

# 1 INFORMÁCIÓ, INFORMÁCIÓS RENDSZEREK

Kapcsolatunk a külvilággal folyamatos, hiszen mi magunk is a világ részei vagyunk. A világ pedig tele van jelenségekkel, személyekkel, tárgyakkal. Ezek a dolgok pedig kivétel nélkül tulajdonságokkal rendelkeznek. Gondoljunk csak a legegyszerűbb tulajdonságra, a névre. Ahhoz, hogy egy emberre, állatra, tárgyra, eseményre hivatkozni tudjunk, meg kell neveznünk azt. Azután vannak sokkal konkrétabb tulajdonságai is a különféle dolgoknak, melyeket akár meg is mérhetünk. Ilyen lehet például egy ember testmagassága, tömege, lábmérete stb. Az objektumok számunkra fontos mérhető és nem mérhető tulajdonságait szokás *adat*nak nevezni. Ezt úgy is megfogalmazhatjuk, hogy a világ adatok formájában tükröződik számunkra.

Az adatok azonban még önmagukban nem jelentenek semmit. Szükség van egy gondolati folyamatra (tanulás, munka), mely során az adatok között összefüggéseket állapítunk meg. Az így keletkező adatokat és a közöttük lévő összefüggéseket együttesen szokás *ismeret*nek nevezni. Azokat az új adatokat pedig, melyek összefüggéseikkel együtt beépülnek ismereteinkbe, *információnak* nevezzük.

Fontos megjegyezni, hogy az információ minden esetben újdonságot közöl vagy bizonytalanságot szüntet meg.

Arról már volt szó, hogy az emberi tevékenységeket adatokkal jellemezhetjük. Azt viszont még nem vizsgáltuk, hogy ezek az adatok statikusak-e. Könnyű belátni, hogy rendszerint nem azok. Az adatok többsége az idő múlásával folyamatosan változik. Vannak adatok, melyek gyorsabban, mások lassabban. Egy tevékenységre jellemző adatok időbeli változását *információs folyamat*nak nevezzük. Az információs folyamatokat a gazdasági szférában a legcélszerűbb megvizsgálni. Például egy termék gyártása nagyon összetett folyamat. A gyártási folyamatot pedig az anyagkészlet mértékétől az előállított termékkészlet nagyságáig számos adat kíséri. A gazdasági szférában ezeket az adatokat folyamatosan figyelik és feldolgozzák. Ezt az adatgyűjtési és feldolgozási folyamatot nevezzük információs folyamatnak. Az információs folyamat az egyes szervezeteken belül részfolyamatokból tevődik össze. A részfolyamatok között azt a kapcsolatot, mely adatok átadásából és átvételéből áll, *kommunikációnak* nevezzük.

Egy szervezeten belül az egymással kapcsolatban álló információs folyamatokat együttesen *információs rendszer*nek nevezzük. Az információs rendszer egységes keretbe fogja az adatok gyűjtését, feldolgozását, valamint a részfolyamatok közötti kommunikációt.

Az információ előállítható és megsemmisíthető, minden szervezet információt fogyaszt és előállít, a folyamatban az össz-információ mennyisége növekszik.

Az *informatika* egy szervezet (mely lehet élő vagy élettelen, ahol alapfeltétel a rend) és annak információs rendszere között teremt kapcsolatot. Az informatika tehát a szervezetek információs rendszereivel, és a szervezetben zajló információcserékkel foglalkozik, alapvető tudományág.

Meg kell jegyeznünk, hogy a köznapi értelemben gyakran azonosítják az informatikát és a számítástechnikát. Azonban megfigyelhetjük, hogy az adat, információ, információs rendszer, kommunikáció a számítástechnikától függetlenül létező fogalmak. Az viszont tény, hogy számítástechnikának alapvető szerepe van az információs rendszerekben.

A továbbiakban a számítógépek információs rendszerekben betöltött szerepével foglalkozunk. Először a következő kérdésre keressük meg a választ: vajon miért terjedtek el a számítógépek az információs rendszerekben? Ennek alapfeltétele volt, hogy az emberek által megszokott kommunikációs jelrendszert képesek voltunk olyan megjelenési formátumra (mechanikus, elektronikus, mágneses jel) alakítani, melyek segítségével könnyedén megértettük magunkat a számítógéppel.

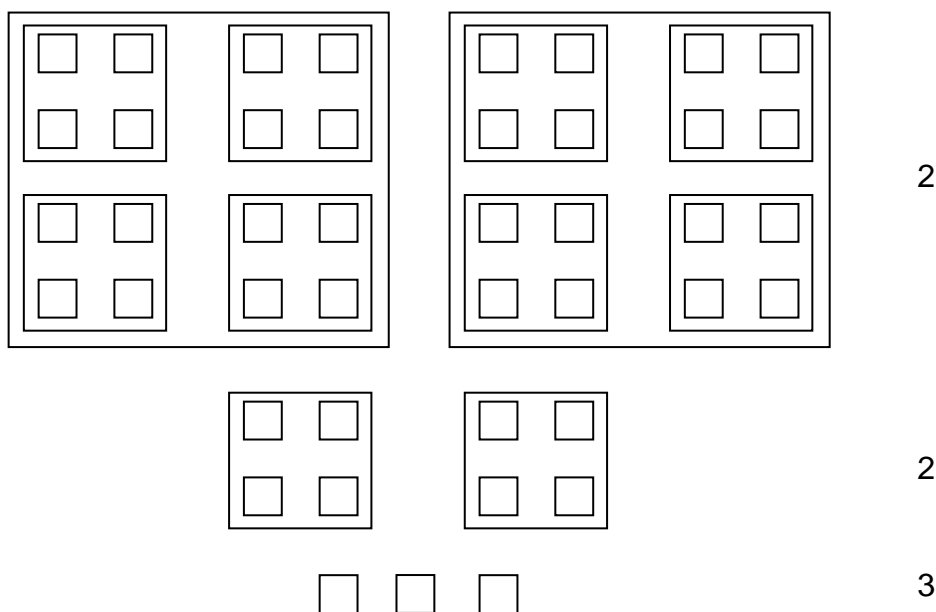
Az adatokat már definiáltuk hétköznapi értelemben. Az adatok számítástechnikai definíciója sokkal konkrétabb: *adat*nak formálisan a számokat, betűket, speciális jeleket és ezek kombinációit tekintjük. Az pedig a számítógép dolga, hogy az „emberi” *adatformátumok* és a saját jelrendszere közötti átalakítást elvégezze.

## 2 A SZÁMÍTÁSTECHNIKA TÖRTÉNETE

### 2.1 A SZÁMRENDSZEREK KIALAKULÁSA

Őseink kitalálták, hogy a számokat célszerű csoportokba rendezni, majd ugyanennyi csoportból csoport-csoportot alakítani és így tovább. Ezek után egy mennyiség megadásakor már elegendő azt megmondani, hogy a különböző szintű csoportból hányat tartalmaz. Ez a csoportosítási mód minden számolás alapja.

Tekintsük például a négyes számrendszert:



2.1. ábra: A négyes alapú számrendszer

A következő nagy felfedezés a helyiértékes írásmód volt, ahol az egymás után írt számoknál az, hogy az utolsó számjegytől hány pozícióval balra, illetve jobbra állnak, azt fejezi ki, hogy a csoportosítás melyik szintjének felelnek meg.

### 2.2 A SZÁMÍTÓGÉPEK FEJLŐDÉSE

A számítógépek története az ókori mechanikus *számolócsontocskákkal*, *kövekkel* kezdődik. Az az eszköz pedig, amelyet még ma is a számolást segítő eszközök őseinek tekintünk, az i.e. IV. évezredben feltalált *abakusz* (golyós számológép), melynek késői változatát még ma is használják egyes országokban.

A 17. században, különösen a hajózás és a navigáció fejlődése szükségessé tette egy olyan eszköz megkonstruálását, mellyel könnyen és gyorsan elvégezhető az egyszerűbb matematikai műveletek. Ez volt a *logarléc*, ami nem más, mint egymáson elcsúsztatható logaritmusos skálák felhasználásával működő matematikai eszköz szorzat, hányados, hatvány, gyök számításához. A logarléc megalkotása William Oughtred nevéhez fűződik.

Ezek az eszközök azonban még nem voltak igazi számológépek, hiszen nem képesek automatikus működésre. De nem kellett sokáig várni, hiszen 1623-ban Wilhelm Schickard elkészítette az első *mechanikus számológépet*, mely a matematikai alapműveleteket már automatikusan végezte.

A ma is használatos *programozható számítógépek* első modelljét Charles Babbage angol matematikus és közgazdász alkotta meg 1822-ben. Ez a szerkezet már szinte akármilyen matematikai feladatot képes volt elvégezni. Babbage a gép elkészítésekor felhasználta a *prog-*

*ramvezérlés* elvét, ami Marye Jacquard nevéhez fűződik, aki 1805-ben forradalmasította a textilipart a műveleti kártyák bevezetésével.

A számítások automatizálásában az elektronika alkalmazása robbanásszerű előrelépést jelentett. Az első *elektromechanikus számítógépeket* Konrad Zuse és Howard Hathaway Aiken készítették.

Az első *elektronikus számítógép* 1946-ban készült el és az ENIAC nevet kapta. Az előkészítő munkálatokban a magyar származású Neumann János is részt vett.

Néhány adat az ENIAC-ról:

- ☪ 18 000 elektroncső
- ☪ 10 000 kondenzátor
- ☪ 70 000 ellenállás
- ☪ 6 000 kapcsoló
- ☪ fogyasztása 800 kW/h
- ☪ 30 méter hosszú, 3 méter széles, 1 méter magas

**2.2. ábra:** *Az első elektronikus számítógép (ENIAC)*



## 2.3 SZÁMÍTÓGÉP-GENERÁCIÓK

Az elektronikus számítógépeket a bennük használt műszaki megoldások alapján generációkba soroljuk. A paraméterek sokfélesége miatt a generációk közötti határvonalak elmosódnak, így nehezen köthetők évszámokhoz. Az alábbi táblázatot is ennek függvényében tekintjük.

