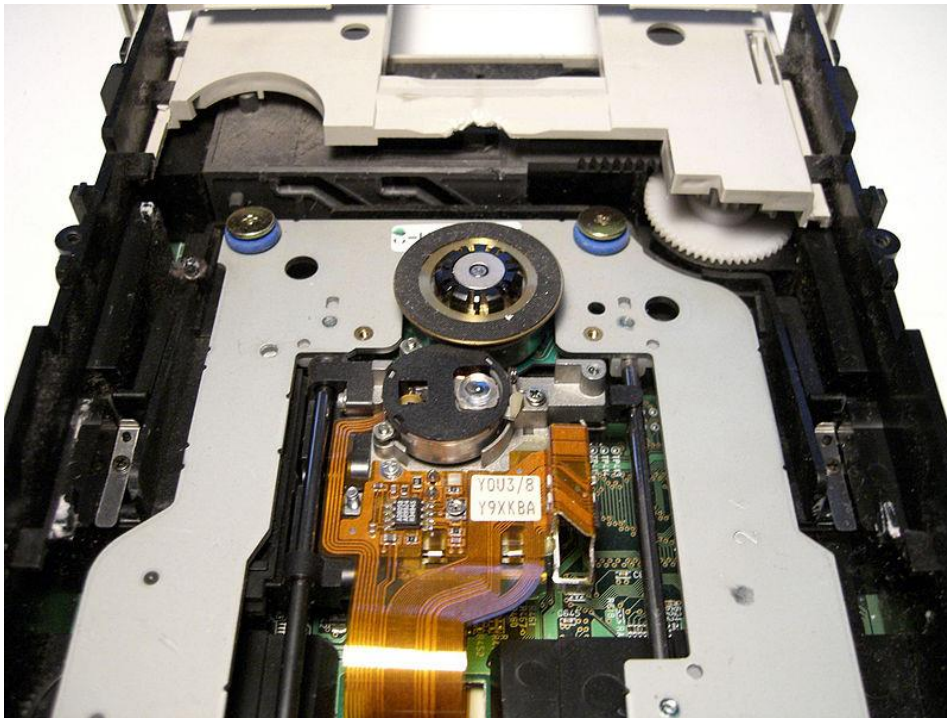
A merevlemez fizikai mérete, maximális sugara – az állandó fordulatszám miatt (állandó szögsebesség) – meghatározza a minimális illetve maximális kerületi sebességet a lemez legbelső illetve legkülső sávján. A kör alakú geometria miatt a külső sávokon a szektorok a nagyobb területet foglalnak el, mint a belső sávokon. A terület optimális kihasználása (azaz a folyamatosan a maximális adatsűrűség elérése) érdekében a HDD sávjait a ZBR (Zone Bit Recording) rendszer zónákra bontja. Egy napjainkban használatos nagykapacitású (2TB, 3TB) HDD akár 30-nál több zónát is tartalmazhat.



Ez egyben azt is jelenti, hogy a például egy teljes körülfordulás (azaz egységnyi idő) alatt a külső sávokból több információ olvasható ki, mint a belső sávokból. Így amennyiben minden egyéb paramétere megegyezik, egy 3.5"-os HDD lemez átlagos adatátviteli teljesítménye nagyobb lehet, mint egy 2.5"-os HDD lemezé.

Érdekes információ, hogy ez a technológiát használta a Commodore 64 (a nyolcvanas évek népszerű otthoni számítógépe) floppy egysége illetve a korai Apple Macintosh floppy-k is. Az általunk korábban tárgyalt IBM FDD-k ezt a rendszert nem alkalmazták. Érdemi előnyt csak sok sáv és sok zóna esetén nyújt ez a rendszer.

4. ODD, Optikai lemezek, CD, DVD, Blu-Ray, DVD-RAM.

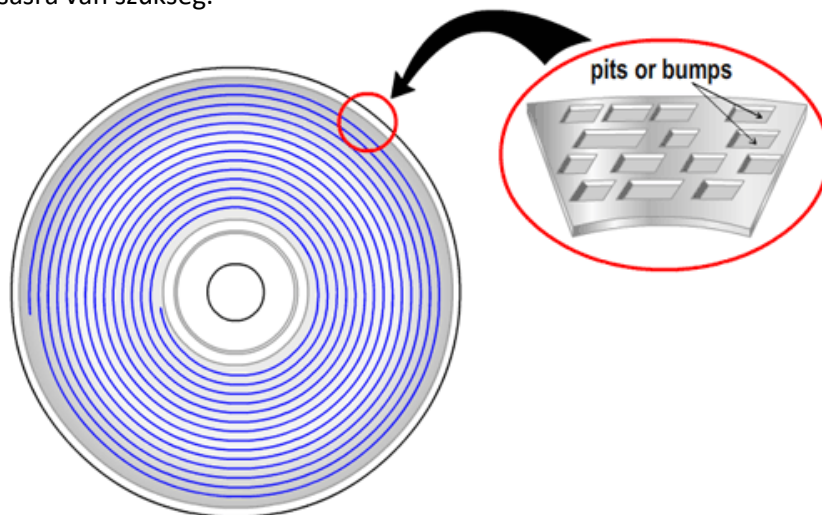


Video: [Az Optical Disk Drive működése](#)

Video: [A CD működése](#)

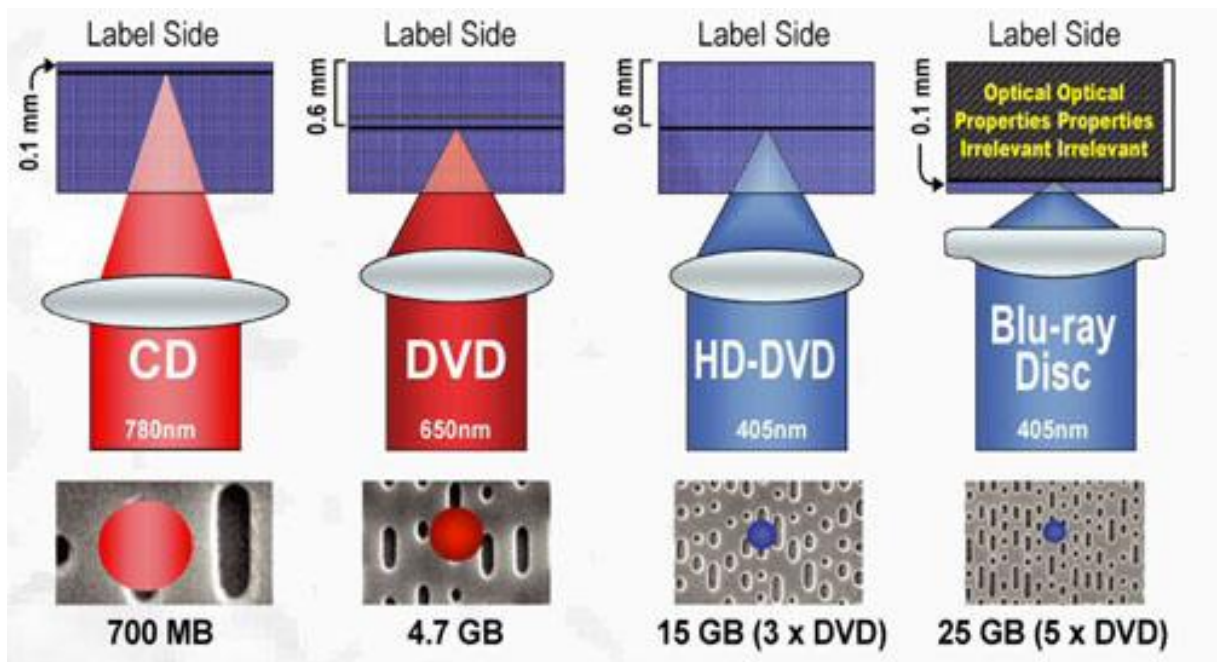
Video: [A Blu-Ray lemez felépítése \(TDK\)](#)

A spirálisan tárolt adatok (úgynevezett pit-ek) olvasásához állandó és több irányú kompenzált fej pozícionálásra van szükség. Az aktuális sáv követéséhez páratlan számú, a szomszédos sorokat is pásztázó olvasásra van szükség.



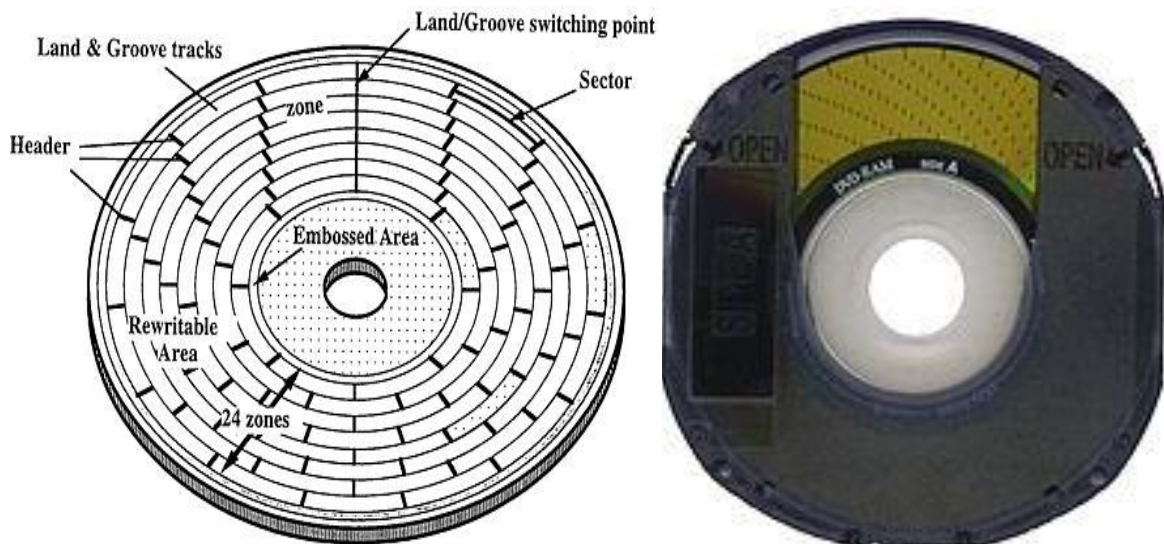
A DVD-R (1997) és a DVD+R (2002) formátumok közti különbség több oldalról is megközelíthető. A legfontosabb különbség a felhasználó által rögzített (illetve rögzítendő) adatokon kívüli, a sávok követését segítő extra információk lemezen történő elhelyezkedésében van. A DVD-R esetében ezen információk külön, míg a DVD+R esetében az adatokkal együtt helyezkednek el. Ebből következik pl. hogy a DVD+R elviseli az írás megszakadását, míg a DVD-R nem. Mivel a DVD+R eleve adattárolásra, a DVD-R pedig video anyagok tárolására szánt technológia, ezért a DVD+R a nagy filmstúdiók másolásvédelmi megoldásait (CSS for DVD Authoring) nem is támogatja.

Video: [A DVD-R és a DVD+R összehasonlítása](#)

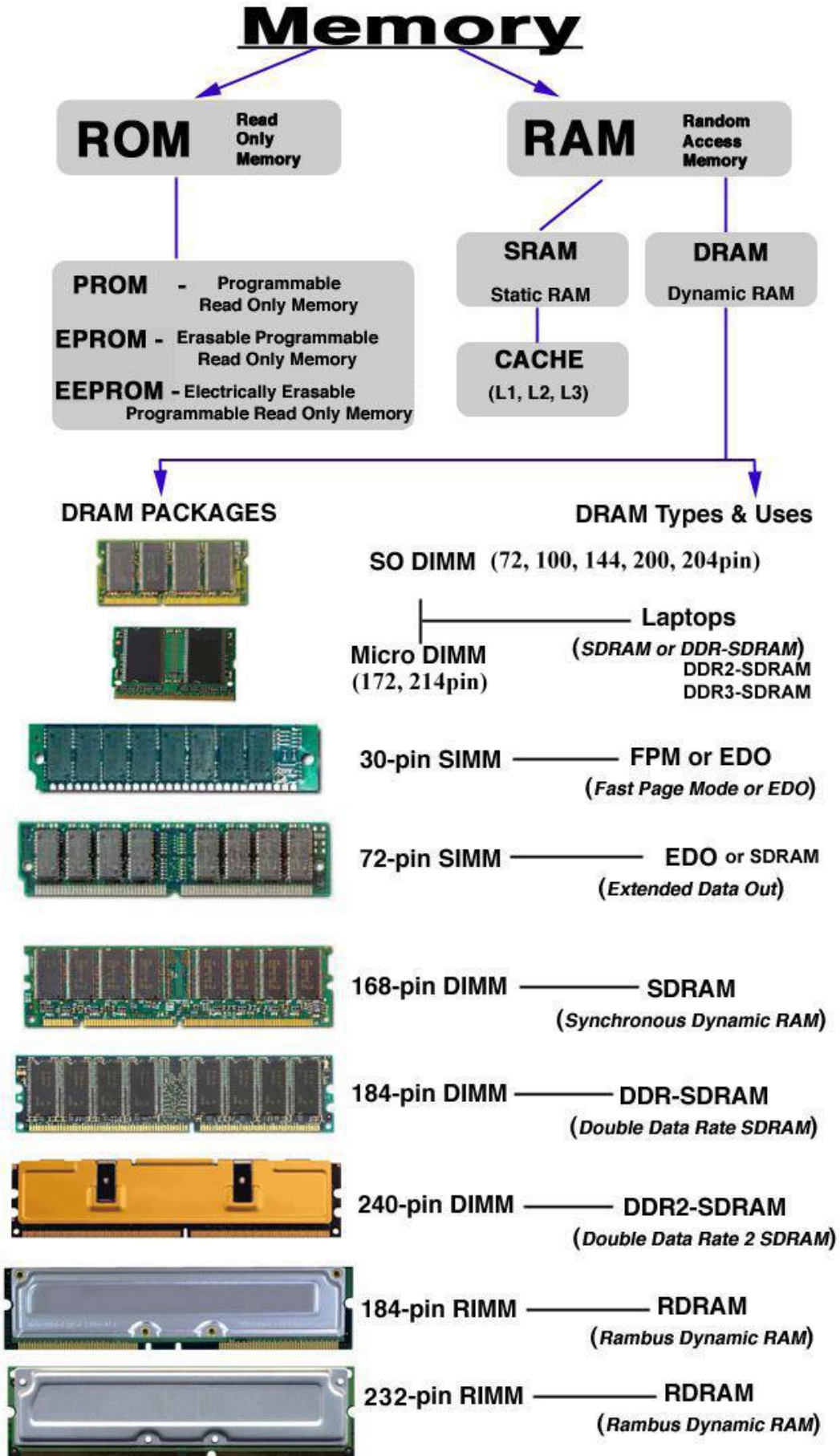


5. DVD-RAM

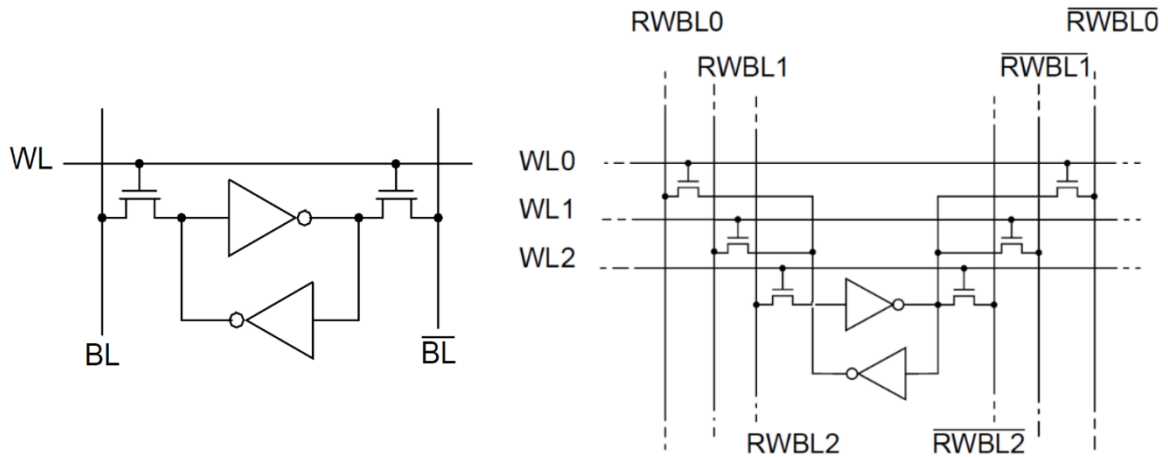
Az adatok tárolás nem spirálban, hanem koncentrikus körökben, azaz sávokban és szektorokban történik, így geometriája a HDD rendszerrel mutat hasonlóságot. A többi optikai tárolástól eltérően a ZBR sektorszerkezetet is használja, fizikailag előre kijelölt lemezterületeket, azaz a Hard Sector megoldást alkalmazva. Adatbiztonsága és újraírhatósága a többi optikai tárolás többszöröse. A specifikáció szerint legalább 30 évig megőrzi az adatokat, és legalább 100.000 alkalommal újraírható.



6. RAM modulok



SRAM



A statikus RAM egy cellájának belső felépítése (ami egy bit tárolására képes) – logikai áramkörként tekintve – két egymással szembe kötött inverterből, és két MOSFET-ből áll. Ebből következik, hogy egyik inverter bemenete a felhasználó által tárolni kívánt bitet, míg a másik inverter bemenete a felhasználó által tárolni kívánt bitnek az inverzét tárolja. Ez az áramkör mindaddig képes megőrizni a felhasználó által tárolni kívánt bit értékét, amíg tápfeszültséggel ellátjuk. A cella kiolvasási és írási művelete két tranzisztor (MOSFET) segítségével végezhető el.

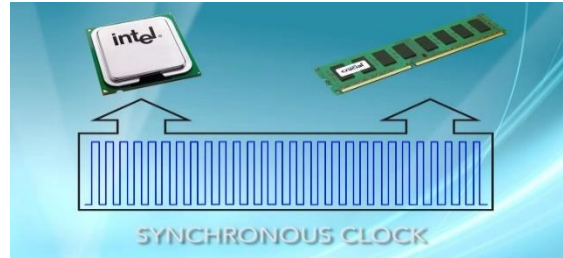
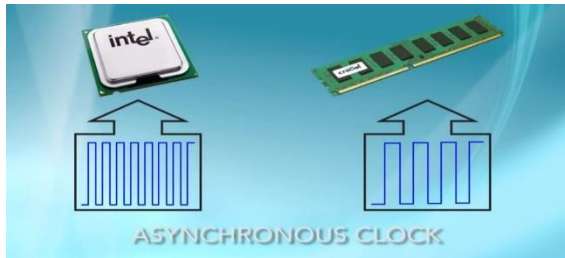
A modern processzorok L2 illetve L3 cache memóriája, azaz gyorsítótára a SRAM elsődleges felhasználási területe. Többmagos CPU-k esetén a magok közös gyorsítótára is egyszerűen kialakítható. További tranzisztorok (MOSFET-ek) segítségével olyan tárolót hozhatunk létre, amelynek tartalmához akár egyszerre több (a fenti ábra szerint három) CPU mag is hozzáférhet.

WL: Word Line

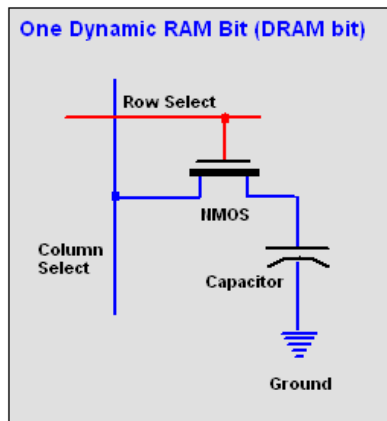
BL: Bit Line

Video: [Az SRAM működése](#)

DRAM



A dinamikus memóriák minden egyes cellája egy kondenzátort és egy tranzisztort (az esetek 99,9%-ában MOSFET-et) tartalmaz. Az információt a kondenzátor tárolja, – a CMOS



technológiában megismert logikai szinteknek megfelelő töltöttségi szintek alapján – megkülönböztetve a "0" illetve az "1" értékű tartalmat. Mivel a kondenzátor a töltöttségét csak nagyon rövid ideig képes megőrizni, a tartalmat ciklikusan frissíteni kell.

A frissítés elve legegyszerűbben egy olyan lyukas pohárral modellezhető, amiben folyamatosan törekszünk a kívánt folyadékszint megtartására.

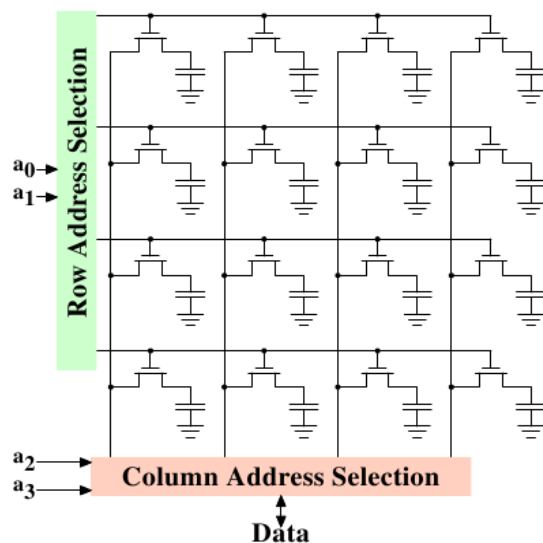
A címzés természetesen nem bitenként történik erre utal az ábrán a sor illetve az oszlop kiválasztásához tartozó vezeték. A kapcsolódó paraméterek a RAS [Row Address Strobe / Sor

Címzési Idő] és a CAS [Column Address Strobe / Oszlop Címzési Idő], valamint a CAS Latency [CL], azaz az a CAS-hoz tartozó optimális késleltetési idő. A Latency elve itt is ugyanaz, mint ahogyan azt a HDD esetében tárgyaltuk. További két paraméter az RCD és az RP. Az RCD [RAS to CAS Delay / a RAS kívánt késedelmé a CAS-hoz képest], ami azt a késleltetést határozza meg, hogy hány buszciklusnak kell eltelnie a keresett adat megcímzése és az adatok fizikai mozgatása között. Az RP [RAS Precharge / Sorelőtöltési idő] azt mutatja meg, hogy hány buszciklusnak kell eltelnie ahhoz, hogy az aktuálisan nyitott sor lezárásra kerüljön, és a következő sor megnyithatóvá váljon. Meg kell említeni továbbá az RC [Row Refresh Cycle Time / Sorfrissítési Ciklusidő] paramétert is.

Négy paraméter esetén a paraméterek sorrendje ez: CL-RCD-RP-RAS. PI.: 2-2-3-4.

Öt paraméter esetén pedig a sorrend a következő: CL-RCD-RP-RAS-RC. PI.: 7-7-7-24-26.

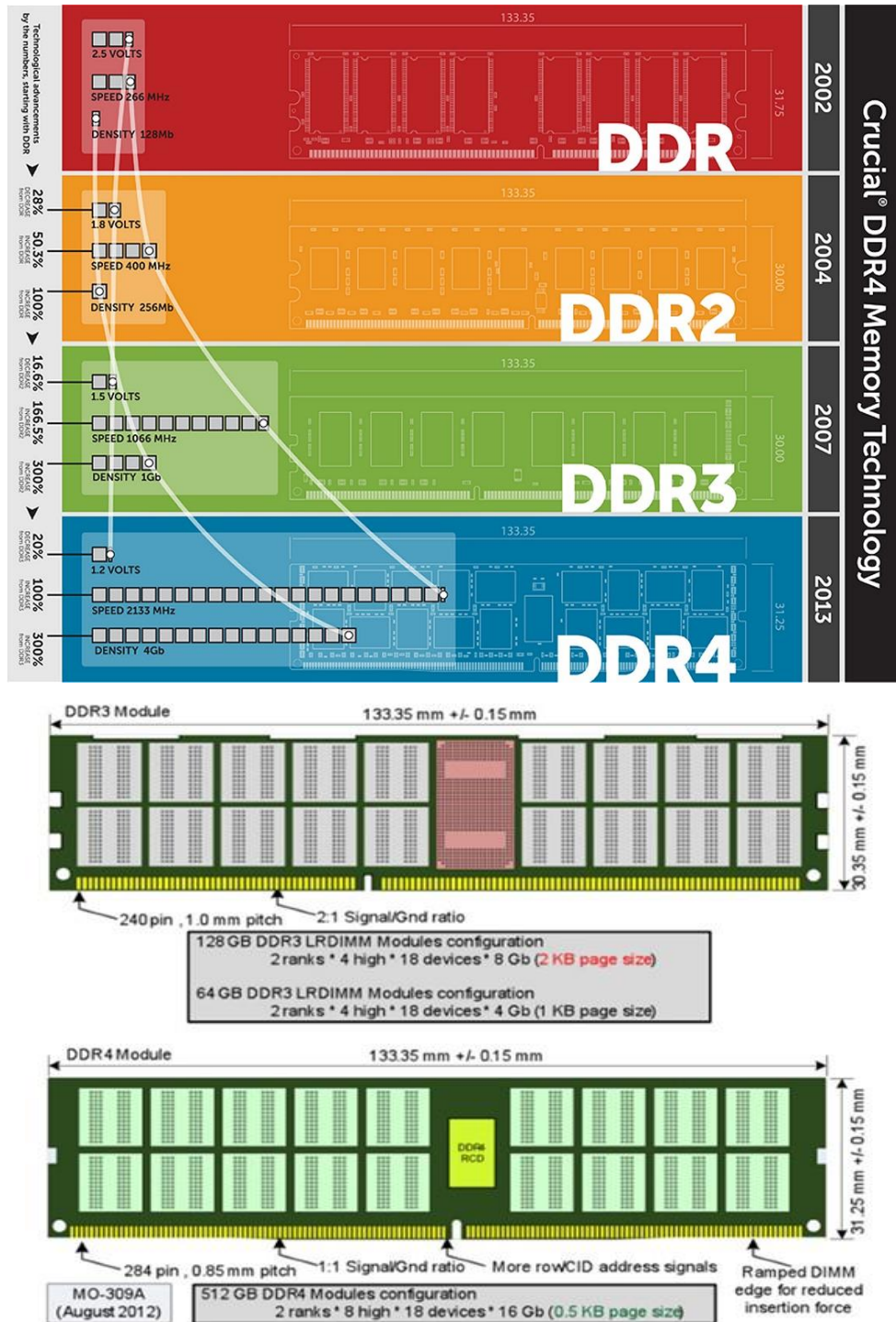
Ezeket a paramétereket a felhasználó a BIOS-ban állíthatja be (de leginkább csak el...☺).



A dinamikus memória jellegéből fakad, hogy minden kiolvasás után újra kell írni az adatokat (az adatfrissítés szükséges a töltés megtartásához). Az olvasási ciklusban az algoritmus a kiválasztott cella sorában bekapcsolja a MOSFET-eket, így a sor kondenzátorait az oszlopválasztó vezetékekhez kapcsolja. Az írás is a sor aktiválásával indul, az írandó adatokat az oszlopválasztó vezetékekhez kell csatlakoztatni, így a kondenzátorok a kívánt értékeket el tudják tárolni. Egyetlen cella írása során tehát az egész sor kiolvasásra kerül, majd mivel (legalább) egy érték megváltozik, az egész sor az új tartalommal visszaíródik.

Video: [A DRAM működése.](#)

A klasszikus SDRAM (Synchronous Dynamic RAM) memóriák az adatátvitel során a SDR (Single Data Rate) technológiát alkalmazza, azaz órajel ciklusonként egyszeri adatátvitelre képesek. A DDRx SDRAM memóriák ezzel szemben [amint az a nevükből is kiderül] a DDR (Double Data Rate) technológiát alkalmazzák, aminek segítségével órajel ciklusonként kétszeri adatátvitelre képesek, mégpedig az órajel felfutó és lefutó éle mentén is. Gyakorlatilag ugyanezen az elven működtek már évekkel a DDRx memóriák megjelenése előtt a RAMBUS memóriák is (amíg a magas árak miatt ki nem szorultak a piacról).



Főleg a professzionális célokra használt számítógépekben (pl. szerverekben) jellemzőek az ECC (Error-Correcting Code) memóriák, melyek paritásgenerálás segítségével képesek a hibajavításra.

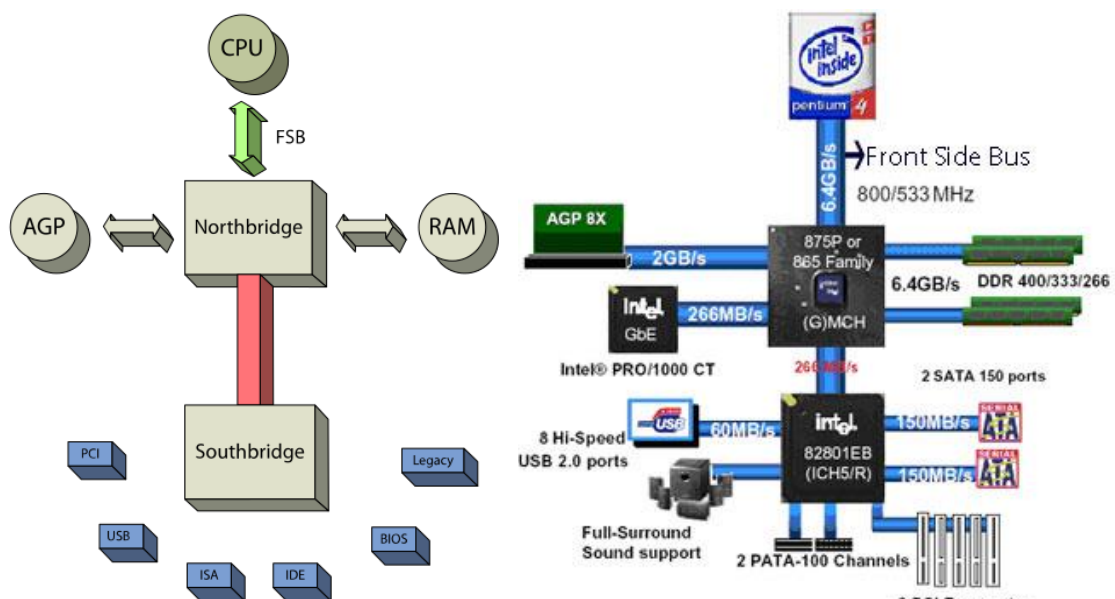
Az operatív táruk ma már gyakorlatilag csak dinamikus memóriából állnak.

RAM → Az egyedileg tokozott DIP-től a SIM, EDO, SD, RAMBUS, DDRx kialakításban.

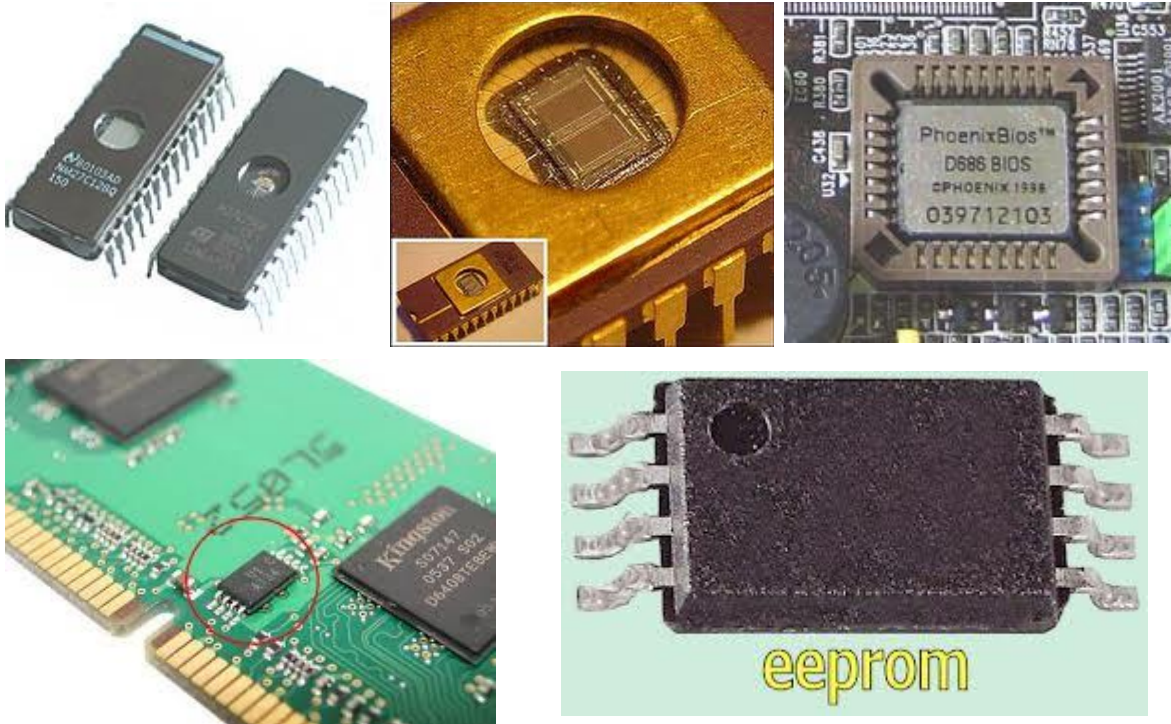
Random Access Memory = Tetszőleges Hozzáférésű Memória

A „random” azaz tetszőleges (tehát ez esetben a jelentés NEM a „véletlenszerű” szó!) kifejezés a „sequential” azaz szekvenciális, kötött sorrendű hozzáférés ellentétéként értelmezendő!

Megnevezés	Real Clock [MHz] Valós órajel	Front Side Bus [MHz] Főoldali busz	Bandwidth [MB/s] Sávszélesség
DDR-200 PC-1600	100 MHz	200 MHz	1600 MB/s
DDR-266 PC-2100	133 MHz	266 MHz	2100 MB/s
DDR-333 PC-2700	166 MHz	333 MHz	2700 MB/s
DDR2-400 PC2-3200	200 MHz	400 MHz	3200 MB/s
DDR2-533 PC2-4200	266 MHz	533 MHz	4200 MB/s
DDR2-667 PC2-5400	333 MHz	667 MHz	5400 MB/s
DDR2-800 PC2-6400	400 MHz	800 MHz	6400 MB/s
DDR2-1066 PC2-8500	533 MHz	1066 MHz	8500 MB/s
DDR2-1333 PC2-10600	667 MHz	1333 MHz	10600 MB/s
DDR3-800 PC3-6400	400 MHz	800 MHz	6400 MB/s
DDR3-1066 PC3-8500	533 MHz	1066 MHz	8500 MB/s
DDR3-1333 PC3-10600	667 MHz	1333 MHz	10600 MB/s
DDR3-1600 PC3-12800	800 MHz	1600 MHz	12800 MB/s
DDR3-1866 PC3-14500	933 MHz	1866 MHz	14500 MB/s
DDR3-2000 PC3-16000	1000 MHz	2000 MHz	16000 MB/s
DDR3-2133 PC3-17000	1066 MHz	2133 MHz	17000 MB/s
DDR4-2133 PC4-17000	1066 MHz	2133 MHz	17000 MB/s
DDR4-2400 PC4-19200	1200 MHz	2400 MHz	19200 MB/s
DDR4-2800 PC4-22400	1400 MHz	2800 MHz	22400 MB/s
DDR4-3000 PC4-24000	1500 MHz	3000 MHz	24000 MB/s
DDR4-3333 PC4-26600	1667 MHz	3333 MHz	26600 MB/s
DDR4-3600 PC4-28800	1800 MHz	3600 MHz	28800 MB/s



7. ROM, PROM, EPROM, EEPROM



8. A Flash alapú tárolás

A Flash memória tulajdonképpen az EEPROM egy speciális, továbbfejlesztett változata, mely nem csak teljes egészében, hanem blokkonként is törölhető, illetve újraírható. Az információ tároláshoz ennek a technológiának sincsen szüksége energiaellátásra, és a tároló mozgó alkatrészeket sem tartalmaz. Jellemző tulajdonsága a gyors olvasás hozzáférési idő. A gyors – azaz jellemzően 50ns – ez esetben azt jelenti, hogy a HDD sebességénél gyorsabb, de a dinamikus RAM sebességénél lassabb.



CF Compact Flash, az elsőt 1994-ben mutatta be a ScanDisk

SM SmartMedia, a Toshiba kártyaformátuma, 1995-ben mutatták be

MS MemoryStick, a Sony kártyaformátuma, 1998-ben mutatták be

SD Secure Digital, az SDA (Secure Digital Association kártyaformátuma, 1999-ben mutatták be
Jelenleg ez a legelterjedtebb kártya formátum.

- o generációi: SD, SD HC (High Capacity) 32GB-ig, SD XC (eXtended Capacity) 32GB felett

- o A „Class” besorolás maximális sebességei:

- o Class2: 2MB/s, Class4: 4MB/s, Class6: 6 MB/s, Class10/UHS-I: 10 MB/s, UHS-II: 30 MB/s

- o Három tokozási módja a Normál, a Mini és a Micro kártyaformátum

xD xD-Picture Card, az Olympus és a Fuji közös fejlesztése, 2002-ben mutatták be, érdekessége, hogy a vezérlőáramkört nem a kártya, hanem a fogadó eszköz tartalmazza.

A PenDrive (USB Kulcs) olyan hordozható Flash alapú tárolóeszköz, amely USB porton keresztül csatlakoztatható a számítógéphez. A legváltozatosabb kialakításokban kerül forgalomba.