Jellemzők	Első generáció (1950-1955)	Második generáció (1956-1964)	Harmadik generáció (1965-1977)	Negyedik generáció (1978-1990)	Ötödik generáció (1991-)
Műveleti sebesség (művelet/s)	10 ezer	1 millió	10 millió	100 millió	1 milliárd
Operatív tárkapacitás (max. byte)	50 k	500 k	10 M	100 M	1 G
Külsőtárkapacitás (max. byte)	10 M	1 G	10 G	100 G	1 T
Eszközök	Rádiócsövek, ferrittár, mágnesdob, mágnesszalag	Tranzistorok, mágneslemez, multiprogramozás	Integrált áramkörök, 16 kbyte-os chippek félvezetőtár, hajlékonylemez, képernyős megjelenítő	LSI áramkörök, 64 kbyte-os chippek, mikroprocesszorok, szuperszámítógépek, távfeldolgozás, gigabyte-os lemezek	VLSI, programozható áramkörök, 256 kbyte-os chippek, optikai táruk
Architektúra	Neumann gépek	Operációs rendszerek	Mikroprogramozott, kompatibilis gépcsaládok, hálózatok	Vektorfeldolgozás, többprocesszoros rendszer, elosztott feldolgozás, lokális hálózatok	Gépi látás, heterogén folyamatszervezés, rekurzió. Kapcsolat az automatikus telekommunikációs és tömegtájékoztatási eszközökkel
Szoftver	Gépi kód, programkönyvtárak	Programozási nyelvek és fordítók, szoftver megjelenése	Adatbázisok, alkalmazói programcsomagok, gépi grafika, dialógusrendszerek, programcsomagok, egységes programozási nyelvek	Egységes operációs rendszeren, adatbázison és nyelveken alapuló hordozható szoftver, a programozás szervezése és gépi eszközökkel való támogatása, feladatosztályok specifikációs nyelvei, szövegfeldolgozás, raszteres grafika	Univerzális specifikációs nyelvek, tudásbázisok, képfeldolgozás, intelligens rendszerek (szakértői rendszerek, automatikus eljárás-generálás és programszintézis, természetes nyelvi interfész)
Alkalmazás	Tudományos	Gazdasági rendszerek, gazdasági és mérnöki számítások	Termelésirányítási rendszerek, információs rendszerek, számítógéppel támogatott tervezés, automatizált kísérletezés	Globális tervezési és gyártási rendszerek, irodaautomatizálás, termelésirányítás	Az információs társadalom kezdetei, intelligens robotok, automatizált rendszerek az iparban, oktatásban, egészségügyben

2-1. táblázat: Számítógép-generációk

### 3 A SZÁMÍTÓGÉP MŰKÖDÉSÉNEK ALAPJAI

#### 3.1 NEUMANN ELVEI

A számítógépek felépítésének és működésének Neumann János által megfogalmazott elvei a következők:

1. Az utasításoknak az adatokkal azonos módon (azaz egyetlen nagy kapacitású memóriában), numerikus kódok formájában való tárolása.
2. A kettes számrendszer alkalmazása.
3. Szükséges egy vezérlőegység, amely különbséget tud tenni utasítás és adat között, majd önműködően végrehajtja az utasításokat.
4. A teljesen elektronikus működés.
5. A számítógép tartalmazzon olyan számológyművet, amely képes elvégezni a logikai műveleteket is.
6. Szükség van olyan ki/bemeneti egységekre, amelyek áthidalják a bizonyos szempontból lassú emberi kommunikációs csatorna és a gyors számítógép közti sebességkülönbséget.
7. Megbízhatatlan alkatrészekből is lehetséges megbízható rendszert konstruálni. Ennek a feladatnak egyik lehetséges megoldása a szerkezeti tartalékolás (strukturális redundancia).

#### 3.2 A SZÁMRENDSZEREK

Az köznapi életben rendszerint a 10-es (decimális) számrendszert használjuk, ezért ezt tartjuk természetesnek. A számítógépes adatok, kódok, programok megértéséhez azonban szükségünk van arra, hogy a különböző számrendszerekben szabadon hajtsunk végre műveleteket. Egy tetszőleges alapú számrendszerben a következőképpen tudunk felírni egy számot:

$$\sum_{i=-m}^n a_i \cdot p^i$$

ahol  $p$  jelöli a számrendszer alapját,  $i$  a helyiértéket,  $n$  a pozitív,  $m$  pedig a negatív helyiértékek számát,  $a$  pedig a számjegyeket.

A tízes számrendszerben felírt számoknál nem jelöljük a számrendszer alapját. Azokat a számjegyeket, amelyeket egy adott számrendszerbeli szám felírásánál használhatunk, a számrendszer *jelkészletének* nevezzük. A jelkészlet nagyságát a számrendszer alapja egyértelműen meghatározza. A  $p$  alapú számrendszer jelkészlete a  $0, 1, 2, \dots, p-1$  számjegyekből áll. Ha  $p > 10$ , akkor a szokásos számjegyekből nem állítható össze a teljes jelkészlet, a 9-nél nagyobb számjegyek jelölésére más jeleket, például betűket használunk.

Egy adott számrendszerben, adott számú helyiértéken ábrázolható legnagyobb ( $K$ ) és legkisebb ( $k$ ) szám:

$$K = p^{n+1} - p^{-m}$$

$$k = p^{-m}$$

Példák: Számoljuk át a következő különböző alapú számrendszerbeli számokat tízes számrendszerbe:

1.  $123,12_{(4)} = 1 \cdot 4^2 + 2 \cdot 4^1 + 3 \cdot 4^0 + 1 \cdot 4^{-1} + 2 \cdot 4^{-2} = 27,375$
2.  $123,12_{(5)} = 1 \cdot 5^2 + 2 \cdot 5^1 + 3 \cdot 5^0 + 1 \cdot 5^{-1} + 2 \cdot 5^{-2} = 38,28$
3.  $123,12_{(6)} = 1 \cdot 6^2 + 2 \cdot 6^1 + 3 \cdot 6^0 + 1 \cdot 6^{-1} + 2 \cdot 6^{-2} = 51,22$

### 3.2.1 KETTES (BINÁRIS) SZÁMRENDSZER

A számítógépes adattárolás eszközei legnagyobb részben kétállapotú fizikai elemek (pl. tranzisztorok). Az egyik állapotot megfeleltetjük a nulla számnak, a másikat az egynek. Így egyetlen kétállapotú fizikai elem alkalmas egy egyjegyű, kettes számrendszerbeli szám értékének tárolására. A kettes számrendszerbe tartozó számot bináris számnak, az egyjegyű bináris számot pedig *bit*nek nevezzük. A bit az információ alapegysége.

Tegyük fel, hogy  $n+m+1$  bit áll rendelkezésünkre, és képzeljük a pontot az  $n+1$ . bit után. Ekkor az  $n+m+1$  biten felírható bináris szám:

$$a_n \cdot 2^n + \dots + a_0 \cdot 2^0 + a_{-1} \cdot 2^{-1} + \dots + a_{-m} \cdot 2^{-m}$$

ahol az  $a_n, \dots, a_0, a_{-1}, \dots, a_{-m}$  értéke 0 vagy 1.

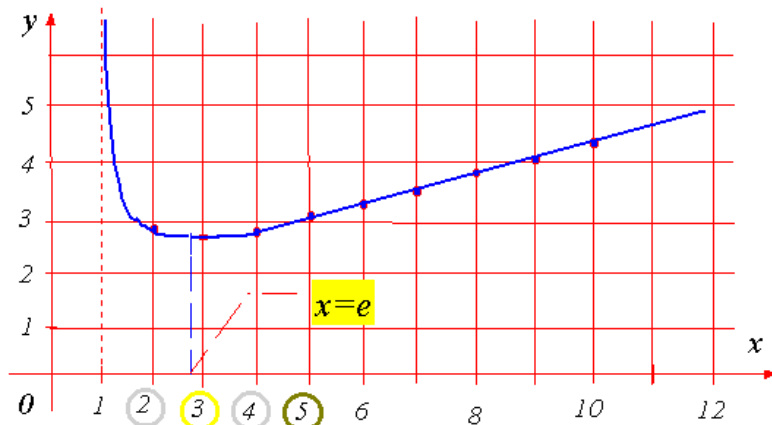
### 3.2.2 MIÉRT JÓ A KETTES SZÁMRENDSZER

Vizsgáljuk meg, hogy a számok ábrázolására melyik a legoptimálisabban alkalmazható számrendszer.

$$S = R \cdot \log_R M = \frac{R \cdot \ln M}{\ln R}$$

ahol  $R$  a radix, vagyis az alapszám,  $\log_R M$  pedig a szükséges számjegyek száma.

Keressük meg  $y = \frac{x}{\ln x}$  minimumát!



3.1. ábra: Szemléltető grafikon

$$y' = \frac{\ln x - 1}{(\ln x)^2} = 0, \text{ ha } \ln x = 1, \text{ vagyis } x = e \text{ és } y = e$$

$$y' = \frac{1}{x(\ln x)^2} \cdot \left( \frac{2}{\ln x} - 1 \right) \quad y'(e) = \frac{1}{e} > 0, \text{ akkor } y \Rightarrow \text{Min.}$$

A legoptimálisabban tehát a természetes alapú számrendszerben tudnánk ábrázolni a számokat, azonban ez a számítást nagyon nehézkesé tenné, és a technikai megvalósítása is rendkívül bonyolult lenne. Ha egész alapú számrendszerekben gondolkodunk, akkor a 3-as számrendszer lenne megfelelő, viszont a számítástechnikában használt kétállapotú elemeknek sokkal inkább megfelel a bináris számábrázolás.

### 3.2.3 TÍZES-KETTES (DECIMÁLIS-BINÁRIS) SZÁMRENDSZERBELI SZÁMKONVERZIÓK

#### a) Az egész rész konvertálása

A decimális egészrészt addig osztjuk kettővel, amíg az osztás eredménye nulla nem lesz. Egy függőleges vonallal képezzünk két oszlopot. A vonal bal oldalán szerepelnek a hányadosok, a jobb oldalán pedig a maradékok.

pl.  $73 = ?_{(2)}$

73	:2
36	1
18	0
9	0
4	1
2	0
1	0
0	1



A kiolvasás iránya

Tehát  $73 = 1001001_{(2)}$

**b) A törtrész konvertálása**

A törtrészt addig szorozzuk kettővel, amíg a szorzás eredménye 1,0 nem lesz. Ez ritkán következik be, ezért a végtelen, szakaszos kettedes tört ismétlődő szakaszának felismeréséig, vagy pedig az ábrázolható helyiértéktől függő oszlophosszúsáig folytatjuk a szorzást. Itt is két oszlopot készítünk egy függőleges vonallal. A vonal bal oldalán szerepelnek a szorzás egész részei, a jobb oldalán pedig a törtrészei. A szorzást csak a törtrésszel folytatjuk.

1. pl.  $0,125 = ?_{(2)}$

2·	0,125
0	,25
0	,5
1	0



A kiolvasás iránya

Tehát  $0,125 = 0,001_{(2)}$

2. pl.  $0,63 = ?_{(2)}$

2·	0,63
1	,26
0	,52
1	,04
0	,08
0	,16
0	,32
0	,64
1	,28



A kiolvasás iránya

Tehát  $0,63 = 0,10100001_{(2)}$

3. pl.  $0,4 = ?_{(2)}$

2·	0,4
0	,8
1	,6
1	,2
0	,4
0	,8



A kiolvasás iránya

Ismétlődés

Tehát  $0,4 = 0,011\dot{0}_{(2)}$

**Kettes számrendszerben ez végtelen szakaszos tört, tehát elvileg végtelen sok hely kell a tárolásához.**

**3.2.4 BINÁRIS SZÁMOK ÁTALAKÍTÁSA DECIMÁLIS SZÁMOKKÁ**

Horner-séma polinomok helyettesítési értékének meghatározására:

$$p_n(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \dots + a_2 \cdot x^2 + a_1 \cdot x + a_0 =$$

$$= x(a_n \cdot x^{n-1} + a_{n-1} \cdot x^{n-2} + \dots + a_2 \cdot x + a_1) + a_0$$

Például  $p_3(x) = 2x^3 + 5x^2 - 3x + 4$  és  $x_1 = 3,14$ . Mekkora  $p_3(x_1)$  értéke?



$$p_3(3,14) = 3,14[3,14(3,14 \cdot 2 + 5) - 3] + 4 = 3,14[3,14(6,28 + 5) - 3] + 4 =$$

$$= 3,14(3,14 \cdot 11,28 - 3) + 4 = 3,14 \cdot 32,4192 + 4 = 105,79629$$

A bináris egészrész átalakításakor  $x_1=2$ , a törtrésznél pedig  $x_2=1/2$  és  $a_0=0$ .

2. pl.:  $1001011_2=?$

1	0	0	1	0	1	1
2	4	8	18	36	74	<b>75</b>

Az utolsó összeget már nem kell megszorozni 2-vel.