Napjaink egyik népszerű technológiája, sok helyen találkozhatunk Flash memóriákkal – például az USB Pen drive-ok és a memóriakártyák is azok, csakúgy, mint a korszerű alaplapok BIOS-át illetve UEFI-jét tároló chip-ek. A lényegi különbség az egyes eszközök vezérlő elektronikájában van.

9. SSD és Hibrid HDD (SSHD)

Technológiai szempontból két típust különböztetünk meg, melyek alapvető tulajdonságai jelentősen eltérőek.

- NOR technológia

A technológia legnagyobb gyártója az Intel.

- párhuzamos adathozzáférés
- közvetlen programfuttatásra alkalmas, Random címzésű, bájtankénti elérésű
- írás: 5 μ s/bájt, Olvasás: 50-100ns/bájt, Törlés: 1s/64kB blokk
- viszonylag drága
- kisebb kapacitású chip-pek

- NAND technológia

A technológia legnagyobb gyártója a Samsung.

Fizikailag kisebb helyigényű, és energiatakarékosabb, mint a NOR technológia.

- soros adathozzáférés
- közvetlen programfuttatásra nem alkalmas, mert csak laponként címezhető
- főleg folyamatos azaz nagyobb adatmennyiségek írására-olvasására alkalmas
- írás: 200 μ s/lap, Olvasás: 10 μ s keresési idő+50ns/bájt, Törlés: 2ms/16kB blokk
- viszonylag olcsó
- nagyobb kapacitású chip-pek

Mindkét technológia a cellánkénti 1 bit tárolásával indult, ami azonban két szempontból is korlátot jelentett. Egyrészt a közvetlenül tárolható adatmennyiség szempontjából, másrészt ezzel összefüggésben az íráskor „kifáradt” azaz elhasználódott cellák adatainak áthelyezésére fenntartott tárolókapacitás szempontjából.

- Az SLC (Single Level Cell) technológia cellánként 1 bit tárolására képes 2 feszültségszinten.
- Az MLC (Multi Level Cell) technológia cellánként 2 bit tárolására képes 4 feszültségszinten.
- A TLC (Triple Level Cell) technológia cellánként 3 bit tárolására képes, 8 feszültségszinten.

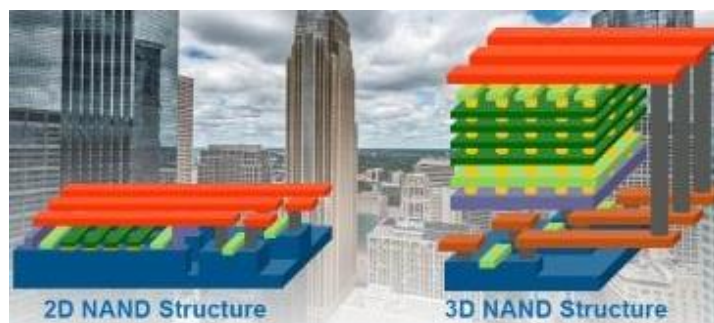
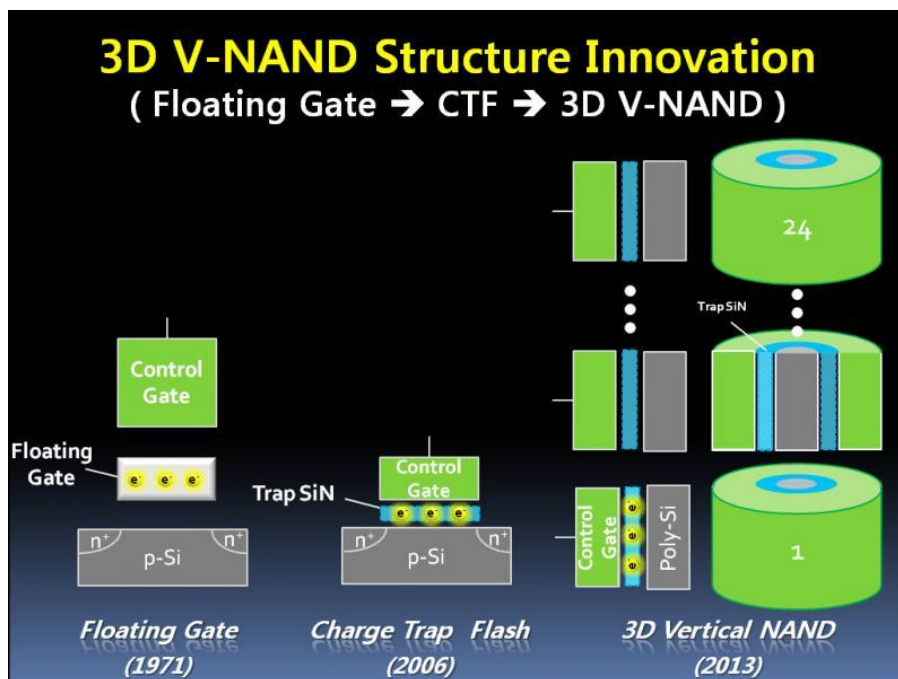
A tárolt bitek száma és a feszültségszintek között 2 hatványait találjuk összefüggésként. Több bit tárolása ugyanakkora helyen nagyobb adatsűrűséget, nagyobb tárkapacitást jelent.

Video: [A VNAND Flash Memória működése \(Charge Trap Flash\)](#)

Video: [A Flash Memória működése \(Floating Gate Transistor\)](#)

Video: [Az SLC, MLC és TLC összehasonlítása](#)

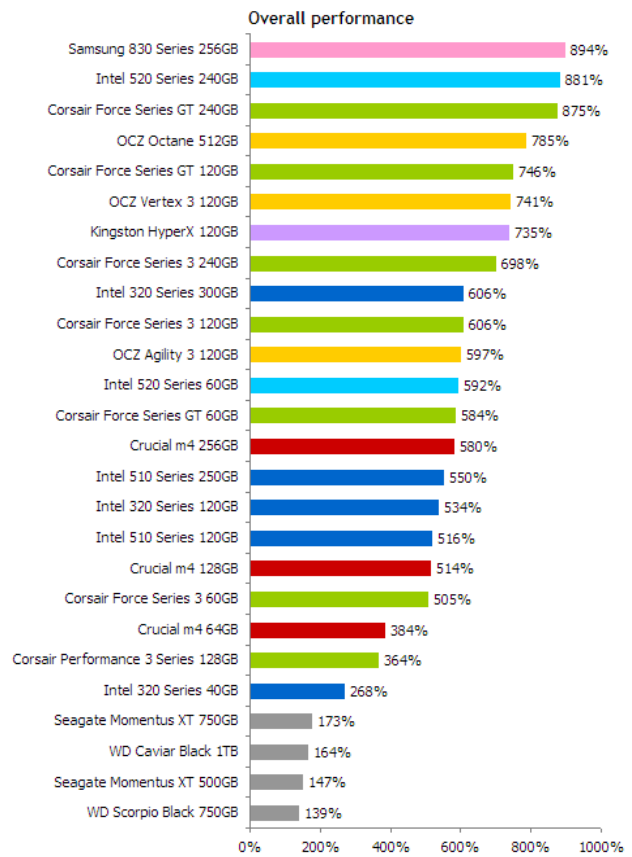
[Csak zárójelben: Jellegét tekintve (mint szilárdtest) a ferritgyűrűs tároló is egyfajta SSD volt ☺]



Az elemi tároló cella első lépésben a Floating Gate Tranzisztor (FGT) volt, ahol a szigetelőréteget szilícium dioxid (SiO_2) képezte. Ezt váltotta fel a geometriáját tekintve kisebb méretű Charge Trap Flash (CTF) technológia, amely viszont egy szilícium nitrid (Si_3N_4) alapú megoldás. A két megoldás elvi, illetve gyakorlati működésében nincs különbség, de a CTF gyártástechnológiája lehetővé tette az elemi cellák több egymás feletti síkban (24, vagy akár 32 síkban) történő elhelyezését. Ezt a többsíkú, nagy cellasűrűségű megvalósítást nevezzük 3D V-NAND technológiának.

Az általános célú SSD meghajtók jellemzően a NAND Flash technológiát használják, amiben az alkalmazott CTF technológia, valamint a soros szervezés, azaz a laponkénti címezhetőség miatt egy cella sokkal kevesebbszer írható, mint ahányszor olvasható. Vezérlő elektronika gondoskodik a cellák minél homogénebb használatáról. Egy pendrive vagy memóriakártya esetében ez a vezérlő elektronika lényegesen egyszerűbb és olcsóbb, mint az SSD meghajtók esetében. A cellák „kifáradása” azaz elhasználódása miatt a tároló (esetleg a nem is publikált teljes) kapacitásának csak kb. 80-90%-val rendelkezhetsz a felhasználó. Emellett a gyártók azt javasolják, hogy a publikált méretből is legalább 10%-ot hagyjunk üresen, hogy az SSD zökkenőmentesen adminisztrálhassa a cellák kiegyensúlyozását. Mióta már a server/storage világában is megjelentek az SSD-k, azóta pl. a polgári használatra szánt Samsung 8x0-es és 9x0 sorozatú SSD meghajtók 5 év garanciával kaphatók, mérettől függő TBW (Terrabytes Written/Írható Adatmennyiség) illetve DWPD (Drive Writes Per Day/Írható Napi Mennyiség) megköötéssel.

A SSHD meghajtók optimálisan ötvözik a HDD olcsó árát és nagy kapacitását az SSD sebességével.



Video: [Az SSHD működése](#)

Video: [Az SSD az SSHD és a HDD összehasonlítása](#)

Video: [Az SSD és a HDD összehasonlítása \(Samsung\)](#)

10. 3D XPoint technológia (3D CrossPoint a kiejtés!)

Tulajdonságai alapján a 2015. júliusában bemutatott 3D XPoint tűnik a jövő technológiájának a nem-felejtő memóriák piacán. A gyártó (az Intel karöltve a Micron-nal) 2016. év közepét jelölte meg, mint a 3D XPoint alapú memóriák piacra kerülésének tervezett dátuma.

A technológia pontos részletei még nem publikusak, annyi bizonyos, hogy a tárolás nem elektron alapú, és klasszikus értelemben vett tranzisztort illetve FET-et sem használ a tárolás megvalósításához, valamint nem igényli a DRAM-ok esetében megismert frissítést sem.

A 3D XPoint technológia a PCM (Phase-change Memory / Az anyag állapotváltozásán alapuló memória) technológia valamint a Memristor (Memory-Resistor / Emlékező ellenállás) technológia továbbfejlesztése.

- 1000-szer gyorsabb, mint a NAND technológia
- 1000-szer hosszabb élettartamú, mint a NAND technológia
- 10-szer nagyobb adatsűrűség, mint a NAND technológia

[Memristor: Az ellenállás, a kapacitás és az induktivitás mellett a negyedik passzív elem, melynek létezését Leon Ong Chua a Berkeley Egyetemen elvi szinten már 1971-ben bebizonyította, de működő prototípust legyártani csak 2008-ban sikerült a HP Labs-nak. Lényege, hogy az alkatrész képes "emlékezni" arra, hogy legutóbb mekkora áram folyt rajta keresztül, és ellenállása is ennek függvényében változik meg.]

A piac rossz felmérése, illetve a szerencsétlen árképzés ez esetben is kudarchoz vezetett, mint annak idején a RAMBUS memóriák esetében. 2021. március 16-án a Micron bejelentette, hogy leállítja a 3D XPoint fejlesztését a Compute Express Linken (CXL) alapuló termékek fejlesztése érdekében. A Micron a teljes technológiát és a Lehi-ben található gyárat is eladta a Texas Instruments-nek 900 millió USD-ért. 2022-ben az Intel is felszámolta az Optane részlegét, ezzel az Intel is kiszállt 3D Xpoint további fejlesztéséből. A Texas Instruments távlati tervei sajnos nem publikusak ebben a témában, a kifizetett hatalmas összeg ellenére sem...

Video: [Intel 3D XPoint Technology I.](#)

Video: [Intel 3D XPoint Technology II.](#)

Video: [Intel Optane Memory Powered by 3D XPoint Technology](#)

3D XPoint™ Technology: An Innovative, High-Density Design

Cross Point Structure

Perpendicular wires connect submicroscopic columns. An individual memory cell can be addressed by selecting its top and bottom wire.

Non-Volatile

3D XPoint™ Technology is non-volatile—which means your data doesn't go away when your power goes away—making it a great choice for storage.

High Endurance

Unlike other storage memory technologies, 3D XPoint™ Technology is not significantly impacted by the number of write cycles it can endure, making it more durable.

Stackable

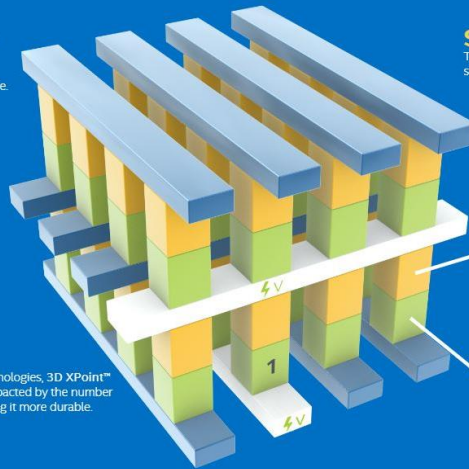
These thin layers of memory can be stacked to further boost density.

Selector

Whereas DRAM requires a transistor at each memory cell—making it big and expensive—the amount of voltage sent to each 3D XPoint™ Technology selector enables its memory cell to be written to or read without requiring a transistor.

Memory Cell

Each memory cell can store a single bit of data.



Transforming the Memory Hierarchy

For the first time, there is a fast, inexpensive and non-volatile memory technology that can serve as system memory and storage.

~8x to 10x Greater Density than DRAM¹

3D XPoint™ Technology's simple, stackable, transistor-less design packs more memory into less space, which is critical to reducing cost.



3D XPoint™ Technology

Processor



DRAM

3D XPoint™ Technology

NAND BASED NVMe SSD

3D XPOINT™ BASED NVMe SSD

7.13X
IOPS PERFORMANCE

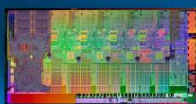
8.11X
LATENCY PERFORMANCE



3D XPOINT™ TECHNOLOGY

SRAM

Latency: 1X
Size of Data: 1X



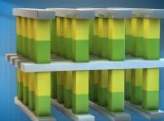
DRAM

Latency: ~10X
Size of Data: ~100X



3D XPoint™

Latency: ~100X
Size of Data: ~1,000X



STORAGE

NAND

Latency: ~100,000X
Size of Data: ~1,000X



HDD

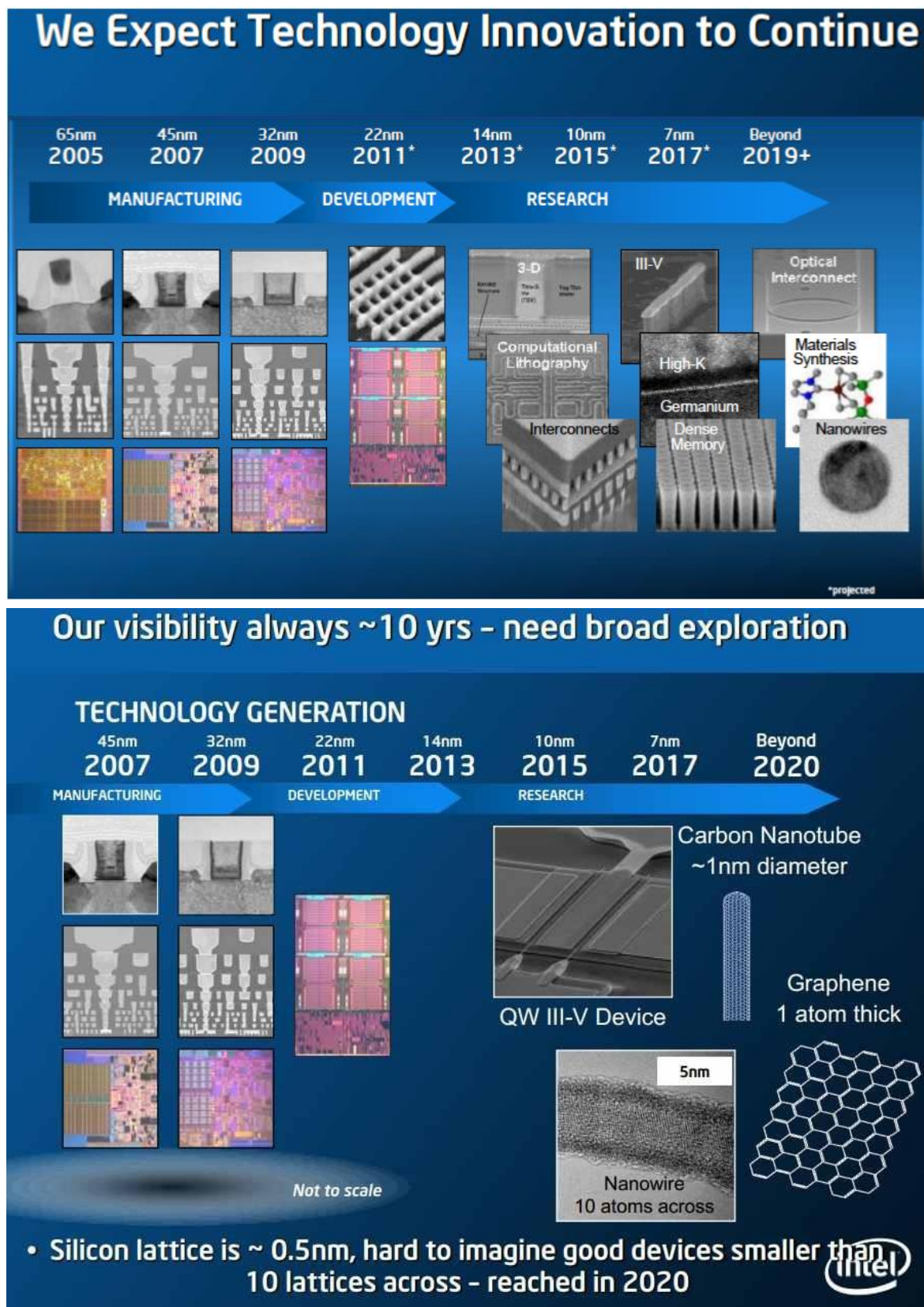
Latency: ~10 MillionX
Size of Data: ~10,000 X



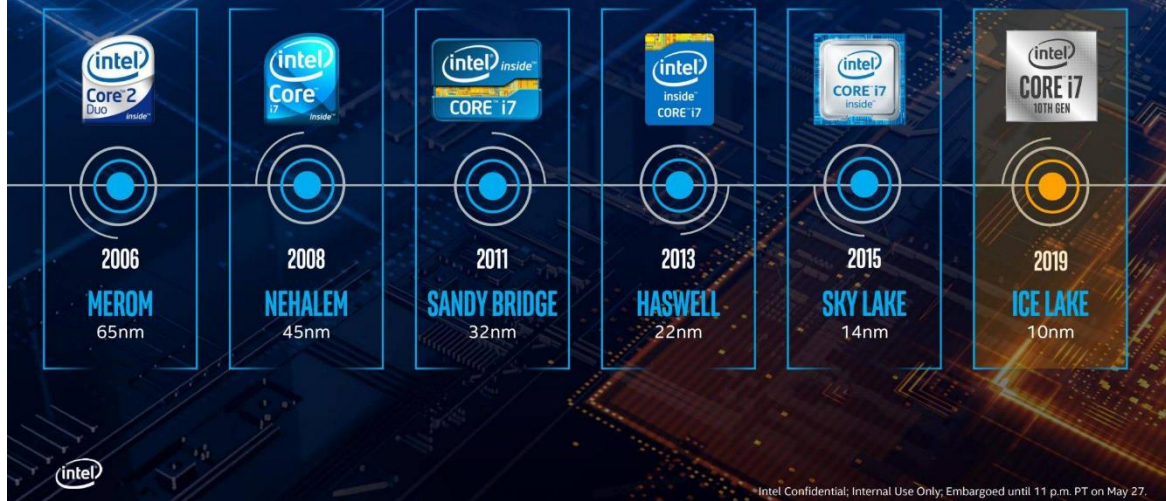
MEMORY

Hetedik fejezet

1. CPU evolúció
Intel



INTEL'S NEXT MAJOR ARCHITECTURE



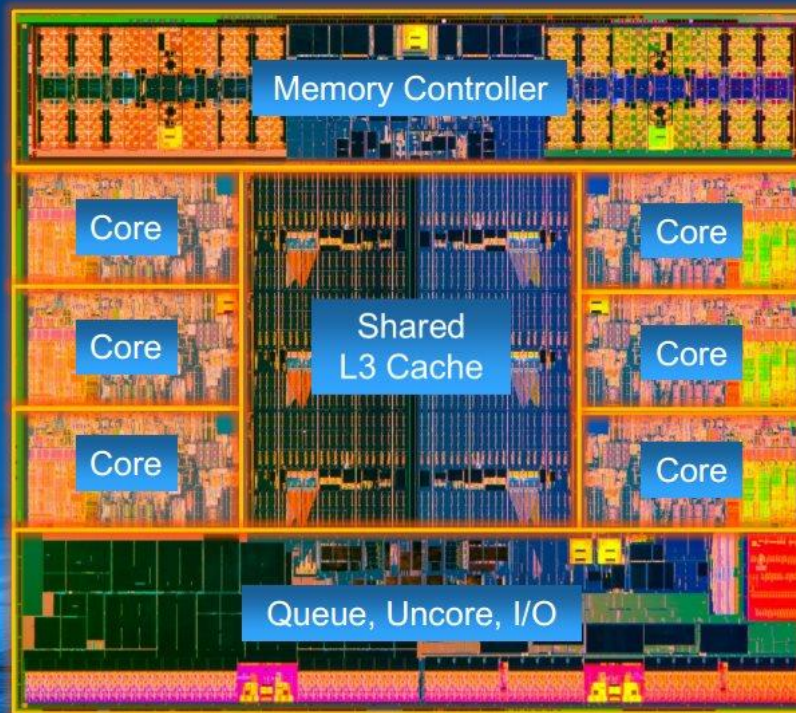
10TH GEN INTEL® CORE™ DESKTOP PROCESSORS

PROCESSOR NUMBER	BASE CLOCK SPEED (GHZ)	INTEL® TURBO BOOST TECHNOLOGY 2.0 MAXIMUM SINGLE CORE TURBO FREQUENCY (GHZ)	INTEL® TURBO BOOST MAX TECHNOLOGY 3.0 FREQUENCY (GHZ)	INTEL® THERMAL VELOCITY BOOST TECHNOLOGY SINGLE / ALL CORE TURBO FREQUENCY (GHZ) ¹	INTEL® ALL CORE TURBO FREQUENCY (GHZ)	CORES/ THREADS	THERMAL DESIGN POWER	UNLOCKED ²	PLATFORM PCIE 3.0 LANES	MEMORY SUPPORT ³	PROCESSOR GRAPHICS	INTEL® OPTANE™ MEMORY ⁴	RCP PRICING (USD 1K)
i9-10900K	Up to 3.7	Up to 5.1	Up to 5.2	Up to 5.3 / 4.9	Up to 4.8	10/20	125	✓	Up to 40	Two Channels DDR4-2933	Intel® UHD Graphics 630	✓	\$488
i9-10900KF	Up to 3.7	Up to 5.1	Up to 5.2	Up to 5.3 / 4.9	Up to 4.8	10/20	125	✓	Up to 40	Two Channels DDR4-2933		✓	\$472
i9-10900	Up to 2.8	Up to 5.0	Up to 5.1	Up to 5.2 / 4.6	Up to 4.5	10/20	65		Up to 40	Two Channels DDR4-2933	Intel® UHD Graphics 630	✓	\$439
i9-10900F	Up to 2.8	Up to 5.0	Up to 5.1	Up to 5.2 / 4.6	Up to 4.5	10/20	65		Up to 40	Two Channels DDR4-2933		✓	\$422
i7-10700K	Up to 3.8	Up to 5.0	Up to 5.1	NA	Up to 4.7	8/16	125	✓	Up to 40	Two Channels DDR4-2933	Intel® UHD Graphics 630	✓	\$374
i7-10700KF	Up to 3.8	Up to 5.0	Up to 5.1	NA	Up to 4.7	8/16	125	✓	Up to 40	Two Channels DDR4-2933		✓	\$349
i7-10700	Up to 2.9	Up to 4.7	Up to 4.8	NA	Up to 4.6	8/16	65		Up to 40	Two Channels DDR4-2933	Intel® UHD Graphics 630	✓	\$323
i7-10700F	Up to 2.9	Up to 4.7	Up to 4.8	NA	Up to 4.6	8/16	65		Up to 40	Two Channels DDR4-2933		✓	\$298

10TH GEN INTEL® CORE™ DESKTOP PROCESSORS

PROCESSOR NUMBER	BASE CLOCK SPEED (GHZ)	INTEL® TURBO BOOST TECHNOLOGY 2.0 MAXIMUM SINGLE CORE TURBO FREQUENCY (GHZ)	INTEL® TURBO BOOST MAX TECHNOLOGY 3.0 FREQUENCY (GHZ)	INTEL® THERMAL VELOCITY BOOST TECHNOLOGY SINGLE / ALL CORE TURBO FREQUENCY (GHZ) ¹	INTEL® ALL CORE TURBO FREQUENCY (GHZ)	CORES/ THREADS	THERMAL DESIGN POWER	UNLOCKED ²	PLATFORM PCIE 3.0 LANES	MEMORY SUPPORT ³	PROCESSOR GRAPHICS	INTEL® OPTANE™ MEMORY ⁴	RCP PRICING (USD 1K)
i5-10600K	Up to 4.1	Up to 4.8	NA	NA	Up to 4.5	6/12	125	✓	Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$262
i5-10600KF	Up to 4.1	Up to 4.8	NA	NA	Up to 4.5	6/12	125	✓	Up to 40	Two Channels DDR4-2666		✓	\$237
i5-10600	Up to 3.3	Up to 4.8	NA	NA	Up to 4.4	6/12	65		Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$213
i5-10500	Up to 3.1	Up to 4.5	NA	NA	Up to 4.2	6/12	65		Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$192
i5-10400	Up to 2.9	Up to 4.3	NA	NA	Up to 4.0	6/12	65		Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$182
i5-10400F	Up to 2.9	Up to 4.3	NA	NA	Up to 4.0	6/12	65		Up to 40	Two Channels DDR4-2666		✓	\$157
i3-10320	Up to 3.8	Up to 4.6	NA	NA	Up to 4.4	4/8	65		Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$154
i3-10300	Up to 3.7	Up to 4.4	NA	NA	Up to 4.2	4/8	65		Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$143
i3-10100	Up to 3.6	Up to 4.3	NA	NA	Up to 4.1	4/8	65		Up to 40	Two Channels DDR4-2666	Intel® UHD Graphics 630	✓	\$122

Intel® Core™ i7-4960X Processor Die Detail

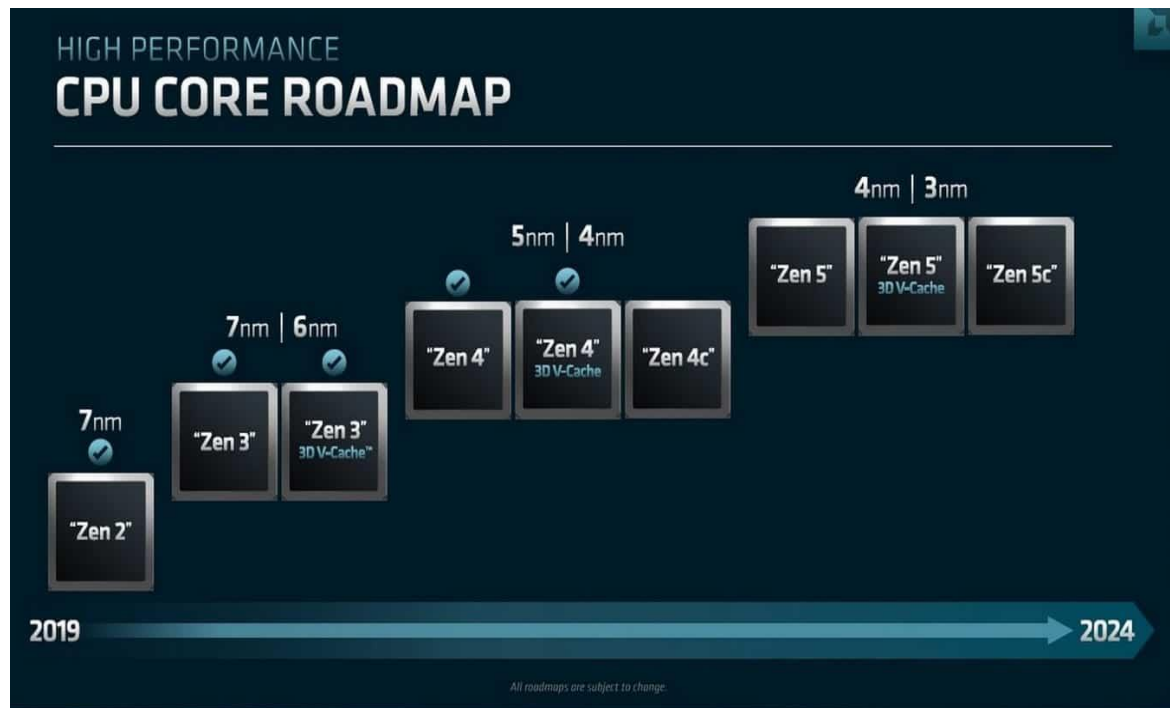
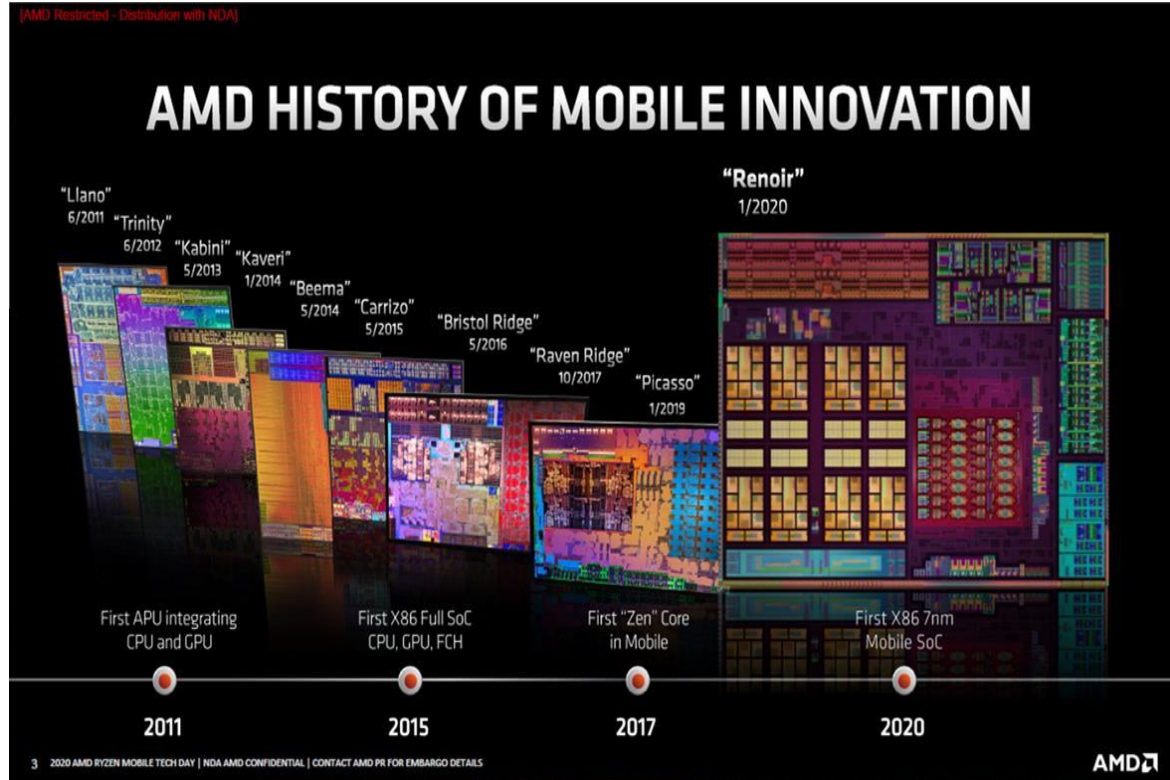


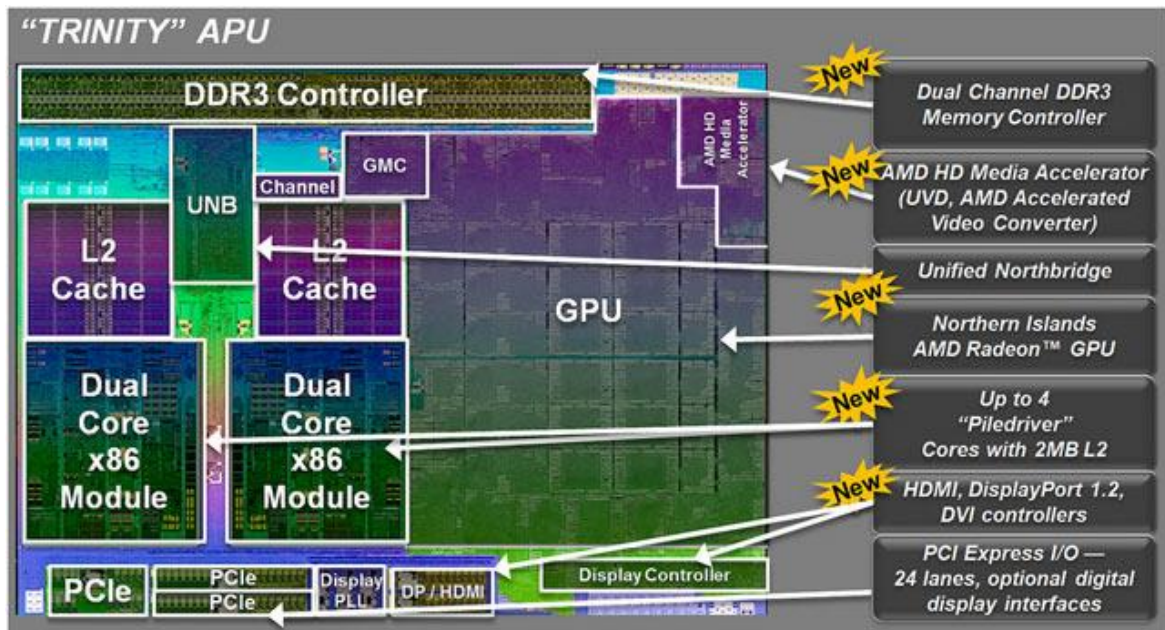
Total number of transistors 1.86B Die size dimensions 15.0 mm x 17.1 mm [257 mm²]

Intel i7-4690X, Ivy Bridge-E, 22nm, 3.6GHz (2013)
 cores: 6, L2 cache: 6x256kB, L3 cache: 15MB, LGA2011, 130W

Intel vs AMD CPU Roadmaps

	2022	2023	2024	2025
Intel	Raptor Lake	Meteor Lake	Arrow Lake	Lunar Lake
	RPT Cove (P) + Gmt (E)	Redwood Cove (P) + Crestmont (E)	Lion Cove (P) + Skymont €	TBA
	8P+16E Cores	8P+16E	16P+16E?	TBA
	7nm-DUV-FinFET	4nm-EUV-FinFET	20A-EUV-GAA	18A-EUV-GAA
	Monolithic	Tiled v1	Tiled v2	Tiled v3
AMD	Outcome 3	Outcome 1	Outcome 2	Outcome 32
	Raphael		Granite Ridge/Strix Point	TBA
	Zen 4		Zen 5 + Zen 4c	TBA
	16P (8+8) Cores		24P + 16E Cores?	TBA
	TSMC 5nm-EUV-FinFET		TSMC 3nm-EUV-FinFET	TBA
	Chiplet v3	Chiplet v4	TBA	