2. pl.:  $0,1010011_2=?$

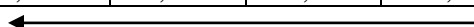
1	1	0	1	0	0	1
0,5	0,75	0,375	0,6875	0,34375	0,171875	<b>0,5859375</b>

Az utolsó összeget is el kellett osztani 2-vel.

3. pl.:  $1011,0101_2=?$

1	0	1	1	,	0	1	0	1
2	4	10	11	11,3125	0,3125	0,625	0,25	0,5

A törtrésznél hátulról kezdjük az osztást.



### 3.2.5 HEXADECIMÁLIS SZÁMOK

Az ember számára a kettes számrendszerben felírt szám nem eléggé áttekinthető, túl hosszú, ezért bevezetjük a binárishoz közeli hexadecimális (tizenhatos alapú) számrendszert, ami megkönnyíti a bináris számok kezelését. A hexadecimális számrendszer számjegyei növekvő sorrendben a következők:

0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

A betűk a 9-nél nagyobb számjegyek ábrázolására szolgálnak, tízes számrendszerbeli megfelelőjük:

A=10; B=11; C=12; D=13; E=14; F=15

Ha egy tetszőleges bináris számot négyes csoportokra osztunk az egészeket a törtektől elválasztó ponttól balra és jobbra indulva, a négyes csoportoknak megfeleltethetünk egy-egy hexadecimális számjegyet.

Ha a négyes csoportokra osztásnál a törtészt jelölő ponttól jobbra vagy balra lévő csoport négyenél kevesebb tagból áll, jobb oldalon a szám után, bal oldalon a szám elé annyi nullát képzelünk, hogy a csoport négy jegyből álljon.

Bináris	Hexadecimális
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

3-1. táblázat: A bináris és a hexadecimális számok közötti közvetlen megfeleltetés



Példa:

		+10	+10						+10		
	1	1	0	1	1	0	1	.	1	0	1
-		1	1	1	1	0	1	.	0	1	0
		1	1	0	0	0	0	0	0	1	1

- 1-0=1            leírjuk az 1-et.
- 10<sub>(2)</sub>-1=1      leírjuk az 1-et.
- 1-1-0=0        leírjuk a 0-t.
- 1-1=0           leírjuk a 0-t.
- 0-0=0           leírjuk a 0-t.
- 1-1-0=0        leírjuk a 0-t.
- 1-1-0=0        leírjuk a 0-t.
- 10<sub>(2)</sub>-1=1      leírjuk az 1-et.
- 11<sub>(2)</sub>-1-1=1    leírjuk az 1-et.
- 1-1-0=0        leírjuk a 0-t.

Mivel a + vagy – előjel kiírására a gépi számábrázolásnál nincs lehetőség, a pozitív és a negatív számokat más módon kell megkülönböztetni. Legegyszerűbb lenne a szám valamelyik bitjét – pl. az első – előjelbitnek kinevezni, pozitív számok esetén az értéke 0 lehetne, negatív számok esetén 1. Az, hogy mégsem mindig ezt az egyszerű eljárást alkalmazzák a számítástechnikában, arra vezethető vissza, hogy a negatív számokat olyan formában célszerű tárolni, hogy a gépi kivonás művelete összeadással legyen helyettesíthető. Minthogy a szorzás összeadások sorozatára, az osztás kivonások sorozatára vezethető vissza, ha sikerül a kivonást összeadásra visszavezetni, akkor a számítógépnek csak egyetlen aritmetikai műveletet, az összeadást kell tudnia elvégezni.

Nézzük meg, hogy a negatív számok speciális ábrázolásával hogyan vezethető vissza a kivonás összeadásra. Az egyszerűség kedvéért térjünk át a tízes számrendszerbe. Tegyük fel, hogy a számok ábrázolására három pozíció áll rendelkezésünkre (a legnagyobb ábrázolható szám  $10^3-1=999$ ).

Vegyünk egy tetszőleges háromjegyű számot, pl. a 234-et. Vonjuk ki 999-ből, az eredmény 765. A 765 az a szám, amely a 234-et csupa kilencesekből álló számra egészíti ki három pozíción. Ezt a számot a 234 kilences komplementének (kiegészítőjének) nevezzük. Általában beszélhetünk a p alapú számrendszerben egy szám p-1-es komplementéről, amely a számot olyan n pozíciójú számmá egészíti ki, amelynek minden jegye p-1. Adjunk hozzá a 765-höz 1-et, az eredmény 766.

A 766-ot a 234 tízes komplementerének nevezzük. Általában azt a számot, amely a p alapú számrendszerben egy n pozíción felírható számot p-re egészít ki, a szám p-s komplementének nevezzük. Láttuk, hogy a p-1-es komplementből a p-s komplementet 1 hozzáadásával kapjuk.

Végezzük el az alábbi kivonást a hárompozíciós decimális számok körében:  

$$723-132=591$$

Ezt a kivonást az alábbi egyenlőség felhasználásával másképp is elvégezhetjük:  

$$723-132=723+(1000-132)-1000$$

Lépésenként végrehajtva, először képezzük a 132 tízes komplementét három pozícióra, ez 868. Ezt hozzáadjuk a 723-hoz, az eredmény:

$$723+868=1591$$

A kapott számból kivonunk ezret, azaz elhagyjuk az első számjegyet, amit a számítógép automatikusan le is vágna, hiszen a szám ábrázolására csak három pozíció áll rendelkezésre. Így a kivonást két műveletre, egy p-s komplementképzésre és egy összeadásra vezettük vissza.

Kettes számrendszerben a p-1-es komplement nem más, mint az 1-es komplement, így a komplementképzéskor a nullákhoz egyet, az egyekhez nullákat kell adnunk, hogy minden

pozíción 1-et kapjunk. A kettes komplement képzése meghatározás szerint egy egyes hozzáadásával valósul meg.

Példa:

11001110 – nyolcpozíciós bináris szám

00110001 – a szám egyes komplemente

00110001 – egyes komplement

+00000001

00110010 – kettes komplement

### 3.3 ADATÁBRÁZOLÁS A SZÁMÍTÓGÉPEN

A számítógép fizikai felépítése miatt minden adat végső gépi formája egy bitsorozat. A számítógépes táruk logikailag legkisebb egysége a nyolc bináris számjegy elhelyezésére alkalmas byte. Egy byte-on  $2^8=256$  egymástól különböző bitsorozat helyezhető el.

A számítógépben alapvetően kétféle módon tárolják az adatokat. Az egyik mód a számok műveletvégzésre alkalmas formájú tárolása, az ún. *gépi számábrázolás*, míg a másik a tetszőleges betűkből, számokból, írásjegyekből és egyéb jelekből álló információ, a *kódolt ábrázolás*.

A számítógépen elvileg kétféle számtípust különböztetünk meg, az egész és a valós számokat. Ezeket az elnevezéseket itt nem a megszokott matematikai értelemben használjuk. A gépi egészeket interpretálhatjuk úgy is, mint olyan bináris számrendszerben felírt egész és tört részből álló számot, amelyben a törtet jelentő pontot valamely két számjegy közé odaképzeltük. A gép a pontot nem tárolja sehol, azt a programozónak kell elképzelnie. Ezt a számábrázolást *fixpontos* ábrázolásnak is nevezzük.

A gépi valós számok ún. hatványkitevős alakjának gépi ábrázolásai. Itt a törtpont helye a számon belül nem fix, ezért ezt a számábrázolást *lebegőpontos*nak nevezzük.

#### 3.3.1 A FIXPONTOS GÉPI SZÁMÁBRÁZOLÁS

A számítógépeken a fixpontos számokat általában két byte-on vagy 4 byte-on ábrázolják, azaz egy szám hossza 16, illetve 32 bit. Ha csak nemnegatív számokat kellene ábrázolni, 16 biten a legkisebb ábrázolható szám a nulla, a legnagyobb  $2^{16}-1$  lenne, míg 32 biten  $2^{32}-1$ .

Ábrázoljuk például a  $-111_{(2)}$  számot 16 biten kettes komplement formában:

0000000000000111	– az eredeti szám
1111111111111000	– egyes komplement
1111111111111001	– kettes komplement

Ha ezt a kettes komplement hozzáadjuk egy tetszőleges 16 bites számhoz, az eredmény ugyanaz, mintha 111-et kivontunk volna belőle, ugyanis a legnagyobb helyiértékű bit egyszerűen elvész. E tulajdonságok miatt a negatív számok helyett azok kettes komplementjét tároljuk.

Hogyan tudjuk eldönteni például azt, hogy az 1010101010101010 szám pozitív-e, vagy egy másik szám kettes komplemente? A felírás alapján sehogyan. Ezért megegyezünk abban, hogy a legnagyobb ábrázolható pozitív szám  $2^{15}-1$  legyen, vagyis a nemnegatív számok csak a 15 alsó helyiérték bitet foglalják el, a negatív számoknak megfelelő kettes komplementek első, legnagyobb helyiértékű bitje pedig 1 legyen.

A pozitív számok így

0000000000000000 és 0111111111111111

között, míg a negatív számok komplementjei

1000000000000000 és 1111111111111111

között helyezkednek el. Átszámolva decimális számokká a két byte-os egész értéke:

$-32768 \leq I \leq 32767$  lehet.

Ebben a számtartományban a pozitív számok legmagasabb helyiértékű bitje mindig 0, a negatívoké 1. Így ezt a bitet előjelbitnek is nevezik.

### 3.3.2 LEBEGŐPONTOS SZÁMÁBRÁZOLÁS

A lebegőpontos számábrázolás a számok hatványkitevős felírásán alapszik. A szám hatványkitevős alakját egy egész és egy törtreszből álló szám, valamint a számrendszer alapja hatványának szorzataként írhatjuk fel. Általánosan az  $M \cdot p^k$  alakban.

Természetesen bármely számot végtelen sok módon írhatunk fel hatványkitevős alakban, hiszen például:

$$11,1 = 0,111 \cdot 10^2 = 0,0111 \cdot 10^3 = 0,00111 \cdot 10^4 \text{ stb.}$$

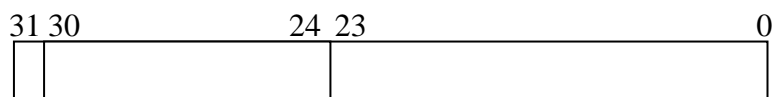
Ahhoz, hogy a hatványkitevős formát egyértelművé tegyük, elfogadunk egy közös elvet, nevezetesen azt, hogy az  $M$  mindig kisebb kell, hogy legyen, mint 1, valamint a törtponttól jobbra álló első jegy nem lehet 0. A  $p$  alapú számrendszerben tehát az  $M$ -re vonatkozó korlátozások:

$$1/p \leq M \leq 1$$

Tíz-es számrendszerben például  $0,1 \leq M \leq 1$ , kettesben  $0,5 \leq M \leq 1$ . Ha  $M$  eleget tesz a fenti korlátozásnak, a szám normál formájú.  $M$ -et a szám *mantisszájának*,  $k$ -t pedig a szám *karakterisztikájának* nevezzük.

A lebegőpontos szám a számítógépen a bináris számok hatványkitevős alakja. A bináris számot először normál formájúvá kell átalakítani, majd a szám részére kijelölt tárrekeszekben ábrázolni. Egy számról négy adatot kell megőriznünk ahhoz, hogy értékét vissza tudjuk nyerni – a mantissza előjelét (a szám előjele), a mantissza abszolút értékét, a kitevő előjelét és a kitevő abszolút értékét. Ennek a négy adatnak a tárolását a különböző számítógépeken különféle módon oldják meg. Tekintsük át a két legáltalánosabban elterjedt megoldást!

A szám ábrázolása 4 byte-on, azaz 32 biten történik. Ábrázoljuk a 32 bitet, 0 sorszámú a legalacsonyabb, 31 pedig a legmagasabb helyiértékű.



3.2. ábra: Fixpontos számok ábrázolása 32 biten

A 31. bit mindig a szám előjelét tartalmazza. A 0 jelenti a pozitív, az 1-es a negatív előjelet. A 24-30 bitek tartalmazzák az előjeles kitevőt, míg a 0-23 bitek a mantisszát. A mantissza képzésénél tér el a kétféle megoldás. Az IBM gépcsaldnál a normálás előtt a bináris számot hexadecimális formába képezzük le, és így végezzük el a normálást, tehát a törtpont utáni első hexadecimális jegynek kell nullánál nagyobbobbnak lennie, míg a PDP típusú gépeknél közvetlenül a bináris számot normálja a gép, azaz az első bináris törtjegynek kell nullánál nagyobbobbnak lennie.

Lássunk néhány példát!

$$11010010.1011_{(2)} = 0.110100101011_{(2)} \cdot 2^8 =$$

a mantissza    a hatványkitevő  
 a binárisan normált alak

$$= D2.B_{(16)} = 0.D2B_{(16)} \cdot 16^2$$

a mantissza    a hatványkitevő  
 a hexadecimálisan normált alak

A binárisan normált alak mantisszája tehát 110100101011, a hexadecimálisan normálté 110100101011, tehát ugyanaz, a kitevő az első esetben (7 biten) 0001000, a második esetben (ugyancsak 7 biten) 0000010.

Nézzünk most egy olyan példát, ahol a mantisszák nem azonosak:

$$111.01_{(2)} = 0.11101_{(2)} \cdot 2^3 =$$

a mantissza    a hatványkitevő  
 a binárisan normált alak

$$= 7.4_{(16)} = 0.74_{(16)} \cdot 16^1$$

a mantissza    a hatványkitevő  
 a hexadecimálisan normált alak

A binárisan normált alak mantisszája tehát 11101, a hexadecimálisan normálté pedig 011101.

Bináris normálásnál tudjuk, hogy a mantissza legmagasabb helyiértékű jegye mindig 1, hiszen ez a normálás feltétele. Így ezt a bitet – azaz a 23-as bitet – nem használjuk a mantissza ábrázolására, hanem összevonjuk a 24-30 bitekkel, és a karakterisztika ábrázolására a 23-30-as részt vesszük igénybe. Általánosan alkalmazott módszer, hogy a szám kitevőjének értékét hozzáadjuk az 10000000 bináris konstanshoz, és az összeget tároljuk a 8 biten.

Például, ha a kitevő -2:

$$-3_{(10)} = -11_{(2)}, \text{ a tárolt érték a 23-30 biten:}$$

$$10000000 - 00000011 = 01111101$$

Ezt a kitevő-tárolási módot feszített előjeles, vagy alapértékes tárolásnak is nevezzük. A bináris 10000000 decimális értéke 128, a pozitív kitevő hozzáadódik a 128-hoz, míg a negatív kitevő levonódik belőle.

Hexadecimális normálásnál nincs mód arra, hogy a mantissza első bitjét más célra használjuk, az lehet nulla is. A karakterisztika ábrázolására tehát marad 7 bit, a 24-30-as rész. Hasonlóan a bináris normáláshoz, itt is az 1000000 számhoz adjuk hozzá a tényleges kitevőt, azaz itt is feszített előjeles eljárást alkalmazunk.

Bináris normálásnál a mantissza értéke 0,5 és 1 közé esik, a kitevőé pedig -128 és 127 közé. Az első esetben a kitevő bitek értéke nulla, a másodikban 1111111. Így a legkisebb és a legnagyobb szám, amely ezzel az eljárással ábrázolható:

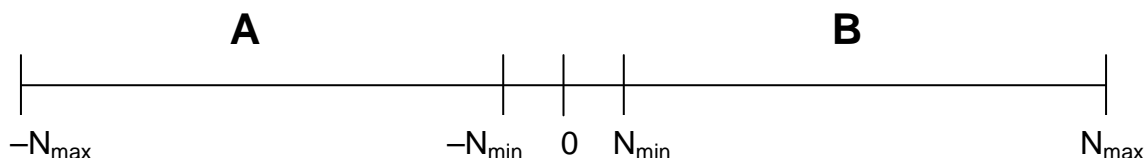
$$N_{\min} = 0,5 \cdot 2^{-128}$$

$$N_{\max} = 1 \cdot 2^{127}$$

Hexadecimális normálásnál a mantissza legkisebb értéke  $1/16 = 2^{-4}$  (amikor a mantissza 0001-gyel kezdődik), a kitevő szélső értékei pedig -64 és +63. Ennek megfelelően a számábrázolás határai:

$$N_{\min} = 0,0625 \cdot 16^{-64}$$

$$N_{\max} = 1 \cdot 2^{63}$$



3.3. ábra: Lebegőpontos számok ábrázolási tartománya

A lebegőpontos számábrázolásnál tehát az  $A+B$  számtartomány használható számábrázolásra. A  $-N_{\max}$ -nál kisebb, a  $-N_{\min}$  és  $N_{\min}$  közé eső, valamint az  $N_{\max}$ -nál nagyobb számok nem ábrázolhatók. Ha egy művelet eredménye  $-N_{\min}$  és  $N_{\min}$  közé esik *alulcsordulásról*, ha  $-N_{\max}$ -nál kisebb, vagy  $+N_{\max}$ -nál nagyobb, *túlcordulásról* beszélünk. A programfutás során fellépő alul-, vagy túlcordulás kezeléséről a programozónak kell gondoskodnia. A pontosság növelésére a számítógépek lehetőséget adnak arra, hogy a lebegőpontos szám több mint négy byte-ot foglaljon el. Sok gép megenged hosszú (*long*) vagy duplapontosságú (*double*) lebegőpontos ábrázolást is.

## 3.4 DECIMÁLIS SZÁMOK, KÓDRENDSZEREK

### 3.4.1 DECIMÁLIS SZÁMOK A SZÁMÍTÓGÉPEN

A gazdasági jellegű feladatok megoldására is használt programozási nyelvek a fixpontos és lebegőpontos ábrázolás mellett közvetlenül is képesek aritmetikai műveleteket végezni decimális számokkal.

A decimális számábrázolás számítógéptől függ. A két legelterjedtebb típus az *ASCII kódú* ábrázolás (pl. PDP<sup>1</sup> gépek) és a *BCD ábrázolás* (pl. IBM gépek).

A PDP gépeken többféle decimális szám létezik, egyes fajták csak *beviteli/kiviteli*, más formátumok aritmetikai célokat szolgálnak. Az ún. számsorok a tízes számrendszerbeli számok közvetlen ASCII kódú megadásai, ahol minden byte egy számjegyet tartalmaz. Ebből a típusból kétféle létezik aszerint, hogy a szám előjele külön byte-ot foglal el a szám előtt, vagy az előjel be van építve a számba. Az első esetben a szám első byte-jának tartalma hexadecimális 2D (mínusz), vagy 2B vagy 20 (plusz), azaz egy  $n$  jegyű szám  $n+1$  byte-ot foglal el.  $n$  megengedett maximális értéke 31. A beépített előjelű decimális számban az előjel az utolsó számjeggyel közös byte-ban jelenik meg, a többi számjegy változatlanul ASCII kódú. Itt is két forma létezik, az ún. *első zónás* és *második zónás formátum*.

<sup>1</sup> A Digital Equipment Co. számítógép gyártó cég (USA) nagysikerű sorozata. Eredetileg folyamatszabályozásra fejlesztették ki, de később széles körben elterjedt az adatfeldolgozási célú alkalmazása is.

Az utolsó byte értékét az alábbi táblázat mutatja:

az előjel és az utolsó jegy	1. zónás forma	2. zónás forma
+0	30	7B
+1	31	41
+2	32	42
+3	33	43
+4	34	44
+5	35	45
+6	36	46
+7	37	47
+8	38	48
+9	39	49
-0	70	7D
-1	71	4A
-2	72	4B
-3	73	4C
-4	74	4D
-5	75	4E
-6	76	4F
-7	77	50
-8	78	51
-9	79	52

**3-2. táblázat:** Decimális számok ábrázolása ASCII kódrendszerben

Ennél a számábrázolásnál ugyanannyi byte kell a számhoz, ahány számjegyből áll. Ez bizonyos fokig luxus, hiszen egy decimális számjegy ábrázolására 4 bit elegendő. Ezt használja ki az ún. *tömörített számábrázolás*. A tömörített formájú decimális számnál két számjegy kerül egy byte-ra. Az előjel az utolsó félbyte-ra kerül, mégpedig oly módon, hogy a hexadecimális A, C, E és F a „+” jelet, a B és D a „-” jelet reprezentálja. Általában a C és D érték használatos. Ha a szám páros számú számjegyből áll, nem marad hely az előjel számára, ugyanis a szám páros számú félbyte-ot tölt ki. Ezt a problémát úgy oldják meg, hogy a szám elé egy zérót írnak.

Példa: -1234

külön előjelbyte-tal: 2D31323334  
 beépített előjellel:  
 első zónás forma: 31323374  
 második zónás forma: 31323334D  
 tömörített forma: 01234D

Az IBM gépeken is két fajtája létezik a decimális számoknak, az ún. zónás hosszabb forma, ahol minden számjegynek külön byte felel meg, és a tömörített, rövidebb alak. A zónás alakban egy byte alacsonyabb helyiértékű félbyte-ja tartalmazza a számjegy bináris alakját, a felső félbyte tartalma hexadecimális F. A szám előjelét a legalacsonyabb helyiértékű számjegy felső félbyte-jában ábrázoljuk, a „+” jelnek A, C vagy F, a „-” jelnek B vagy D felel meg. A szabvány jelölés: „+”=C, „-”=D.

A tömörített alakban egy byte-on két számjegy helyezkedik el bináris formában. Az előjel az utolsó byte alsó félbyte-jába kerül.

Példa: -1234

Zónás: F1F2F3D4  
 Tömörített: 01234D

### 3.4.2 KÓDRENDSZEREK

A számjegyek, betűk, írásjelek és egyéb jelek (összefoglaló szóval *karakterek*) számítógépes ábrázolására bináris kódokat használunk. Az ábrázolandó karakterek száma



nem haladja meg a 256-ot, az ábrázoláshoz elegendő  $2^8$  lehetőség, azaz egy byte. Nagyon fontos, hogy ugyanaz a bitkombináció ugyanazt a karaktert jelentse mindenütt a világon. Ezt biztosítják a nemzetközi *kódszabványok*. A világon ma két fő számítógépes kódszabvány létezik, az egyik az EBCDIC (Extended Binary Coded Decimal Interchange Code), a másik az ASCII (American Standard Code for Information Interchange). Ezek a kódszabványok az angol ABC kis- és nagybetűit, a decimális számjegyeket, néhány írásjelet és adatátviteli vezérlőjeleket tartalmaznak. A vezérlőkaraktereknek megfelelő bitkombinációk vagy valamilyen akciót váltanak ki abból a berendezésből, amelyekre küldjük őket, vagy jeleznek valamit. Azokat a karaktereket, amelyek a nyomtatón megjeleníthetők *nyomtatható*, a többi *nem nyomtatható karakterek*nek nevezzük.