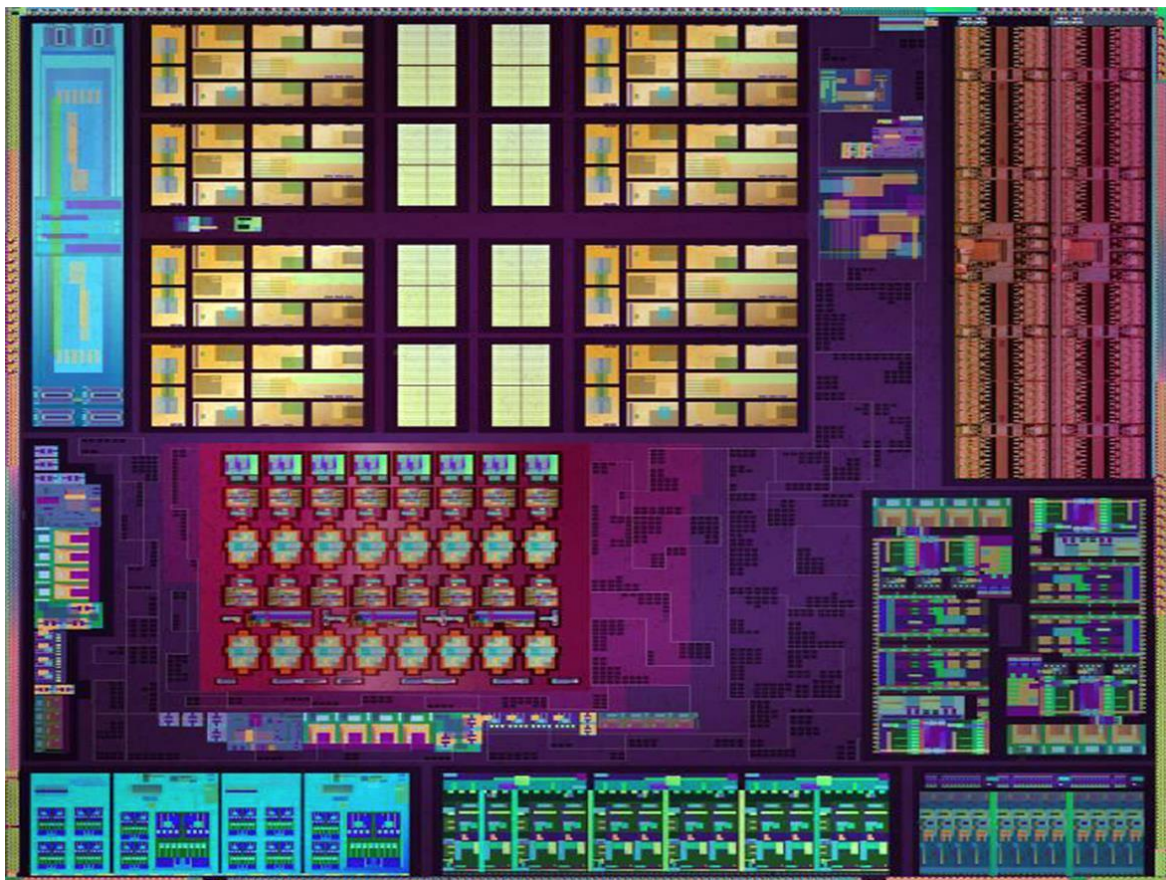




AMD A6-5400K, Trinity, 32nm, 3.8GHz (2012)

cores: 2, L2 cache: 2x1MB, FM2, 65W

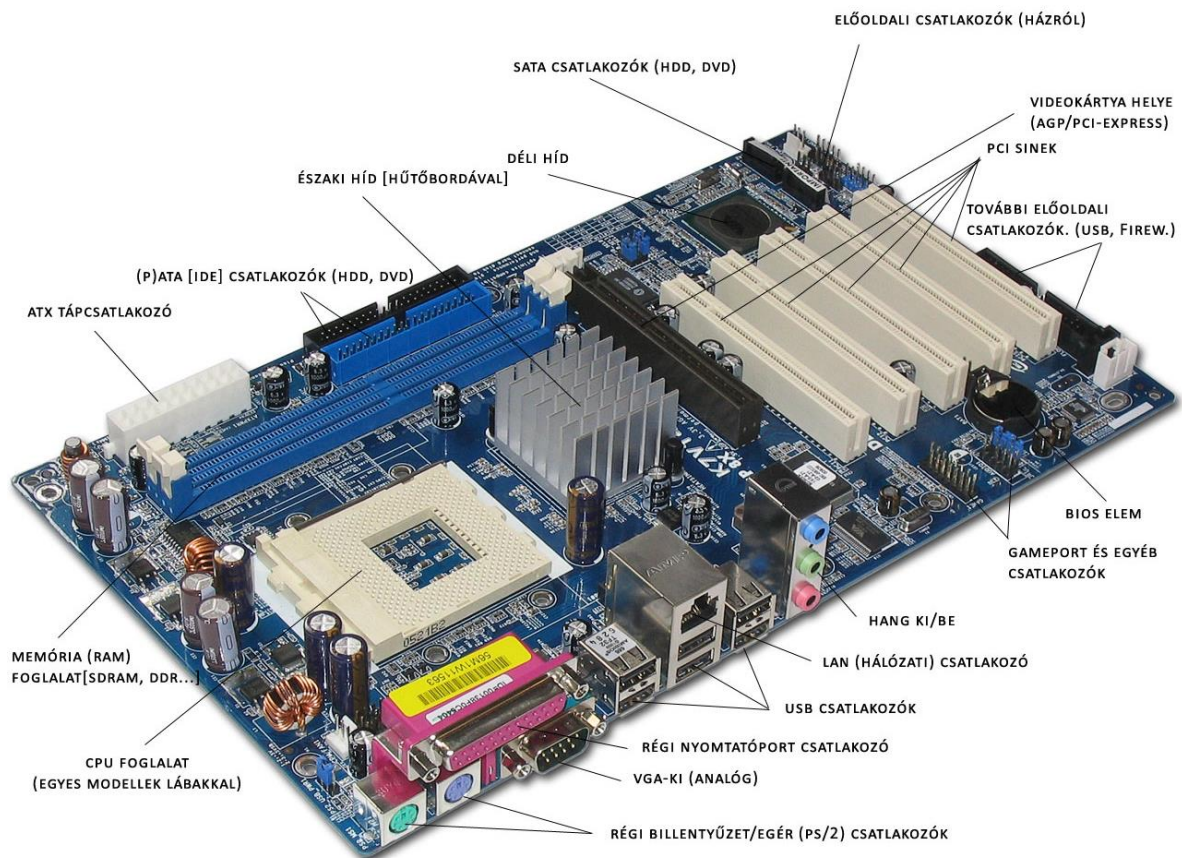
[APU: Accelerated Processing Unit = CPU + GPU]



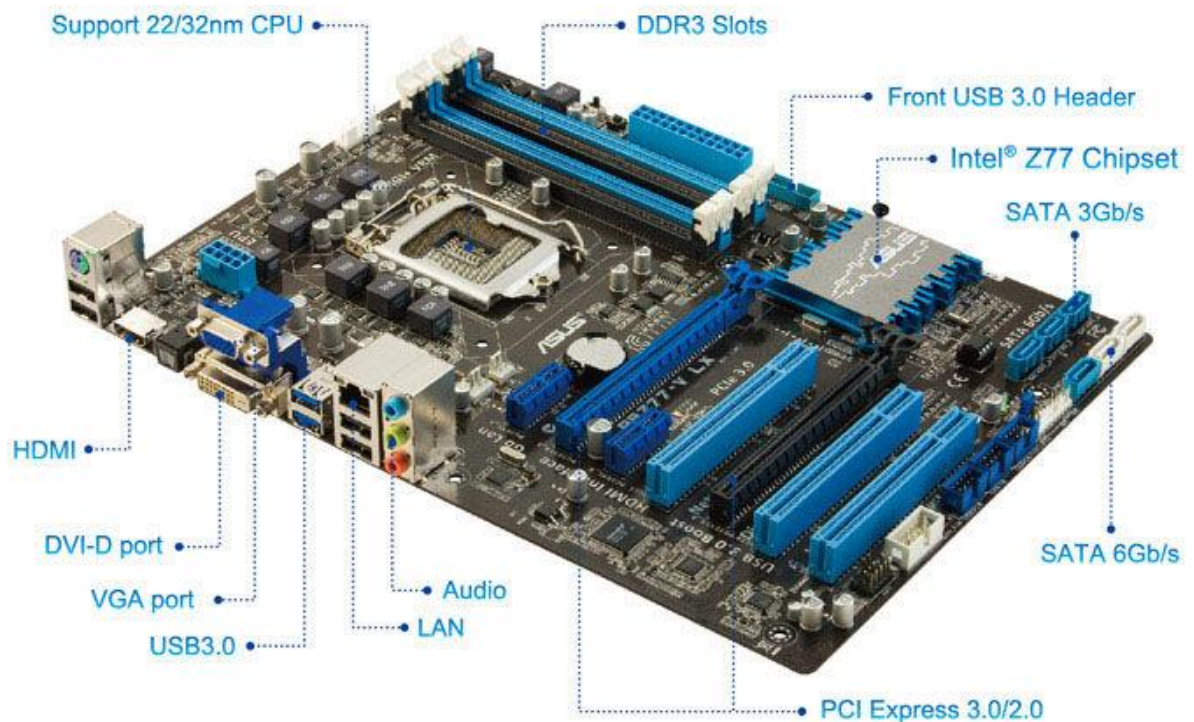
AMD Ryzen7 4750G, 7nm, 3.6GHz (2020)

cores: 8 CPU + 8 GPU, L2/L3 cache: 12MB, AM4, 65W

ASRock K7VT4A Pro alaplap
 AMD SocketA [462 pin], VIA KT400A, FSB333, DDR1 400MHz, 8xAGP



ASUS P8Z77-V LX alaplap
 Intel S1155, Z77, DDR3 2400, 2x PCIe 16x, USB3.0



2. A főbb hardware elemek és perifériák

A főbb hardware elemek és perifériák – egyrészt az előző előadás közvetlen folytatásaként, illetve a „perifériák.ppt” a #22. diáig, felhasználásával.

3. I/O kommunikációval kapcsolatos szabványok rövidítései:

- fontosabb belső csatlakozási módok

MFM:	Modified Frequency Modulation	5Mb/s
RLL:	Run Length Limited	7.5Mb/s
ESDI:	Enhanced Small Disk Interface	10 – 20Mb/s
IDE:	Integrated Drive Electronics	
ATA:	Advanced Technology Attachment	33, 66, 133MB/s
[PATA:	Paralell ATA]	
SATA:	Serial ATA [max / device]	1.5Gb/s / 150MB/s, 3Gb/s / 300MB/s, 6Gb/s / 600MB/s 7pin

- fontosabb külső csatlakozási módok (néhány használatos belső csatlakozásra is)

SCSI:	Small Computer System Interface	160Mb/s, 320Mb/s, 640Mb/s
SAS:	Serial Attached SCSI	3 – 6Gb/s
iSCSI:	Internet SCSI	1 – 10Gb/s
eSATA:	External SATA	115Mb/s
FC:	Fibre Channel	1 – 16Gb/s
IB:	InfiniBand [1x, 4x, 12x Link Bandwidth]	
	QDR (Quad Data Rate, 2008)	8Gb/s, 32Gb/s, 96Gb/s
	FRD (Fourteen Data Rate, 2011)	14Gb/s, 56Gb/s, 168Gb/s
	EDR (Enhanced Data Rate, 2014)	25Gb/s, 100Gb/s, 300Gb/s
		2x42pin CPX connector, 3db 4x port

PnP:	Plug And Play	
USB:	Universal Serial Bus (PnP)	12Mb/s, 480Mb/s, 5Gb/s
FW:	Fire Wire (nem PnP!)	400Mb/s, 800Mb/s
PCMCIA:	Personal Computer Card International Association	133Mb/s
ExpressCard:		280Mb/s, 1.6Gb/s, 3.2Gb/s
SPDIF:	Sony Philips Digital Interface Format	optical audio
D-SUB:	„D”-shaped Subminiature	DSUB-15 analog video, 15 pin
DVI:	Digital Visual Interface	digital+analog video, 24+4 pin
DisplayPort:		digital video+audio, 20 pin
HDMI:	High Definition Multimedia Interface	digital video+audio, 3D, 19 pin

4. RAID technológia, RAID Szintek

Redundant Array of Inexpensive Disks / Redundant Array of Independent Disks

[Redundancia: A mindenképpen szükséges (adat)mennyiségen felül tárolt, illetve átvitt (adat)többség. A redundancia igénybevételenek fő okai az adatbiztonság és a hibajavíthatóság.]

Adat tárolási technológia, melynek alapja az adatok elosztása vagy replikálása több fizikailag független merevlemezen. A végeredmény egy logikai lemez. Minden RAID szint alapjában véve és/vagy az adatbiztonság növelését és/vagy az adatátviteli sebesség növelését szolgálja. Magasabb szinteken a tároló diszk(ek) meghibásodása esetén is az adatok biztonsága, visszaállíthatósága a fő szempont.

- RAID 0

Összefűzés. Nem tartalmaz redundanciát, és így hibatűrést sem, azaz egyetlen meghajtó meghibásodása az egész tömb hibáját okozza. Mind az írási, mind az olvasási műveletek párhuzamosítva történnek, ideális esetben a sebesség az egyes lemezek sebességének összege lesz, így a módszer a RAID szintek közül a legjobb teljesítményt nyújtja (a többi módszernél a redundancia kezelése lassítja a rendszert). A megoldás lehetővé teszi különböző kapacitású lemezek összekapcsolását is, viszont a nagyobb kapacitású lemezeket is csak a tömb legkisebb kapacitású lemezének méretéig lehet használni.

- RAID 1

Tükrözés. Az információk egyidejű tárolása a tömb minden elemén. A kapott logikai lemez a tömb legkisebb elemével lesz egyenlő méretű. Az adatok olvasása párhuzamosan történik a diszkekről, felgyorsítván az olvasás sebességét; az írás normál sebességgel, párhuzamosan történik a meghajtókon. Az eljárás igen jó hibavédelmet biztosít, bármely meghajtó meghibásodása esetén folytatódhat a működés.

- RAID 2 (nem használatos technológia, főleg technikatörténeti jelentősége van)

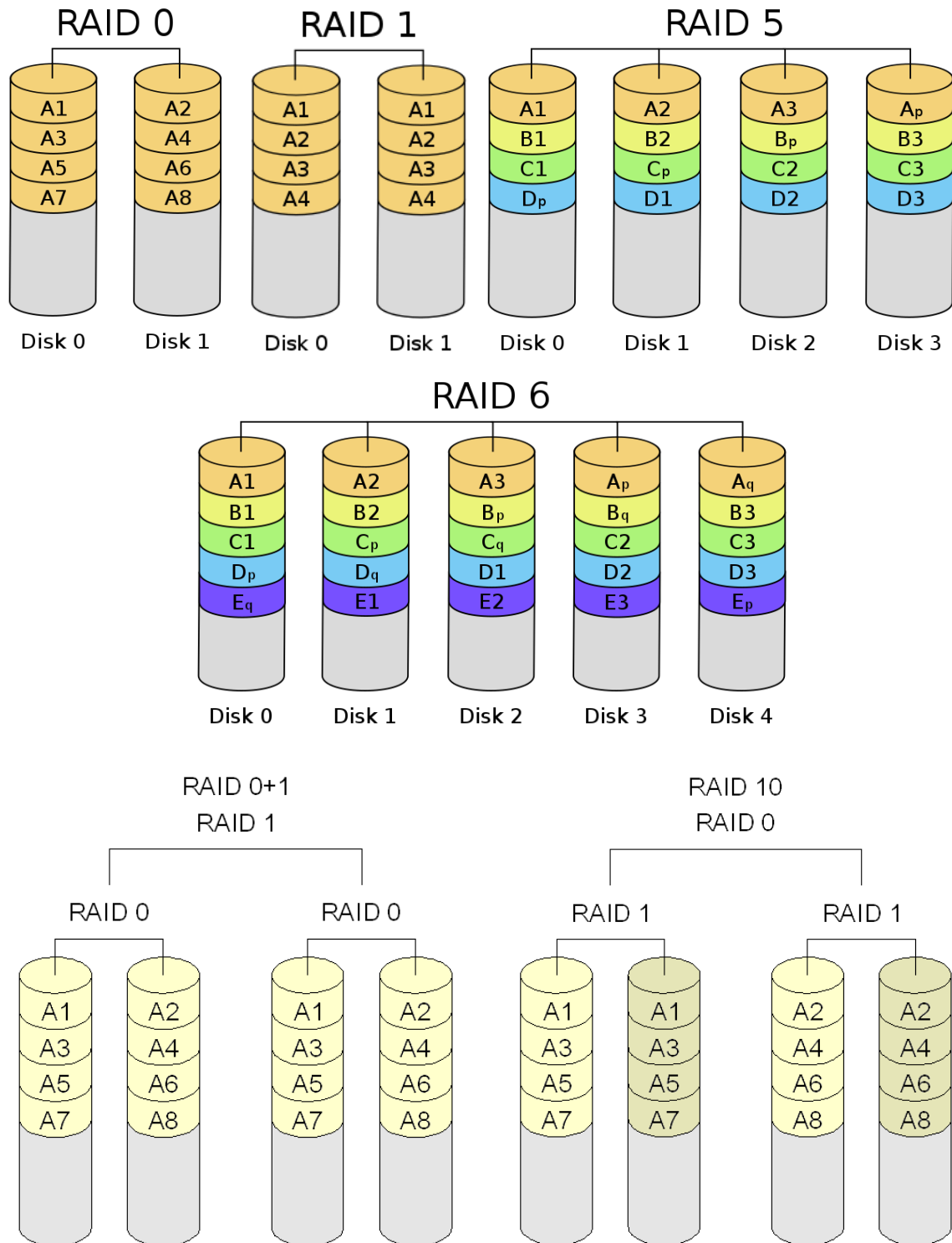
Nem abszolút redundáns tárolás, a csíkokra bontás módszerét használja, egyes meghajtókat csak a bit szintű hibajavító kód (Hamming kód) segítségével előállított redundáns bitek tárolására tart fenn (pl. 4db adat diszkhez 3db hibajavításra használatos diszk kell). A használt eljárástól függően a kapott kód akár több bithiba észlelésére, illetve javítására alkalmas. A hibajavításra használt meghajtók egy-egy csíkjában a két különböző lemez azonos pozícióban elhelyezkedő csíkjából képzett hibajavító kód tárolódik. A módszer lemezhiba esetén képes annak detektálására, illetve kijavítására. Már nem használatos megoldás, okafogyottá vált, hiszen már az SCSI meghajtók óta minden egyes szektorban az adott szektorhoz tartozó hibajavító kód (ECC) is eltárolódik magán az adott diszken.

- RAID 3 (nem használatos technológia, főleg technikatörténeti jelentősége van)

A RAID 3 felépítése hasonlít a RAID 2-re, viszont ez esetben mindig csak egy lemeznyi hibajavítási, úgynevezett paritásinformáció tárolódik. Egy adott paritáscsík a különböző lemezek azonos pozícióban elhelyezkedő csíkokból (a RAID 2-től eltérően nem bit, hanem bájt szinten) egy XOR művelet segítségével kapható meg. A rendszerben egy meghajtó kiesése nem okoz problémát, mivel a rajta lévő információ a többi meghajtó adataiból – a paritást tároló meghajtót is beleértve – csíkonkénti XOR művelettel megkapható. Nyilvánvaló, hogy csak az XOR-os paritásbit segítségével az egyik meghajtón egy adott bájt megsérülése esetén csak azt vennénk észre, hogy a különböző meghajtókon az azonos csíkba tartozó részek esetében az XOR művelet nem nullát ad, ebből az még nem állapítható meg, hogy melyik meghajtón van a hiba, sem azt, hogy hogyan javítsuk ki. A hibajavításhoz tehát ez esetben már eleve szükség van a diszkek szektoronkénti hibajavító kódjára is.