### 3.5 RELÁCIÓS MŰVELETEK A SZÁMÍTÓGÉPEN

A program futása során gyakran van szükség arra, hogy két számról el tudjuk dönteni, hogy melyik a nagyobb, vagy két karaktersorozatról el tudjuk dönteni, hogy melyik áll előbb az ABC sorrendben. Ezeket az összehasonlító műveleteket relációs műveleteknek nevezzük. A számítógépen az összehasonlítás technikailag kétféle módon mehet végbe, attól függően, hogy milyen adattípusról van szó. Másképpen hasonlítja össze a gép a fix vagy lebegőpontos számokat és a karakterenként kódolt jelsorozatokat (*karakterláncokat* vagy *sztringeket*). A fix és lebegőpontos számoknál az összehasonlítás eredménye ténylegesen azt mutatja meg, hogy a két szóban forgó szám közül melyik a nagyobb. A karaktersorozatnál az összehasonlítás karakterenként történik, az a karakter a „nagyobb”, amelynek kódja nagyobb. A karakterek kódértékét a kódtáblázatok tartalmazzák. Az összehasonlítás addig folytatódik, amíg a két karakterláncban nem talál a gép eltérést. Ha van eltérés, az összehasonlítás elvégezhető, ha nincs, azaz az egyik lánc véget ér, kétféle eredmény lehetséges – programozási nyelvtől függően. A gép vagy egyenlőnek tekinti a sorozatokat, vagy a rövidebbet tekinti kisebbnek.

Összehasonlítási műveletek:

- ☒ kisebb, <
- ☒ kisebb vagy egyenlő, ≤
- ☒ egyenlő, =
- ☒ nagyobb, >
- ☒ nagyobb vagy egyenlő, ≥
- ☒ nem egyenlő, ≠

Az összehasonlítás eredménye *logikai igen* vagy *nem* érték. A relációk kombinálásával bonyolultabb döntéseket is hozhatunk.

### 3.6 A SZÁMÍTÓGÉPES LOGIKA

A logikában ún. állításokkal dolgozunk. Minden logikai állítás két értéket vehet fel: lehet *igaz* vagy *hamis*. A logikai állítások egyszerűbb kezelésére ún. *logikai változókat* vezetünk be, az állításoknak megfeleltetünk egy önálló tartalommal nem bíró jelölést (pl. A), amelynek értéke igaz, ha a logikai állítás igaz, és hamis, ha az állítás hamis.

Két logikai változó között értelmezhetünk különféle logikai műveleteket. Ezek közül az ÉS műveletet, a VAGY műveletet és a NEM műveletet alpműveleteknek nevezzük, mivel e három művelet kombinációjaként az összes többi logikai függvény felírható.

#### ÉS művelet

A ÉS B

Jelölése:  $Q=AB$

A művelet eredménye (Q) igaz, ha A és B is igaz, egyébként hamis.

**VAGY művelet**

A VAGY B

Jelölése:  $Q=A+B$

A művelet eredménye (Q) igaz, ha A vagy B igaz, hamis, ha mindkettő hamis.

**NEM művelet**

NEM A

Jelölése:  $Q=\bar{A}$

A művelet eredménye (Q) igaz, ha A hamis, hamis, ha A igaz.

A műveletek könnyebb áttekintésére bevezetjük az összes lehetséges eredményt tartalmazó, ún. *igazságtáblákat*, amelyek a műveletek meghatározásának is tekinthetők. A számítástechnikában elfogadott jelölést használva, ha egy változó értéke igaz, azt az „1” számmal, ha hamis, a „0”-val jelöljük.

A bemenet értékei		Konstans 0	NEM-VAGY	A tiltása ÉS B	Egyváltozós NEM (A)	B tiltása ÉS A	Egyváltozós NEM (B)	Kizáró VAGY (ANTIVALENCIA)	NEM-ÉS	ÉS	Egyenértékűség (EKVIVALENCIA)	Egyváltozós egyenértékűség (B)	Ha A akkor B (IMP-LIKÁCIÓ)	Egyváltozós egyenértékűség (A)	Ha B akkor A (IMP-LIKÁCIÓ)	Megengedő VAGY	Konstans 1
A	B	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

**3-3. táblázat:** Egy és kétváltozós Boole-műveletek igazságtáblázatai







A logikai függvények alapja a Boole-algebra. A legfontosabb összefüggéseket és törvényeket összefoglaljuk:

$$\begin{array}{l} \bar{\bar{A}} = A \\ A(B+C) = AB + AC \\ A(A+B) = A \\ AA = A \\ \overline{AB} = \bar{A} + \bar{B} \end{array} \qquad \begin{array}{l} A+(BC) = (A+B)(A+C) \\ A+(AB) = A \\ A+A = A \\ \overline{A+B} = \bar{A}\bar{B} \end{array}$$

Az utolsó két összefüggést De Morgan azonosságoknak nevezzük.

A logikai műveletek képezik az aritmetikai műveletek alapját a számítógépes végrehajtásban, hiszen az egyes komplementek képzése a bitenkénti NEM művelet, két bit összeadása a kizáró VAGY művelet két bit között, két bit összeszorozása az ÉS művelet.

A logikai műveleteket a számítógépekben ún. kapuáramkörök segítségével valósítják meg.

Kapuáramkör neve	Logikai függvény	Rajzjele
Inverter (NEM kapu)	$Q = \bar{A}$	
Kétbemenetű ÉS kapu	$Q = AB$	
Kétbemenetű VAGY kapu	$Q = A + B$	
Kétbemenetű NEM ÉS (NAND) kapu	$Q = \overline{AB}$	
Kétbemenetű NEM VAGY (NOR) kapu	$Q = \overline{A + B}$	
Kétbemenetű kizáró VAGY (XOR) kapu	$Q = A \oplus B$	

3.4. ábra Logikai kapuáramkörök

## 4 A SZÁMÍTÓGÉPEK ARCHITEKTÚRÁJA

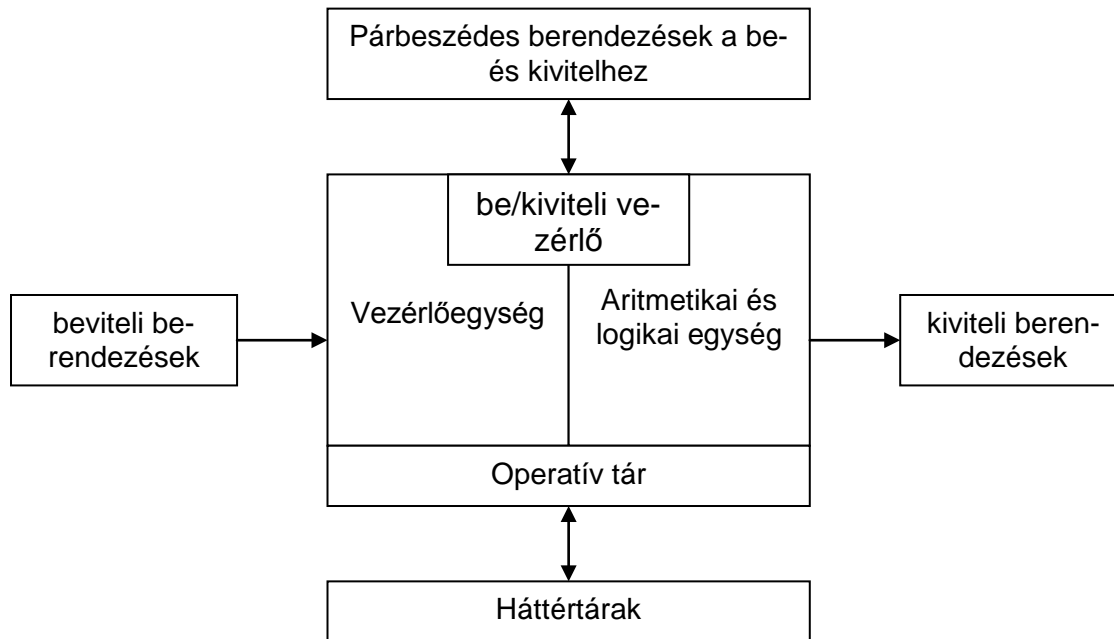
A számítógép két komponensből álló együttes. A fizikailag megfogható részt *hardvernek*, míg a logikai, azaz programrészt *szoftvernek* nevezzük. A szoftver összefoglalóan a programokat, ezek dokumentációit, általában a gép használatához szükséges szellemi termékeket jelenti.

### 4.1 A SZÁMÍTÓGÉPES HARDVER SZERKEZETE

A számítógép feladata az adatok feldolgozása. A gép ehhez az alábbi funkciókat valósítja meg:

- ☞ adatok *bevitele*,
- ☞ adatok *tárolása*,
- ☞ adatok *feldolgozása*,
- ☞ adatok *kivitele*.

A különböző funkciókhoz általában különböző berendezések tartoznak. A gép irányítását a *központi egység* (CPU=Central Processing Unit) végzi. Ugyanitt történik az adatok tényleges feldolgozása is. A központi egység irányítása alatt működnek a különböző kiviteli, beviteli és tárolóeszközök. A *kiviteli, beviteli és tárolóeszközöket* együttesen *perifériáknak* nevezzük.



4.1. ábra: A számítógép hardverszerkezete

A *vezérlőegység* és az aritmetikai és logikai egység együttes neve a processzor (CPU).

A vezérlőegység feladata a program utasításainak végrehajtása. A vezérlőegység elektronikus vezérlőjeleket küld az aritmetikai és logikai egységnek, az operatív tárnak, a be/kiviteli vezérlőegységnek, és utasítja azokat a kijelölt műveletek végrehajtására.

Az *aritmetikai és logikai egység* áramkörei az aritmetikai és logikai műveleteket végzik. Itt történik a relációk kiértékelése is.

Az *operatív tárban* vannak mindazok az adatok és utasítások, amelyek az aktuálisan futó program végrehajtásához szükségesek.

A *be/kiviteli vezérlőegység* funkciója a perifériák és a központi vezérlőegység közötti adatforgalom vezérlése. Ehhez elektronikus jelekkel kiválasztja a megfelelő perifériát, kiépíti a kapcsolatot és vezérli az átvitelt.

Miután a vezérlőegység aktiválta a be/kiviteli vezérlőegységet, és átadta a be/kivitel lebonyolításához szükséges információkat, a be/kiviteli egység önállóan képes működni.

A gyors működésű operatív tár csak viszonylag kevés adat tárolására alkalmas, ezért a nagy adattömegeket, nyilvántartásokat ún. *háttértároló*kon őrizzük. A háttértárolók rendszerint mágneses vagy optikai elven működő berendezések, melyek igen nagy mennyiségű adat tárolására képesek. A programok szintén a háttértárakon őrződnek, csak a végrehajtáskor kerülnek az operatív tárba.

A gép és ember közötti kapcsolatot valósítják meg a *be- és kiviteli egységek*. Ezek közül különleges helyet foglalnak el a be- és kivitel egyaránt végző párbeszédes módú *terminálok*. A terminálokon az adatokat billentyűzetről, egérrel, fényceruzával stb. tudjuk bevinni, a kivitel pedig képernyőre vagy nyomtatóra történik.

A hardver fontos elemei még az *adatátviteli berendezések*. Ezeket több gép összekapcsolása során, vagy a központi egységtől nagyobb távolságra elhelyezett perifériák használatánál alkalmazzuk.

## 4.2 A SZÁMÍTÓGÉPES SZOFTVER SZERKEZETE

Azoknak a programoknak az összességét, amelyek a felhasználó feladatait oldják meg *felhasználói szoftver*nek nevezzük.

Azokat a programokat, pontosabban azt a programrendszert, amelyek a gép egyes berendezéseit működtetik, *rendszer-szoftver*nek nevezzük. A rendszer-szoftver a gép része.

A két réteg között elhelyezkedő hasznos segédprogramokat *rendszerközeli szoftver*nek nevezzük. Ezek a programok a felhasználó munkáját könnyítik meg.



4.2. ábra: A számítógép szoftver rétegei

A számítógéppel együtt megvásárolt rendszer-szoftver és a rendszerközeli szoftver együttesét *operációs rendszer*nek nevezzük.

## 5 A KÖZPONTI EGYSÉG

### 5.1 A SZÁMÍTÓGÉPHÁZ ÉS A TÁPEGYSÉG

A számítógépház fémből készült, műanyag előlappal ellátott doboz. A háznak többféle típusa létezik, ezek méretben és a beépíthető perifériák számában térnek el.

#### 5.1.1 TÁPEGYSÉG

A számítógép áramkörei kisfeszültséggel működnek, az energiát pedig a 230 V-os hálózatról kapják. Ezért szükség van egy olyan egységre, amely a 230 V-os feszültségből előállítja a kisfeszültségeket. Ez a tápegység feladata.

A hálózati feszültég a terheltségtől függően változik akár 10-15 %-ot is. Ezt az elektronikus áramkörök nem, vagy csak nehezen viselik el. Az optimális megoldás az, ha mindig azonos feszültségről üzemelnek. A stabilizált tápegységek képesek a kimeneti feszültséget – a terheléstől függetlenül – azonos szinten tartani.

Alapvető követelmények a tápegységekkel szemben:

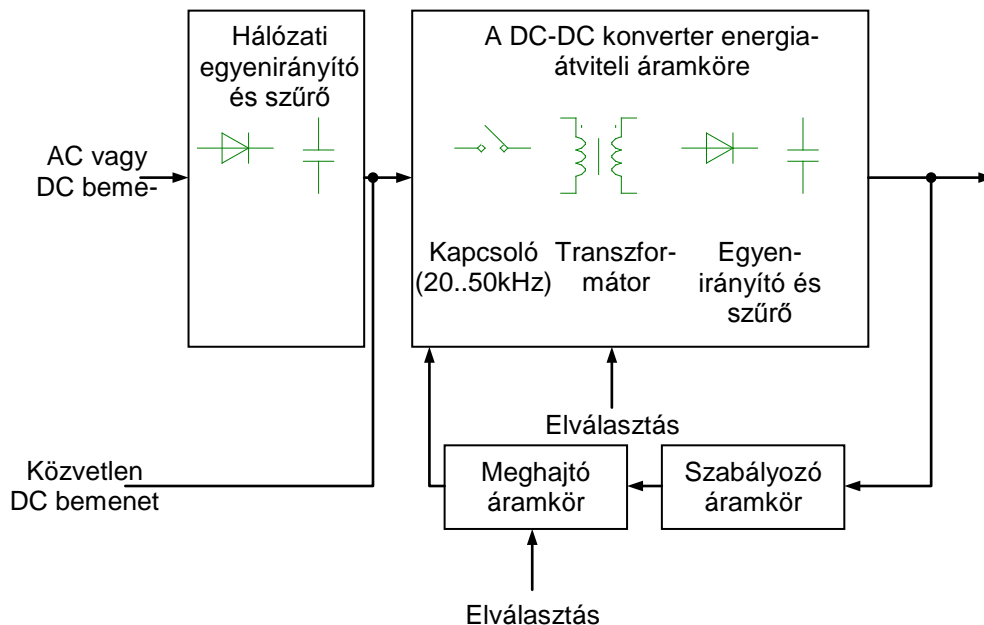
- ☞ A rendelkezésre álló tápforrás feszültségének átalakítása a készülék üzemeltetéséhez szükséges egyenfeszültséggé.
- ☞ Az előállított feszültség stabilizálása.
- ☞ Rövid időtartamú bemeneti feszültségkimaradás esetén a kimeneti feszültség szünetmentes biztosítása.
- ☞ A táplált áramkör védelme a bemenetről származó tranziensektől vagy a tápegység meghibásodásától.
- ☞ A tápegység kimenetének a bemenettől való galvanikus leválasztása.

A tápegységek csoportosítása üzemmód szerint:

- ☞ Lineáris tápegység: kisebb elektronikus eszközökben használják.
- ☞ Kapcsolóüzemű tápegység: a számítógépekben ezt a típust alkalmazzák.
- ☞ Előszabályozott tápegység: csak ritkán, speciális esetekben használatos.

A számítógép működése szempontjából tehát a kapcsolóüzemű tápegységnek van jelentősége. Nézzük meg a működését!

A kapcsolóüzemű tápegység hálózati oldalán nincsen transzformátor. A hálózati feszültséget egyenirányítjuk, szűrjük, majd a feszültség egy DC-DC konverterre kerül. Ez a bemenetre kapcsolt feszültségből valamilyen más, kisebb vagy nagyobb feszültséget állít elő. Funkcionálisan négy részből áll. A kapcsolóval az egyenfeszültségből néhány tíz kHz frekvenciájú négyszögjelet képezünk. Ezt egy nagyfrekvenciás transzformátorra vezetjük, mely a galvanikus elválasztást valósítja meg. Az átalakított kimeneti feszültséget újra egyenirányítjuk és egy szűrőn átvezetve kapjuk a tápegység kimenetén a stabilizált egyenfeszültséget. A stabilizálást a kapcsolóelemen kívül a szabályozó és meghajtó áramkör végzi. A szabályozó egység a kapcsoló megfelelő periódusidejű kapcsolgatásáért felelős. A bemenetere a kimeneti feszültség kapcsolódik, kimeneti jele pedig a kapcsolójel. Ha a kimeneti feszültség csökken, akkor a kapcsolót gyakrabban kell kapcsolgatni, ha viszont a kimeneti feszültség növekszik, akkor pedig ritkábban, tehát a kimeneti feszültség a kapcsoló be- és kikapcsolásának idejétől függ. Ezt a módszert nevezzük impulzusszélesség modulációnak. A meghajtó áramkör a visszacsatoló áramkörben is biztosítja a galvanikus elválasztást.



5.1. ábra: A kapcsolóüzemű tápegység elvi felépítése

A számítógépben lévő áramkörök négyféle feszültséget igényelhetnek, legalábbis ennyire van lehetőség: +5V, -5V, +12V, -12V.

Mivel a stabilizálás visszacsatolás útján történik, a megfelelő feszültségértékek rendelkezésre állásához bizonyos idő szükséges. Ezt nevezzük beállási időnek. Mivel ezt a jellemzőt sok tényező befolyásolhatja, a rendelkezésre állás tényét a tápegység közli a számítógép alaplapjával. Ez a Power-Good jel. Ez a jel a kimeneti egyen- és a bemeneti váltakozófeszültséget érzékelő jelnek a logikai ÉS kapcsolatából jön létre. Hibás működés esetén alacsony logikai szintű, míg hibátlan működés esetén magas logikai szintet ad.

Ha valamilyen oknál fogva megszűnik a bemeneti váltakozó feszültség, akkor az érzékelő a kimeneti feszültség csökkenése előtt minimum 1 ms-mal alacsony szintre állítja a Power-Good jelet. Az egyenfeszültség-érzékelő a bekapcsoláskor alacsony szintre állítja, majd az összes kimenet megfelelő értéke után minimum 100, de maximum 500 ms ideig még ott is tartja.

Ha bármelyik kimenet túlterhelődik, a tápegység 20 ms-on belül kikapcsol, ezzel védve meg az áramköreit a károsodástól. Ezzel a megoldással a tápegység a rövidzárral szemben is védett.

Lehetőség van olyan kialakításra, hogy a monitor a tápegységen keresztül kapja a hálózati feszültséget. Ekkor a POWER gombbal egyszerre lehet a monitort és a számítógépet bekapcsolni.

A tápegység fizikailag külön dobozban található meg a házban belül. A megfelelő szellőzés biztosítása érdekében egy +12 V-ról működő ventilátor szívja ki a meleg levegőt a dobozból. Ez a +12 V-os kivezetés nincs összekapcsolva a perifériák számára előállított +12 V-tal.

### 5.1.2 NYOMÓGOMBOK, KAPCSOLÓK, KIJELEZŐK

Ezek szolgálnak a számítógép működésébe való beavatkozásra.

#### Kapcsolók

**RESET:** a kapcsoló megnyomásával a számítógép hardveres újraindítását lehet elérni.

**TURBO:** a Pentium előtti számítógépek esetében volt jelentősége. Segítségével kétféle működési sebesség közül lehetett választani.

**POWER:** hálózati kapcsoló, a 230 V-ot kapcsolja a tápegységre és a monitorra.

A POWER kapcsoló vezetékén kívül mindegyik kapcsoló és nyomógomb csatlakozóját az alaplapon megfelelő helyére tudjuk ráilleszteni.

#### Visszajelzők

**POWER:** A bekapcsolt állapotot jelző, általában zöld színű LED.

**TURBO:** A kapcsoló benyomott állapotát jelzi.

**HDD:** Piros színű LED, mely akkor bocsát ki fényt, ha merevlemez művelet van.

### 5.1.3 HANGSZÓRÓ

Általában minden számítógépházban találunk ilyen, alapvető feladata az akusztikus visszajelzés működési probléma esetén. Ha a monitorvezérlő működésképtelen, e nélkül nem lehetne a hibát detektálni. Ma már más jelentősége nincs, mert gyakorlatilag minden számítógépben van hangkártya és hozzá illeszthető hangszóró. A hangszóró vezetéke szintén az alaplapon csatlakozik.

### 5.1.4 BŐVÍTŐ HELYEK

Minden ház hátulján vannak nyílások, ahol a bővítőkártyák hátlapjai érhetők el. Használaton kívül ezeket csavarral rögzített lemezcsíkok fedik.

### 5.1.5 HÁZTÍPUSOK

#### Baby ház

Régebben ez a fekvő elrendezésű ház volt a legkedveltebb. A meghajtó bővítőhelyek száma gyártófüggő. A számítógépház alján különböző furatok teszik lehetővé az alaplapon biztonságos rögzítést.

A fedél rögzítése háromféleképpen történhet:

1. Csavarok tartják, melyek kicsavarása után a fedél az előlappal együtt lehúzható a fém tartóvázról.
2. Csavarok rögzítik a fedelet, ez azonban nincs összeépítve az előlappal, így a fedél felfelé egyszerűen leemelhető a házról.
3. A fedél hátsó részét csavarok, míg az első részt két retesz rögzíti. A reteszeket benyomva a ház fedele felnyitható.

#### Mini torony

Az előlapon található a kezelőszervek. Általában két meghajtó építhető bele, de ez gyártótól függ. Az alaplapon függőlegesen helyezkedik el és egy fémlapra van szerelve. Ezt a lemezt csavarok rögzítik a ház keretéhez. A kártyák és így a nekik kiképzett nyílások is vízszintesen helyezkednek el. A fedél csavarokkal van rögzítve a kerethez, ezek száma 4, 5 vagy 6 lehet.

#### Midi torony

Hasonló, mint a mini torony, azonban a bővítőhelyek számát megnövelték és a doboz méretei nagyobbak. Általában három meghajtó szerelhető bele.

#### Torony vagy nagytorony

A toronyházak közül a legnagyobb méretű. Általában a nagyteljesítményű szerverekhez használják. Minimum öt meghajtó építhető bele.