- RAID 4 (nem használatos technológia, főleg technikatörténeti jelentősége van)
A RAID 4 felépítése a RAID 3-mal megegyezik. Az egyetlen különbség a nagyobb, blokk szintű csíkméretben van. Ez a megoldás a nagyobb méretű fájlok használata esetén előnyös, mivel lehetővé teszi egyszerre több (különböző meghajtókon elhelyezkedő) rekord párhuzamos írását, illetve olvasását (Multi-User Mode). Probléma viszont, hogy a paritás meghajtó jelentős méretű csíkjait minden egyes íráskor frissíteni kell (ez plusz egy olvasást és írást jelent), aminek következtében párhuzamos íráskor a paritás meghajtó válik a rendszer szűk keresztmetszetévé. Ebből az is következik, hogy valamely meghajtó kiesése esetén a rendszer olvasási teljesítménye lecsökken, a paritás meghajtó jelentette szűk keresztmetszet miatt.
- RAID 5
Ez a megoldás a paritás információt nem egy kitüntetett meghajtón, hanem „körbeforgó paritás” (Rotating Parity) használatával, egyenletesen az összes meghajtón elosztva tárolja, így kiküszöbölve a paritás meghajtó jelentette szűk keresztmetszetet. A csíkméret paraméterezhető, kisméretű csíkok esetén a RAID 3 jellegű, míg nagyméretű csíkok esetén a RAID 4 jellegű működést kapunk. A minimális meghajtó szám: 3. Mind az írási, mind az olvasási műveletek párhuzamosan végezhetőek el. Egy meghajtó meghibásodása esetén az adatok nincsenek veszélyben, a hibás meghajtó adatait a rendszer a többi meghajtó adataiból vissza tudja állítani. A hibás meghajtót ajánlott azonnal cserélni, mert két meghajtó meghibásodása esetén az adatok biztosan elvesznek. A tárolható adatmennyiség „a legkisebb kapacitású meghajtón tárolható adatmennyiség” * („meghajtók száma” - 1) lesz. (Pl. 4db egyenként 1TB-os HDD RAID 5-be fűzésének eredményeként egy 3TB kapacitású logikai meghajtót látunk.) A RAID 5 vezérlők a hibás diszk helyére betett új, üres diszke automatikusan fel tudják tölteni az eredeti adatokat.
- RAID 6
Ez a megoldás a RAID 5 logikai kibővítése. Itt nemcsak soronként, hanem oszloponként is kiszámítják a paritást. A módszer segítségével kétszeres meghajtó meghibásodás is kiküszöbölhetővé válik. A paritáscsíkokat itt is az egyes meghajtók között, egyenletesen elosztva tárolják, de ezek természetesen kétszer annyi helyet foglalnak el, mint a RAID 5 esetében.
- RAID 0+1 (vagy RAID 01)
Ez egy olyan hibrid megoldás, amelyben a RAID 0 által hordozott sebességet a RAID 1-et jellemző biztonsággal ötvözhetjük. Hátránya, hogy minimálisan 4 eszközre van szükségünk, melyekből 1-1-et összefűzve, majd páronként tükrözve építhetjük fel a tömbünket, ezért a teljes kinyerhető kapacitásnak mindössze a felét tudjuk használni. Mivel a tükrözés (RAID 1) a két összefűzött (RAID 0) tömbre épül, ezért egy lemez meghibásodása esetén az egyik összefűzött tömb mindenképp kiesik, így a tükrözés is szétesik.
- RAID 1+0 (vagy RAID 10)
Hasonlít a RAID 01 megoldáshoz, annyi különbséggel, hogy itt a lemezeket először tükrözzük, majd a kapott tömböket fűzzük össze. Ez biztonság szempontjából jobb megoldás, mint a RAID 01, mivel egy diszk kiesése csak az adott tükrözött tömböt érinti, a rá épült RAID 0-t nem. Sebesség szempontjából a RAID 01 és a RAID 10 között nincs különbség.

A RAID 10-hez hasonló elven két (vagy több) RAID 5 tömb összefűzésével építhetünk egy RAID 50-es tömböt, illetve két (vagy több) RAID 6 tömb összefűzésével építhetünk egy RAID 60-as tömböt. A RAID technológia nyújtotta adattárolási biztonság és az üzembiztonság a korszerű „Disc Storage”-ok használata esetén a „Local Spare” és „Global Spare” diszkek kijelölésével tovább növelhető.



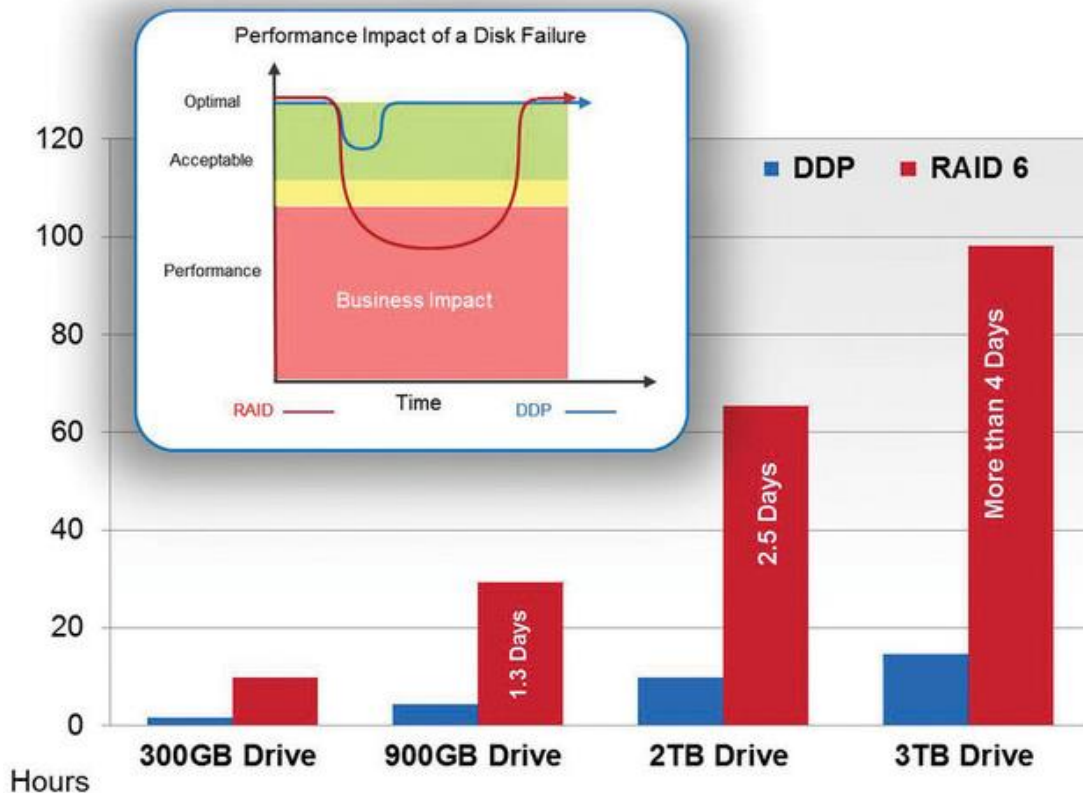
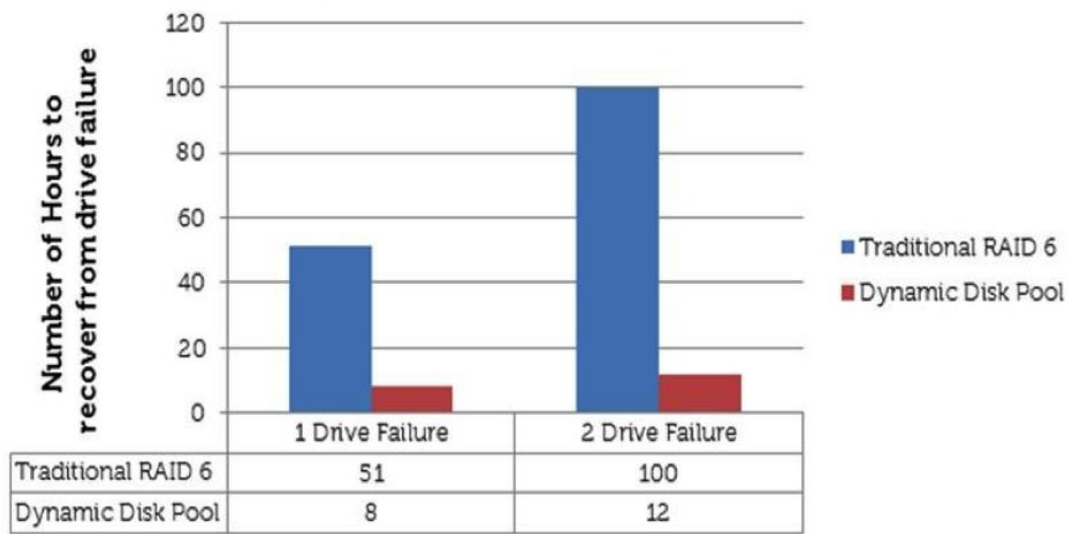
RAID calculator: <http://www.raid-calculator.com/default.aspx>

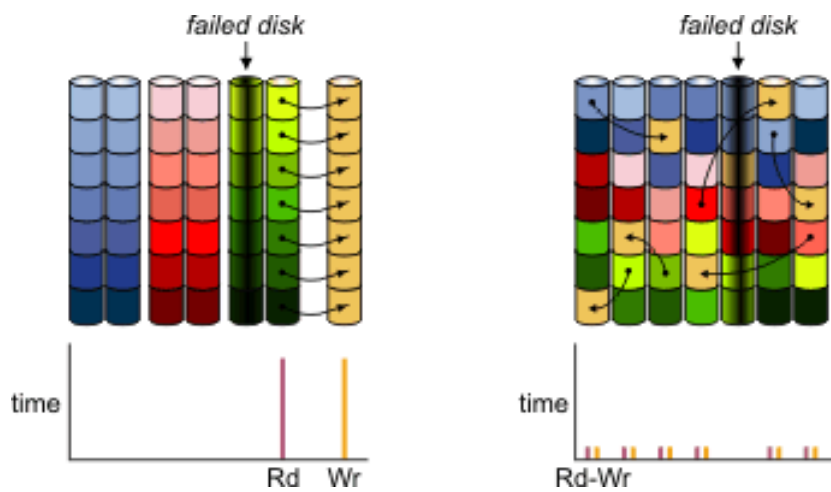
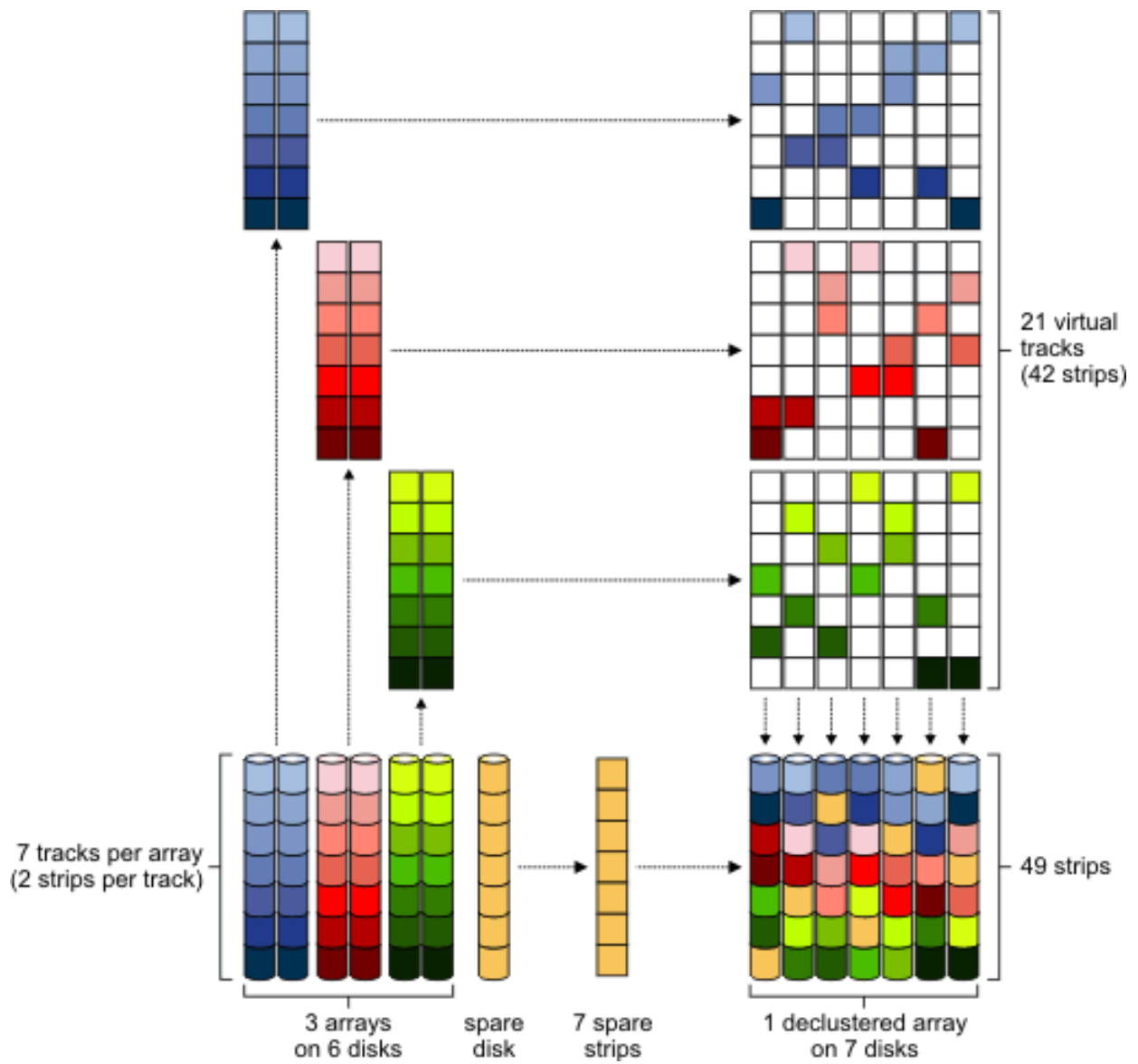
Video: [A RAID technológia működése.](#)

5. Dynamic DiskPool technológia

A RAID technológia megalkotásakor (1987) a diszkek kapacitása legfeljebb néhány GB volt. A technológia igazi fénykora pedig a néhány száz GB kapacitású diszkekhez köthető. A diszkek kapacitásának növekedését az adatátvitel sebességének növekedése nem volt képes követni, így az az érdekes helyzet állt elő, hogy a RAID tömböket alkotó egyre nagyobb diszkek esetében a hibajavítás, azaz a tartalom átmásolása (és az azt megelőző logikai leképezése) egyre hosszabb időt vett igénybe, és jelentős írási és olvasási erőforrást igényelt a tárolóeszköztől.

Traditional RAID 6 vs. Dynamic Disk Pool Recovery Time for 3TB NLSAS Drives





A fenti 7 lemezes példa egy elvi működési magyarázat. A valóságban minimum 11db HDD vagy SSD szükséges a Dynamic Disk Pool működéséhez.

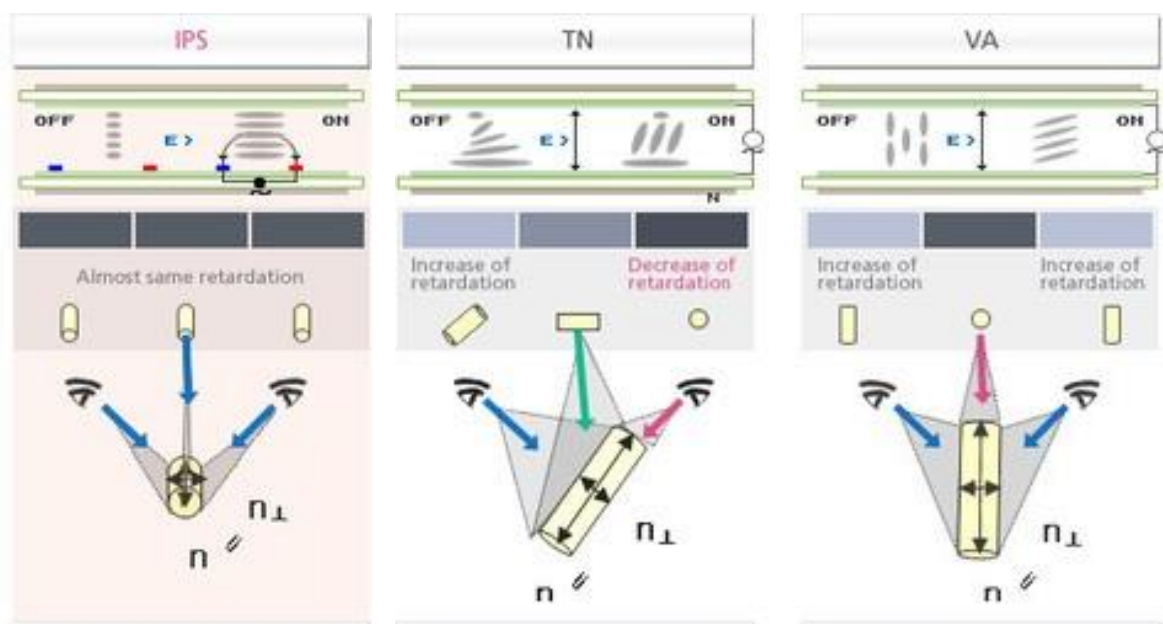
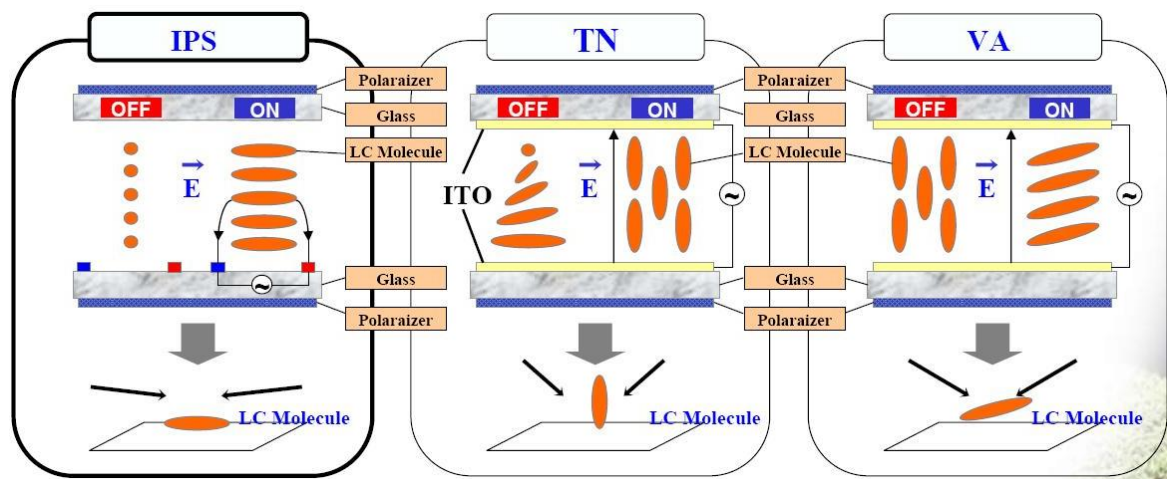
A Biztonsági másolat, a Mentés és az Archiválás összehasonlítása

- Biztonsági másolat
Adott állomány dupla (esetleg többszörös) tárolása, lehetőleg különböző tároló eszközökön.
- Mentés
Valamely adatról (fájl, e-mail, adatbázis-rekord stb.) másolat készítése az adatvesztés elkerülése érdekében. A mentést azért végezzük, hogy vész esetén az adatokat helyreállíthassuk. A mentésnél többnyire nem cél a keresés biztosítása. A mentés lehet teljes, vagy inkrementális [növekményes], amikor csak a megváltozott állományok kerülnek mentésre. Egy inkrementális mentés utáni mentéskor lehetőségünk van úgynevezett differenciális [különbségi] mentésre, amikor csak az utolsó inkrementális mentés óta megváltozott adataegységek kerülnek elmentésre. Ha két teljes mentés között több differenciális mentést végzünk, akkor pl. a második differenciális mentés csak az első óta történt változásokat fogja rögzíteni. Ennek köszönhetően maga a mentés folyamata gyorsabbá válik, és esetenként kevesebb helyet foglal el. Hátránya azonban, hogy a visszaállításhoz a legutolsó teljes mentésre, és az azt követő összes differenciális mentésre szükség van.
- Archiválás
Valamely adat (fájl, e-mail, adatbázis-rekord stb.) áthelyezése olyan adathordozóra, amelyen azután az adat minőségének megfelelő ideig megőrizhessük az archivált adatot. Az archiválást azért végezzük, hogy (többnyire olcsó) adathordozón meghatározott (többnyire szabályozott) ideig megőrizzük az adatainkat. Az archiválásnál általában cél a kereshetőség biztosítása.

A tömörítés a gyakorlat szerint a legtöbb esetben inkább hátrány, hiszen mindkét irányban idővesztéssel jár, és a ma használatos adatstruktúrák esetében (általában) nem jelent érdemi tároló hely megtakarítást.

Nyolcadik fejezet

1. A főbb hardware elemek és perifériák
(a „perifériák.ppt” a #23. diától, és a „10_ch10_hun.pdf”, mint háttér információ felhasználásával).
2. Kijelzők: CRT – Catode Ray Tube – katódsugárcsöves kijelző
Video: [A CRT működése](#)
3. Kijelzők: Plazma
A Neon és Xenon gáz plazma kijelzők előnye a magas kontrasztarány és színhűség, mivel egyedileg felvillantott sub-pixelekkel alkotja a képet. A többi lapos kijelzőhöz képest hátránya a fajlagosan drágább ár, és a működéséhez szükséges többlet energia.
Video: [A Plazma kijelző működése](#)
4. Kijelzők: LCD – Liquid Cristal Display – folyadékkristályos kijelző
TN: Twisted Nematic, VA: Vertical Alignment, IPS: In-Plane Switch



Video: [Az LCD kijelző működése](#)

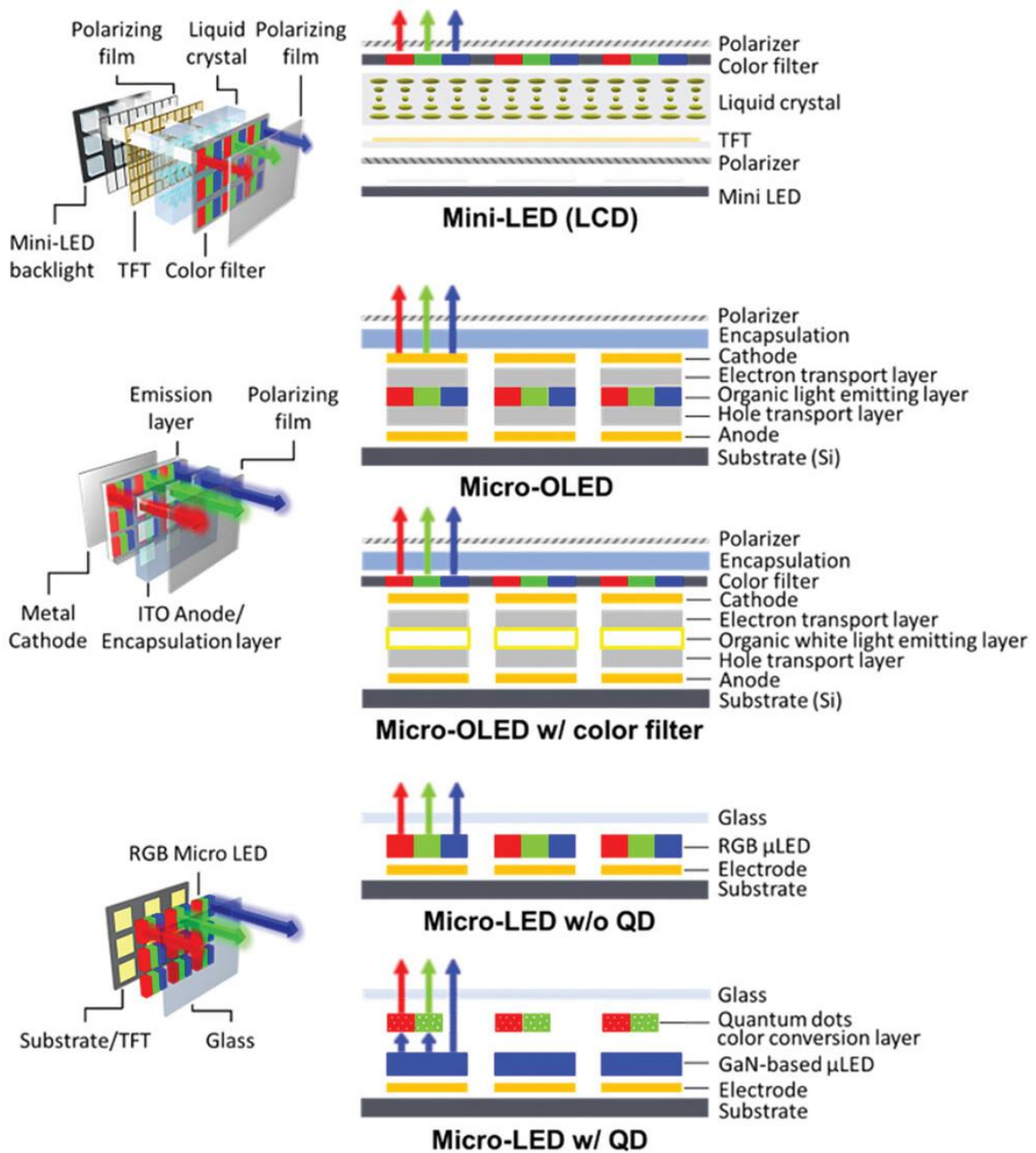
Video: [A CCFL és a LED háttérvilágítás összehasonlítása \(SONY\)](#)

5. Kijelzők: IGZO – indium, gallium, zinc, oxygen
 Video: [Az IGZO bemutatása I. \(Sharp\)](#)
 Video: [Az IGZO bemutatása II. \(Sharp\)](#)
 Video: [Egy 32" 4K IGZO monitor bemutatása \(Sharp\)](#)

6. Kijelzők: OLED

Előnye, hogy nem igényel háttérvilágítást és hajlékony is lehet, de hátránya, hogy nem időtálló, mivel organikus azaz szerves molekulák végzik a fény kibocsájtást. A szerves molekulák kifáradása miatt idővel (évek alatt) veszít a színhelyességéből (sárgább lesz a kép) illetve a fényerőből is.

Video: [Az OLED kijelző működése](#)



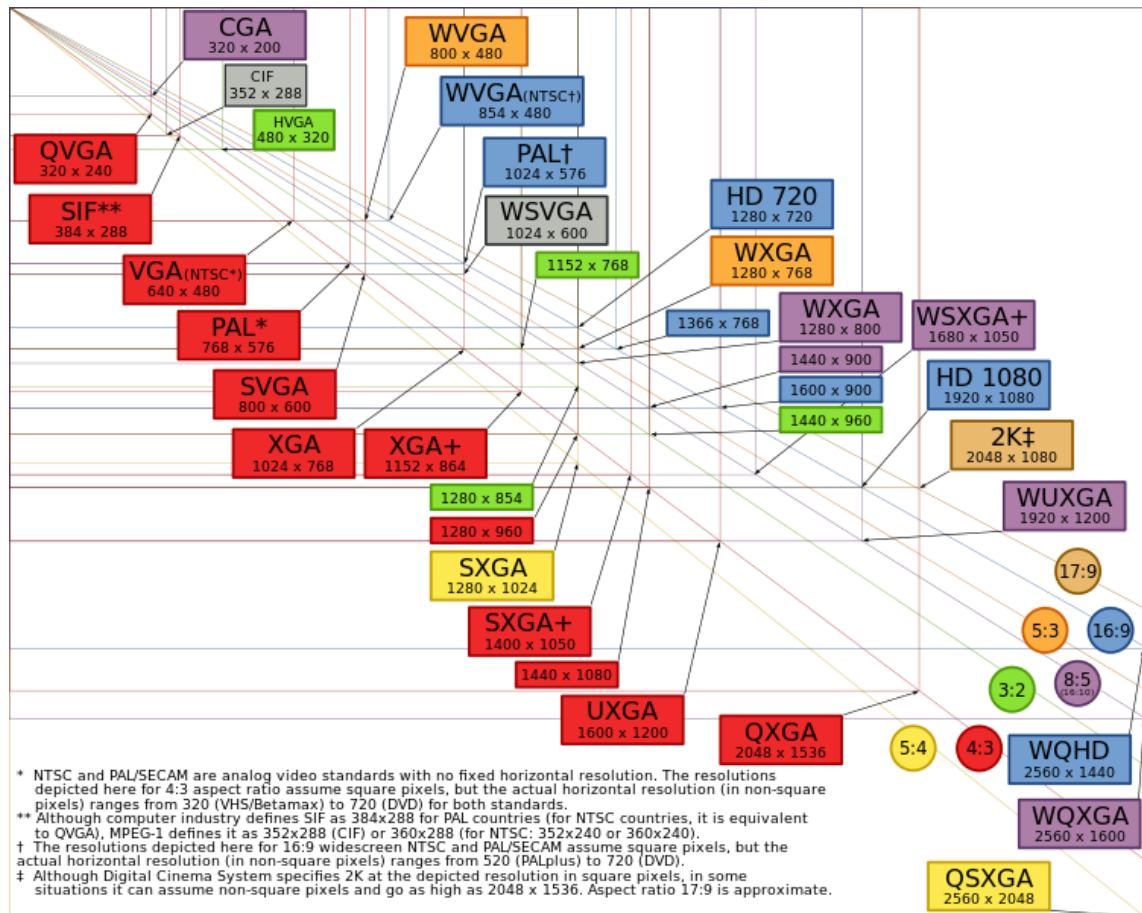
7. Kijelzők: MicroLED

Az önálló fényforrás mikró méretűre csökkentése a SONY nevéhez fűződik, 2012-ben mutatták be a technológiát Crytal LED néven. Ezt fejlesztette tovább 2018-ra a Samsung Micro LED elnevezéssel.

8. Érintőképernyő

Video: [A Touchscreen működése](#)

9. A kijelzők által használt leggyakoribb felbontások.



10. Projektorok: DLP

Video: [A DLP projektor működése I.](#)

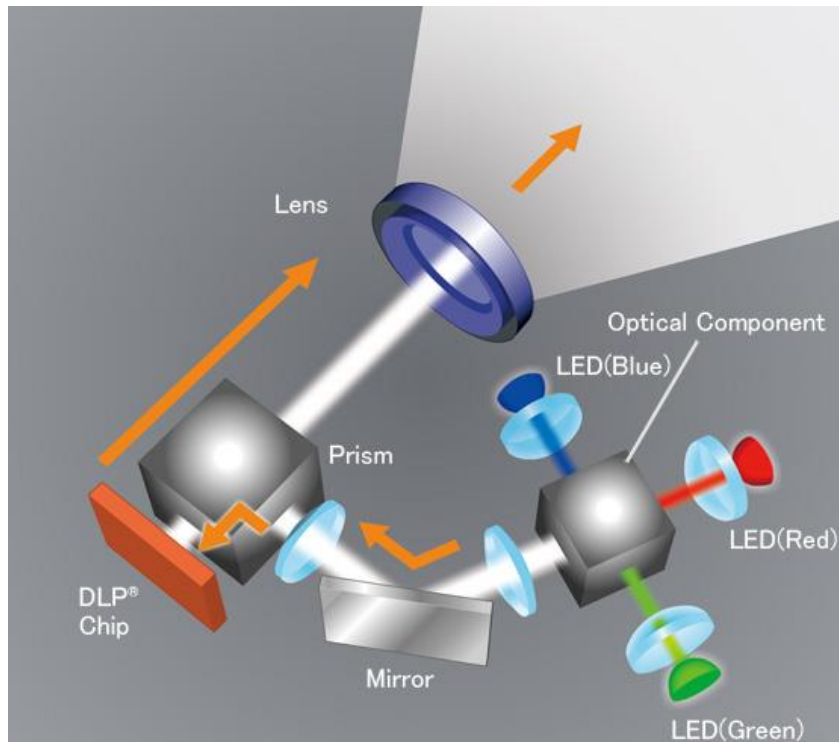
Video: [A DLP projektor működése II.](#)

11. Projektorok: 3LCD

Video: [A 3LCD projektor működése I.](#)

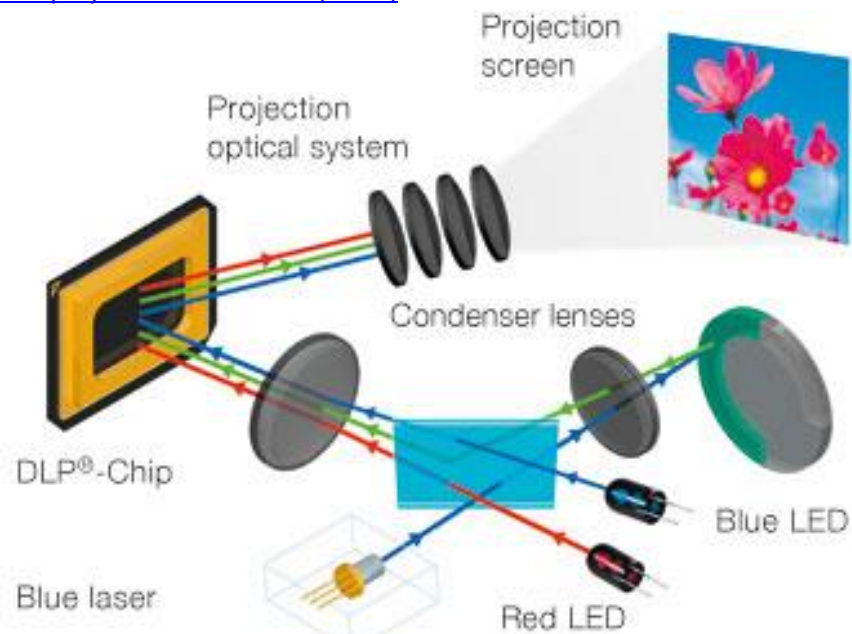
Video: [A 3LCD projektor működése II.](#)

12. Projektorok: LED



13. Projektorok: Laser + DLP Hybrid

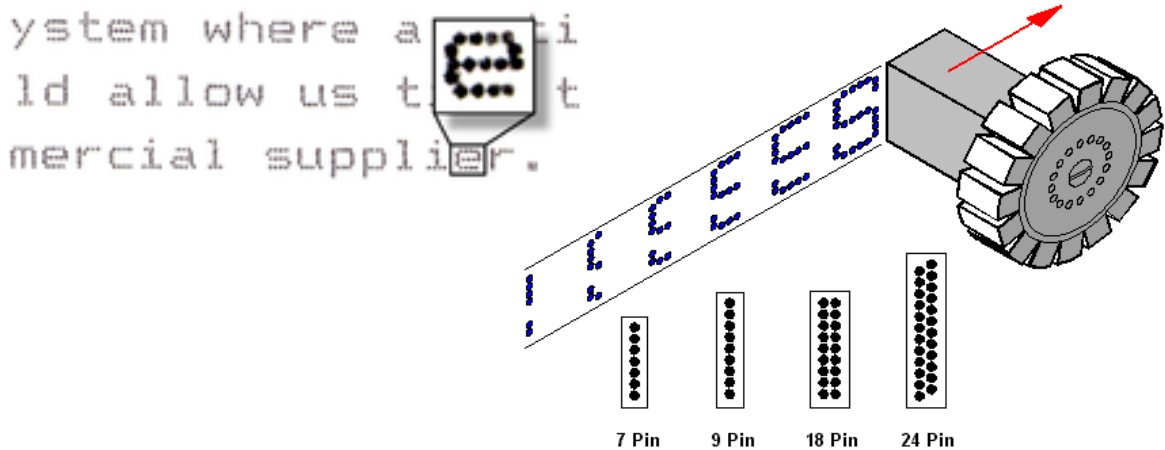
Video: [A hybrid projektor bemutatása \(Casio\)](#)



14. Nyomtatók: Mátrixnyomtatók

Video: [A színes mátrixnyomtató működése \(Citizen\)](#)

From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.