### **Slim ház**

Ez is egy fekvő ház, azonban a magassága pontosan akkora, hogy egy 3,5" floppy függőlegesen elfér benne. Az alaplap vízszintesen fekszik benne, de a bővítőkártyák nem férnek el függőlegesen. A megoldás egy olyan bővítőkártya használata, melyen két kártyacsatlakozó hely van és az alaplap csatlakozójába dugható. Így a bővítőkártyák is vízszintesen helyezkednek el.

### **ATX ház**

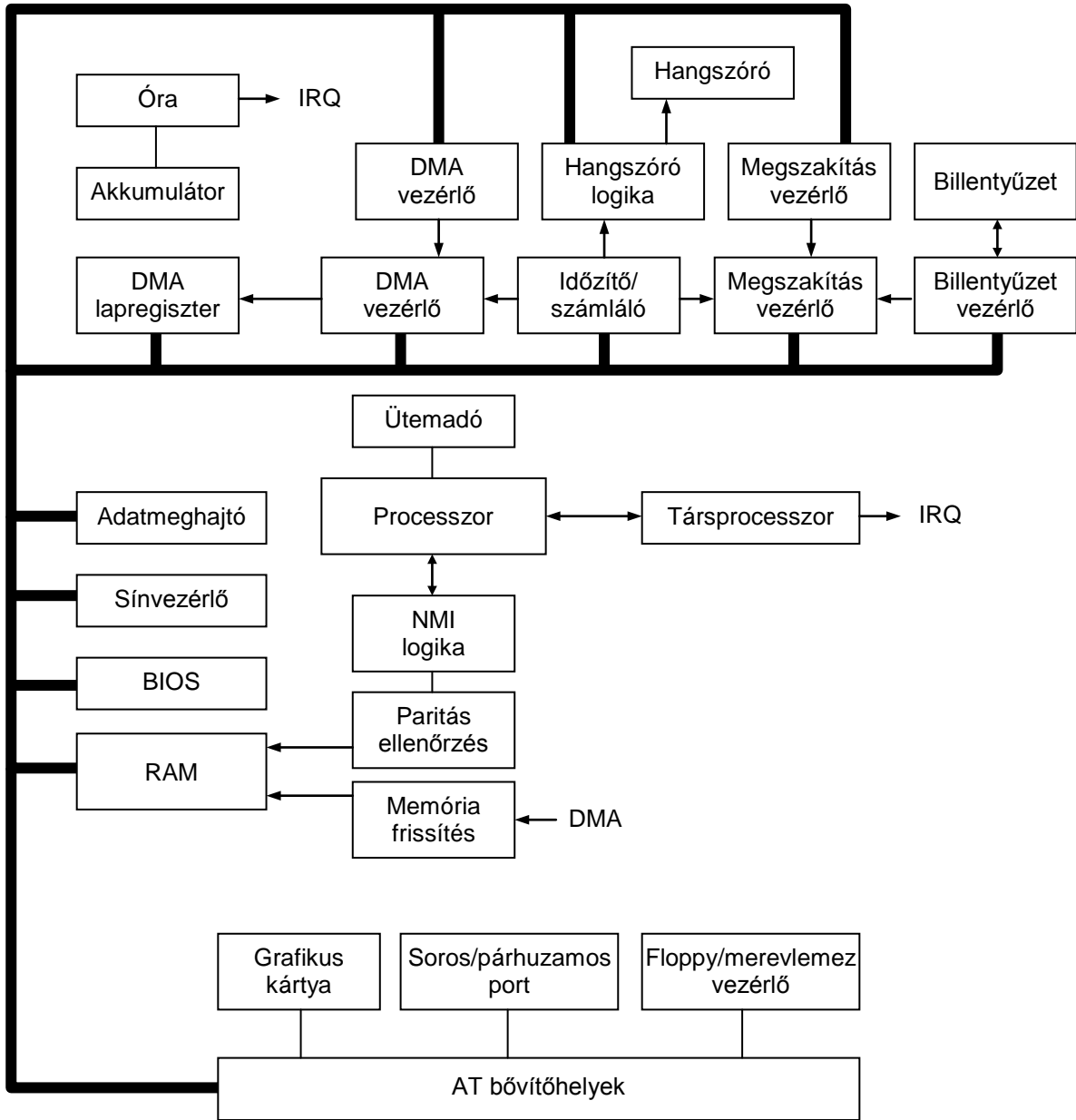
A mai legkorszerűbb házak. Kialakításukkor megalkottak egy szabványt, melyet ATX szabványként ismerünk. Ez kimondja, hogy minden csatlakozó az alaplapra kerüljön előre rögzített helyre. Az alaplap méreteit is és a felfogatás helyeit is meghatározták. Az ATX alaplapoknak más tápcsatlakozójuk van, melyen keresztül kihasználhatják az ATX ház intelligens tápjának tulajdonságait.

A legfontosabb, hogy az előlapon lévő POWER kapcsoló nem a tápot kapcsolja le, hanem csak takarékos üzemmódba helyezi. Ekkor úgy tűnik, mintha ki lenne kapcsolva a gép. Valamilyen esemény bekövetkezése esetén a számítógép automatikusan bekapcsolható, majd a megfelelő tevékenységek után szintén automatikusan elvégezhető a kikapcsolás is. Ezeknek a házaknak a tápegységeit is kikapcsolhatjuk a ház hátulján található kapcsolóval.

Az ATX házak is kaphatók többféle méretben. Ezek elnevezései hasonlóak, mint a hagyományos házaké. A méreteik azonban nagyobbak.

## 5.2 AZ ALAPLAP

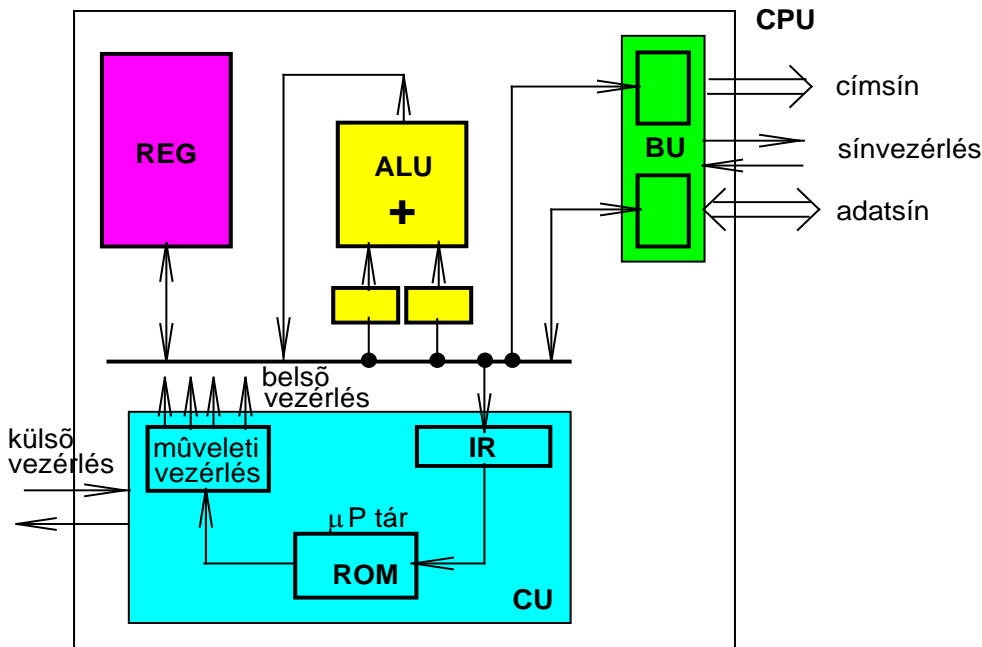
A számítógép működéséhez elengedhetetlen áramköröket egy közös áramköri lapon alakították ki. Ez a számítógép alaplappja.



5.2. ábra: Az AT alaplapp felépítése

### 5.3 A MIKROPROCESSZOR

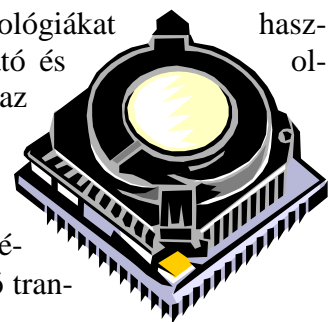
A mikroprocesszor elnevezésben a mikro a kis fizikai méretre utal, a processzor pedig a számítógép központi egységét jelöli. A  $\mu P$  segítségével minden logikai függvény megvalósítható.



5.3. ábra: A mikroprocesszor fő részei

#### 5.3.1 A MIKROPROCESSZOROK TECHNOLÓGIÁJA

A mikroprocesszorok gyártásánál NMOS, illetve CMOS technológiákat használnak. A tervezés és gyártás során cél az, hogy gyors, megbízható és olcsó legyen az előállítás. Ugyanakkor figyelembe kell venni, hogy az integráltsági fok növelésének következménye az egyre nagyobb hőfejlődés, tehát az alkalmazás során meg kell tudni oldani a mikroprocesszor hűtését.



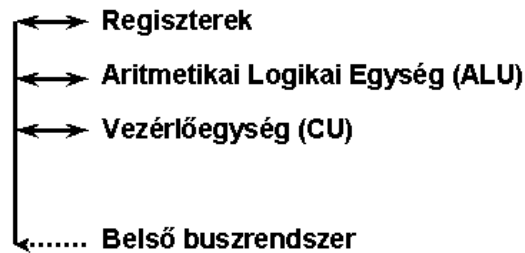
Például a Pentium II mikroprocesszor CMOS technológiával készül, a chip mérete 15x15 mm, a vonalak mérete 0,25  $\mu m$ , 7,5 millió tranzisztort tartalmaz és a belső tápfeszültség értéke 2 V.

#### 5.3.2 A MIKROPROCESSZOROK CSOPORTOSÍTÁSA

A mikroprocesszorokat csoportosíthatjuk szóhosszúság, utasításformátum, utasításkészlet, ciklusidő, a címezhető memória nagysága valamint a buszrendszer típusa szerint.

- ☞ szóhosszúság 4 .. 64 bit
- ☞ utasításformátum CISC, RISC
- ☞ utasításkészlet 100 .. 1000
- ☞ ciklusidő 4,77 .. 400 MHz
- ☞ címezhető memória 64 kB .. 4 GB
- ☞ buszrendszer 8 .. 64bit, 8 .. 66 Mhz

### 5.3.3 A MIKROPROCESSZOROK FELÉPÍTÉSE



5.4. ábra: A mikroprocesszor funkcionális egységei

### 5.3.4 REGISZTEREK

A mikroprocesszor *regiszterei* nagyon gyors működésű átmeneti tárolók. Statikus RAM-okból alakítják ki őket, így elérési idejük ns nagyságrendű, ez 10-szer, 100-szor gyorsabb, mint a számítógép operatív táráé. A processzor típusától függően 8, 16 ... 512 db regisztertárat tartalmaz.

Egy regiszter tárolókapacitása általában egy szó (pl. 32 bit). Léteznek rendszerregiszterek, melyekhez a felhasználó nem férhet hozzá, és általános célú regiszterek, melyeket a programok is használhatnak.