15. Nyomtatók: Tintasugaras nyomtatók

Video: [A színes tintasugaras INK JET nyomtató működése \(HP\)](#)

Video: [A színes tintasugaras BUBBLE JET nyomtató működése \(Canon\)](#)

Video: [A színes tintasugaras PIEZO nyomtató működése \(EPSON\)](#)

16. Nyomtatók: Lézernyomtatók

Video: [A szkenneres lézer nyomtató működése \(Lexmark\)](#)

Video: [A szkenneres színes lézer nyomtató működése \(HP\)](#)

Video: [A színes LED lézer nyomtató összehasonlítása a szkenneres nyomtatóval \(OKI\)](#)

17. Nyomtatók: Termotranszfer nyomtatók

Video: [A termotranszfer nyomtató bemutatása \(Canon\)](#)

18. Nyomtatók: 3D nyomtatók

Video: [A 3D nyomtató működése \(Objet\)](#)

Video: [A 3D nyomtató bemutatása \(Isis One\)](#)

Video: [A 3D nyomtató működése \(Carbon3D\)](#)

Video: [A 3D nyomtató bemutatása \(Carbon3D\)](#)

Video: [A 3D nyomtató bemutatása \(Carbon3D\)](#)

19. Nyomtató: ZINK (Zero Ink) Technology

Video: [A ZINK működése](#)

20. Szkenner: A szkennerek technológia működése

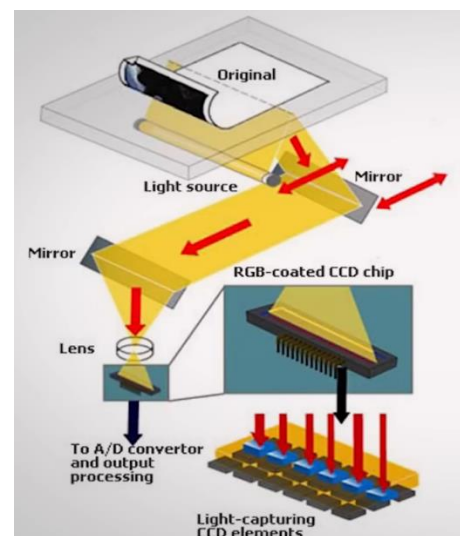
Video: [A lapszkennerek működése](#)

Video: [A vonalkód olvasó működése](#)

Video: [A vonalkód rendszer](#)

Video: [Az UPC-A vonalkód](#)

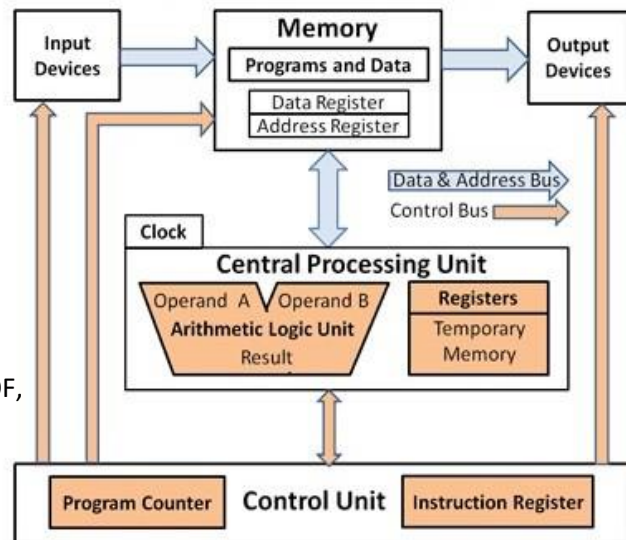
Video: [A QR \(Quick Response\) kód](#)



Kilencedik fejezet

1. A CPU részei:

- ALU – Aritmetikai és logikai egység (Arithmetic and Logic Unit)
- CU – Vezérlő egység (Control Unit)
- Regiszterek
 - Utasításszámláló (Program Counter PC)
 - Utasításregiszter (Instruction Register IR)
 - Állapotregiszter család (Flag Register Family Carry Flag CF, Parity Flag PF, Auxiliary Carry Flag AF, Zero Flag ZF, Sign Flag SF, Trap Flag TF, Overflow Flag OF, Interrupt Enable Flag IF.)
 - Veremmutató (Stack Pointer SP)



2. Az LMC „Little Man Computer” modell (harmadik diáor [06_ch06_hun_v01_hefop_v03x.ppt])

- Az LMC felépítése

Két, klasszikus értelemben vett regiszter funkciót ellátó részegység van, az utasítás számláló (Program Counter) illetve a számológép aktuális tartalma (Accumulator).

 - Be- és Kimeneti kosár (In Basket/Out Basket): Kapcsolat I/O eszközökkel
 - Számológép (Calculator): ALU
 - Postaláda (Mailboxes): Operatív memória
 - Reset: „Meleg” Reset
- Az LMC utasítás készlete
 - gépi kód – csak számokból (Op Code) áll, abszolút gépközele szemléletű
 - assembly – néhány betűs rövidítésekből (Mnemonic) áll, emberibb szemléletű

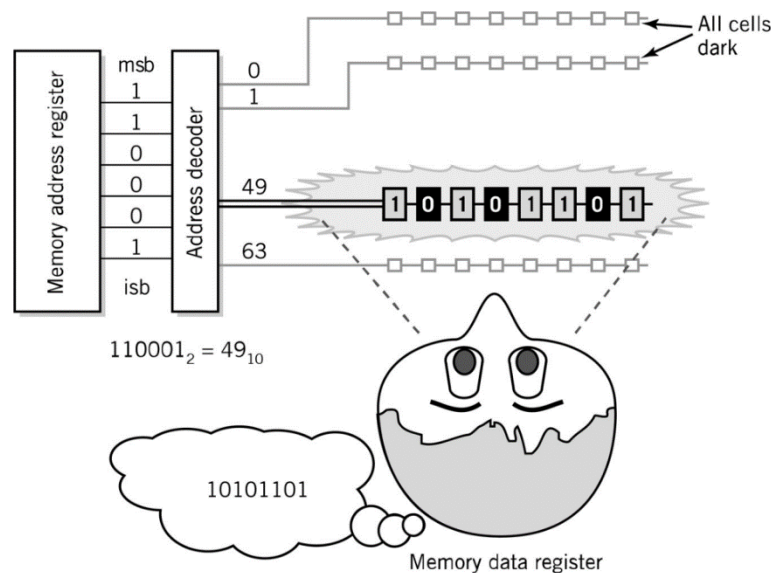
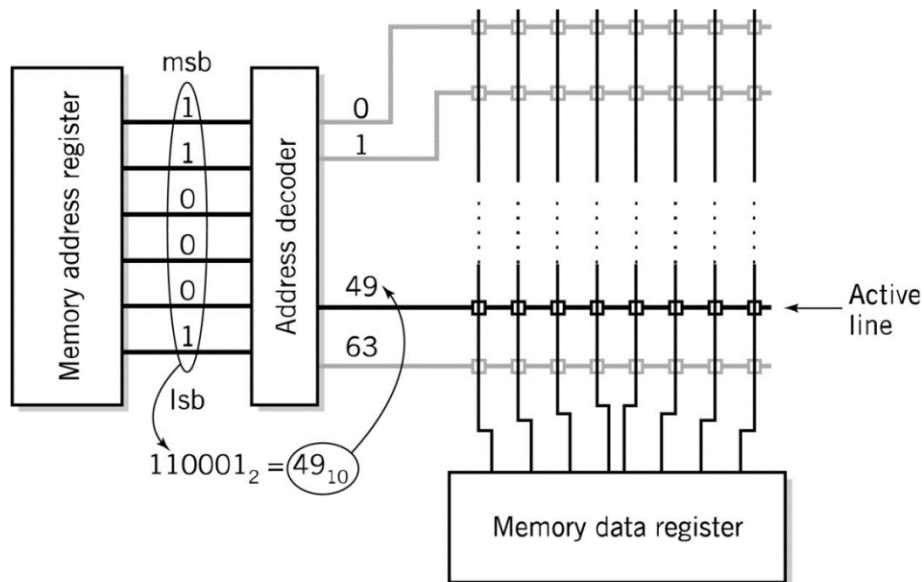
Mnemonic	op code	Description
ADD	1xx	Add
SUB	2xx	Subtract
STO	3xx	Store
LDA	5xx	Load
BR	6xx	Branch
BRZ	7xx	Branch on zero
BRP	8xx	Branch on positive
IN	901	Input
OUT	902	Output
HLT	000	Halt or Stop
DAT		Data storage location

- Az LMC utasítás végrehajtásának fázisai
 - Fetch – Előkészítés
 - Execute – Végrehajtás

3. A memória tartalmának elérése

A memóriában tárolt adatok eléréséhez, és az adatok CPU-ba történő eljuttatásához két regiszter használatára van szükség. A címregiszter, a MAR (Memory Address Register) segítségével választjuk ki a tárolóhelyet. Az adatregiszter, az MDR (Memory Data Register) segítségével olvassuk ki a tárolóhely tartalmát.

Az ábrán egy 6 bites MDR segítségével 2^6 , azaz 64db 8 bites tárhelyet címezhetünk meg. Az 110001_2 , azaz a 49. tárhely kiválasztása után kerül be az MDR-be az ott eltárolt érték: 10101101_2 , azaz 173.



4. A Scott CPU modell

Video: [A CPU működése \(Scott modell\)](#)

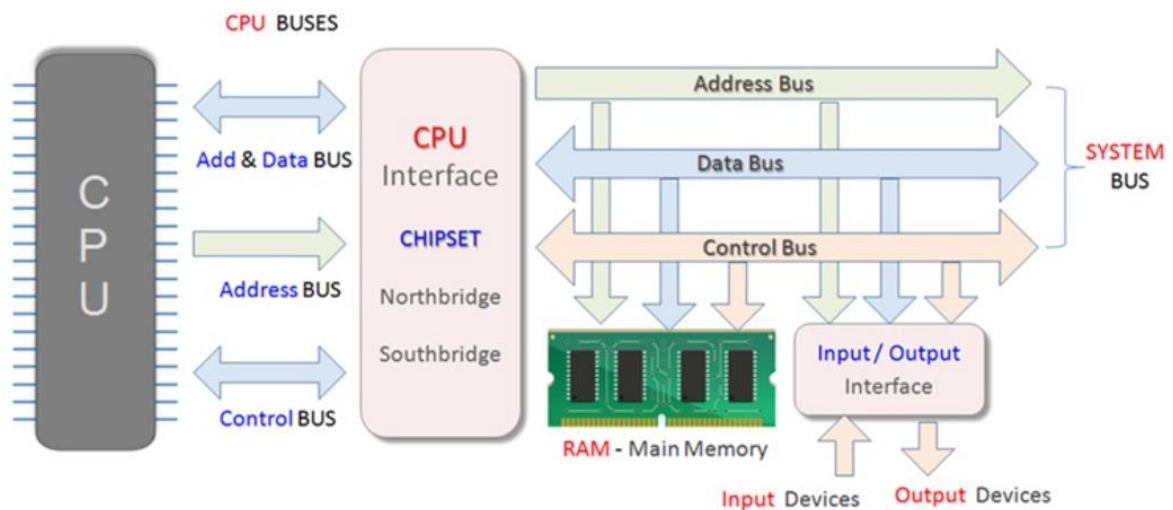
Tizedik fejezet

1. Buszok, Buszrendszerek

- Tápellátó busz (Power Bus)
- Címbusz (Address Bus)
- Adatbusz (Data Bus)
- Vezérlő busz (Control Bus)

Fő paraméterek

- Adatátviteli kapacitás (bit/s).
- Busz szélessége (bit).
- Busz maximális hossza (m) (Input/Output Interface kapcsolat esetén jellemző).



2. A CPU utasításai

Az utasításkészlet, illetve az utasítások hossza processzor (illetve processzor architektúra) függő.

Az utasítás hossza ráadásul lehet rögzített méretű, vagy változó méretű is.

Az utasítások alapvető formátuma: Műveleti kód + Forrás operandus + Eredmény operandus

- A műveleti kód (vagy operációs kód) az ez elvégzendő feladat kódját tartalmazza.
- A forrás operandus az adott művelethez szükséges adat(ok) memória címét tartalmazza.
- Az eredmény operandus az elvégzett művelet eredményének a memória címét tartalmazza, amennyiben az eredményt elmozgatjuk az akkumulátorból (Accumulator).

3. Utasítástípusok

- adatmozgató utasítások
- aritmetikai utasítások
- logikai utasítások (Boolean)
- egy operandusú utasítások
- bitmanipuláló utasítások (bitenkénti eltolás, elforgatás)
- összetett működésű utasítások (call, return)
- veremtár kezelő utasítások
- összetett adatelemeken végzett utasítások utasítások (SIMD, MMX)
- vezérlő utasítások

4. Szekvenciális végrehajtás – CISC és RISC architektúra

- CISC

Complex Instruction Set Computer (Komplex Utasításkészletű Számítógép)

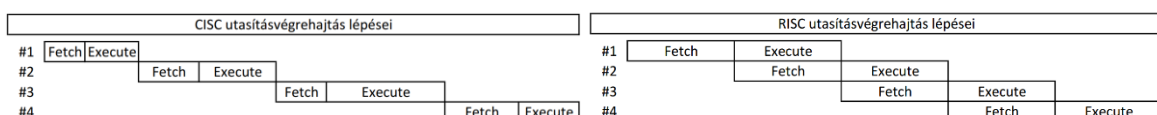
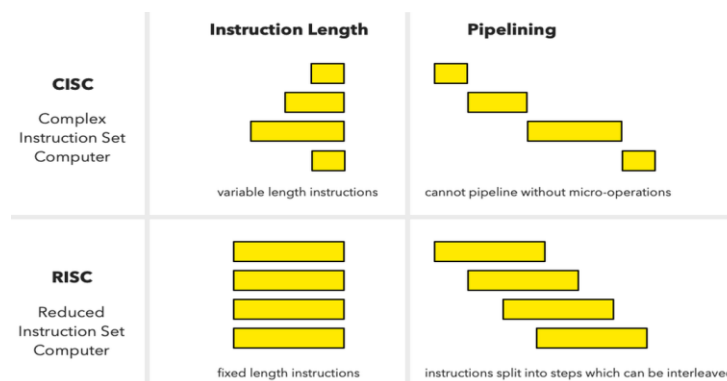
- Memória elérés sokféle utasítás segítségével.
- Kiszámú általános regiszter.
- Több különböző címzési mód az operandusok definiálására.
- Nagy utasítás szám. Komplex, összetett utasítások, melyek végrehajtása akár több gépi ciklust is igénybe vehet.
- Változó méretű utasítás kódok.
- Bonyolult mikroprogram, egyszerű fordítóprogram.

- RISC

Reduced Instruction Set Computer (Csökkentett Utasításkészletű Számítógép)

- Memória elérés csak Load és Store műveletek segítségével.
- Nagyszámú regiszter, jellemzően körkörös szervezésben, ami a hatékony szubrutin hívási eljárásokat segíti.
- Regiszter orientált utasításkészlet, azaz olyan utasítások, melyek operandusai a regiszterekben vannak.
- Kiseb utasításszám és egyszerűbb utasítások (kb. 50-80 utasítás).
- Fix méretű utasítás kódok.
- Hatékony Pipelining: Egy utasítás Fetch fázisa párhuzamosan tud az előtte lévő utasítás Execution fázisával végrehajtódni.
- Korlátozott számú címzési mód.
- Egyszerű mikroprogram, bonyolult fordítóprogram.

Azonos feladat megoldásához a CISC processzornak kevesebb utasítást kell végrehajtania, míg a RISC processzornak ugyan több utasítást kell végrehajtania, de azok viszont egyenként rövidebbek. A vezérlési struktúra meghatározza a processzorok struktúráját is. Míg a CISC processzorok esetében a mikroprogramozott (azaz szoftveres) vezérlés a jellemző, addig a RISC processzoroknál a huzalozott (vagyis hardveres, logikai kapu szintű) vezérlést alkalmazzák. A CISC processzorok esetében az utasítások feldolgozása egy mikroprogrammal megvalósított, értelmezős interpreter segítségével történik, viszont a RISC processzorok esetében erre nincs szükség, ugyanis a felhasználói program eleve egy egyszerű utasításokból álló gépi kódra fordítódik.



5. Párhuzamos végrehajtás – VLIW és EPIC architektúra

- VLIW

Very Long Instruction Word (Nagyon Hosszú Utasítás Szó)

- alapja a 128 bites Crusoe architektúrájú CPU család, mely 64db 32 bites általános célú regisztert tartalmaz
- a 128 bites CPU utasítás csomagok 32 bites utasítás csoportokból épülnek fel
- a CPU 128 bites utasítás csomagokat kap, melyek 4x32 bites elemi utasításokból állnak, így a 128 bites csomag tartalma, a 4db 32 bites utasítás párhuzamosan hajtódik végre

- EPIC

Explicitly Parallel Instruction Computer

(Közvetlenül Párhuzamosított Utasításkészletű Számítógép)

- alapja a 128 bites IA64 architektúrájú CPU családok (Itanium), melyek 128db 64 bites általános célú, és 128db 82 bites lebegőpontos regiszter tartalmaznak [az IA64 család processzorai képesek az x86 család utasításainak végrehajtására is]
- a 128 bites CPU utasítás csomagok 41 bites utasítás csoportokból és egy 5 bites kódból épülnek fel, ami a 41 bites utasítás csoportok típusát tartalmazza
- a CPU 128 bites utasítás csomagokat kap, melyek 3x41 bites elemi utasításokból állnak, így a 128 bites csomag tartalma, a 3db 41 bites utasítás párhuzamosan hajtódik végre, a kiegészítő 5 bites típusjelzés segítségével

A fenti architektúrák alapja az ILP (Instruction-Level Parallelism), azaz az utasítás szintű párhuzamosítás. A buszok szélességének növekedése azt eredményezte, hogy egy memóriaszóban már több utasítás is elfér, melyek így egyetlen gépi ciklus alatt beolvashatók. Az egymást követő utasításokat úgynevezett „utasítás csoportokba” kell szervezni olyan módon, hogy az egyes utasítás csoportokban lévő utasítások párhuzamosan legyenek végrehajthatók. Az azonos elvi alapokon túl a gyakorlati különbséget a csoport szervezési mód határozza meg.

Az alapvetően szekvenciális utasítások párhuzamosított végrehajtásának két elvi akadálya van, az adatfüggés illetve a vezérlésfüggés. Amennyiben ezen függések bármelyike fennáll, akkor a fennállás idejére le kell mondanunk a párhuzamos végrehajtásról. Miről is van szó:

- Adatfüggés:
A párhuzamosítani kívánt egyik utasítás fel szeretné használni a másik utasítás eredményét, így kénytelen kivárni annak befejeződését.
- Vezérlésfüggés:
A párhuzamosítani kívánt egyik utasítás elágazást tartalmaz, aminek a végrehajtása a másik utasítás eredményétől függ, így kénytelen kivárni annak befejeződését.

6. Szuperskalár processzorok

A szuperskalár processzorok legfontosabb jellemzője az, hogy egy gépi ciklus alatt több utasítást is képesek végrehajtani. Ellentétben az VLIW és az EPIC által használt párhuzamosítási technikákkal, a szuperskalár processzor saját Pipelining megoldást használ. A fordítóprogram a beérkező utasításfolyamot kielemezi, és amennyiben az adatfüggés és/vagy vezérlésfüggés nem korlátozza, akkor a processzor architektúrájának megfelelő módon úgy rendezi át az utasításokat, hogy az egy utasítás csoportba kerülő utasítások egyszerre legyenek végrehajthatók. Ez a végrehajtási mód bonyolultabb hardver kialakítást igényel, mivel az utasítások ilyen módon történő átrendezése miatt a Pipeline elemeit többszörösen, redundáns módon kell tárolni, illetve a funkcionális egységek (mint például az ALU) több példányban kell, hogy rendelkezésre álljanak. A bonyolultabb hardver kialakítás magasabb energiafogyasztást is eredményez.