#### Tipikus regiszterek:

Általános célú regiszterek:

- ☞ Akkumulátor (A): Az akkumulátor tárolja az aritmetikai/logikai műveletek operandusait, és általában az eredmény is ide kerül.
- ☞ Az akkumulátoron kívül a processzornak van még B, C, D, E, H és L regisztere, a H és L regiszter a memória címezésére szolgál, a többi rendeltetése tetszőleges.
- ☞ Utasításregiszter (IR): Itt tárolja a processzor a végrehajtandó utasítás műveleti kódját, tartalmát a vezérlőegység értelmezi és előállítja a műveletek végrehajtásához szükséges vezérlő jeleket.
- ☞ Utasításszámláló (PC): Mindig a következő végrehajtandó utasításra mutat, tartalma automatikusan inkrementálódik, értéke programozottan módosul, a megcímezett rekesz tartalma általában az utasításregiszterbe kerül.
- ☞ Adatszámológó regiszter (DC): A beolvasandó adatra mutat, a megcímezett rekesz tartalma általában az akkumulátorba kerül, értéke programban módosítható.
- ☞ Címregiszterek
- ☞ Adatregiszterek

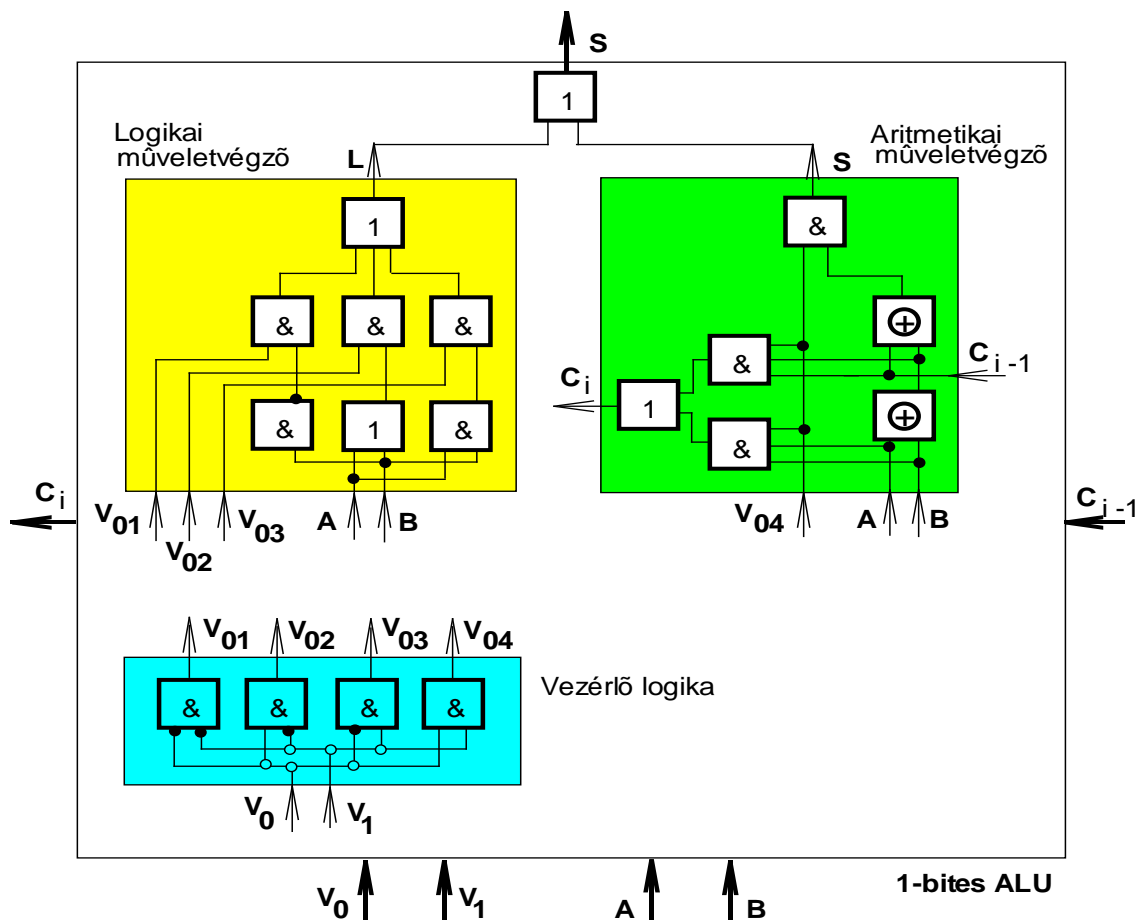
Rendszerregiszterek:

- ☞ Címbusz regiszter (Memory Address Register)
- ☞ Adatbusz regiszter (Memory Data Register)

### 5.3.5 ARITMETIKAI ÉS LOGIKAI EGYSÉG

Az ALU feladata a különböző aritmetikai és logikai műveletek végrehajtása. Ezek tipikusan a következők:

- ☞ Bináris összeadás
- ☞ Boole-algebrai műveletek (AND, OR, XOR, NOT)
- ☞ Léptetés jobbra-balra
- ☞ Komplement képzés
- ☞ Állapotjelzők előállítása (Flag)

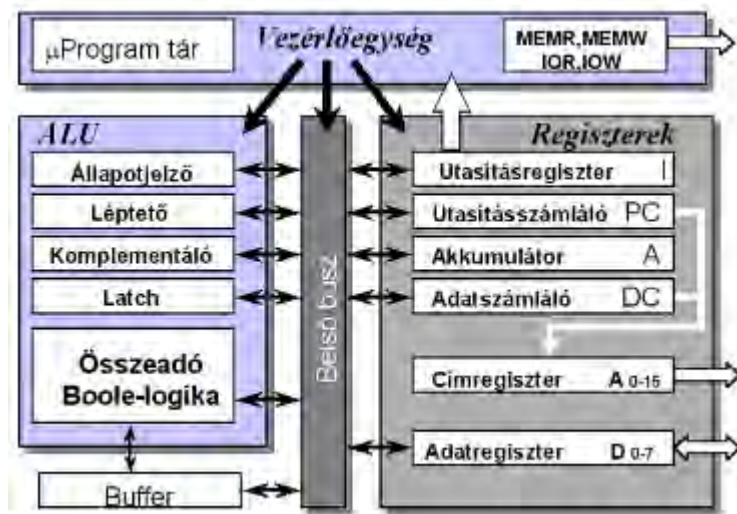


5.5. ábra: 1 bites ALU elvi felépítése

### 5.3.6 A VEZÉRLŐEGYSÉG (CU)

A vezérlőegység kiolvassa a memóriából a szükséges adatokat, utasításokat, értelmezi és végrehajtja az utasításokat, vezérli az ALU-t, engedélyező jeleket generál. Összehangolja a CPU többi egységének működését. Tartalmazza a mikroprogram-tárat.

A vezérlőegység feladata, hogy a megfelelő adatok a megfelelő helyen és a megfelelő időben rendelkezésre álljanak.



5.6. ábra: A mikroprocesszor belső felépítése

A vezérlőjelek előállításának és időzítésének két módja van: *mikroprogramok* vagy *hardver* útján.

Az első esetben a vezérlőjel-sorozatokat egy speciális program, a mikroprogram szolgáltatja. A gépi kódú utasítások elemi lépések sorozatára bonthatók. Létezik az elemi lépéseknek egy halmaza, amelyekből bármely gépi kódú utasítás is felépíthető. Ha ezeket az elemeket valamilyen sorrendbe rakjuk, akkor egy nullákból és egyesekből álló bitsorozat tetszőleges részsorozatot ki tud jelölni az elemekből oly módon, hogy ha az adott elemi lépésnek megfelelő bit egyes, a lépés végrehajtódik, ha nulla, nem. Így egy bitsorozattal leírható egy gépi kódú utasítás. Ez a mikroprogram.

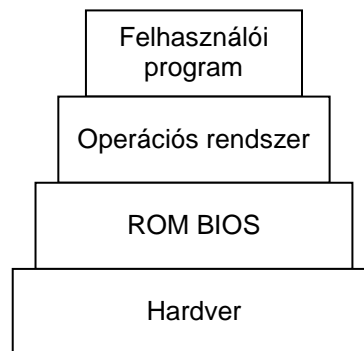
A mikroprogramok egy külön tárban, a *mikroprogramtárban* helyezkednek el. A gép utasításkészlete gyakorlatilag csak a megadott mikroprogramoktól függ.

A második esetben, a hardver vezérlésénél az utasítások „*be vannak huzalozva*”, azaz minden utasítás fix áramköri kapcsolatok alapján hajtódik végre.

A hardvervezérlés jóval gyorsabb, mint a mikroprogramozott, de ugyanakkor jóval bonyolultabb áramköröket igényel.

## 5.4 BIOS

Minden PC alaplapján található egy vagy több csak olvasható memória (ROM). Ez azokat az alapvető rutinokat tartalmazza, melyekre az operációs rendszernek szüksége van. Ezeket a rutinokat nevezzük közös néven *központi be/kimeneti rendszernek* (BIOS – Basic Input/Output System). A BIOS tartja a kapcsolatot a hardver és az operációs rendszer között. Az operációs rendszerre épülnek a felhasználói programok. A BIOS teszi lehetővé, hogy az operációs rendszer elérhesse a számítógép áramköreit (erőforrásait).

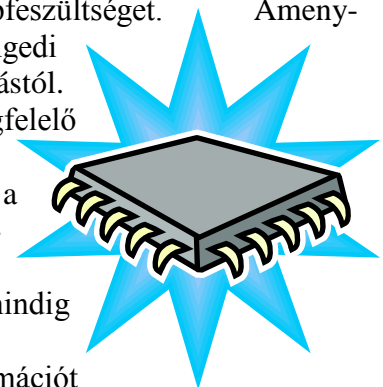


5.7. ábra: A PC virtuális felépítése

Mivel a ROM csak olvasható, ezért a tartalmát nem lehet megváltoztatni. Amennyiben új rutinokra lenne szükség, akkor az egész tokot ki kellene cserélni. A BIOS-t az alaplap gyártója készíti el az alaplapra integrált áramköröket figyelembe véve. A BIOS tehát önálló programmodulok gyűjteménye.

Ha bekapcsoljuk a számítógépet, akkor működésbe lép egy öntesztelő áramkör. Ez az áramkör megvizsgálja a számítógépben jelen lévő összes tápfeszültséget. Amennyiben valami nem működik rendesen, akkor az áramkör nem engedi működni a számítógépet, ezzel próbálja megvédeni a meghibásodástól. Ha mindent rendben talál, akkor ezt az alaplapnak jelzi egy megfelelő jellel, majd engedélyezi az alaplap órajel generátorát. Ez az áramkör biztosítja a processzor és az egyéb áramkörök részére a működtető órajelet. Mivel már van órajel, ezért a processzor elkezd működni. A processzorban megtalálható az a cím, ahonnan az elindulásakor be kell olvasni az első utasítást. Ezen a címen mindig egy ugróutasítás van, amely átadja a vezérlést a ROM-BIOS-nak.

Indulás esetén a BIOS megvizsgál egy jelzőbitet, ami információt



nyújt arról, hogy kezdeti indítás (hidegindítás) volt-e vagy csak ún. melegindítás történt. Ettől a bittől függ a számítógép indítási folyamatának folytatása. Ha hidegindításról van szó, akkor el kell indítani egy diagnosztikai programot, ami megvizsgálja, hogy az elérhető áramkörök működőképesek-e. Ha valamilyen hibát talál, akkor a képernyőre írja a megfelelő hibaüzenetet (POST kód). Természetesen ez csak abban az esetben megfelelő módszer, ha a videóvezérlő biztonságosan működik. Ha ezzel hiba van, tehát a vizuális hibakezelés lehetetlen, akkor a BIOS fűtty kódokkal jelzi a hibát. Ez a teszt melegindítás esetén elmarad.

A hibakeresésre alkalmas programot a BIOS tartalmazza. A legelterjedtebb program az IBM POST (Power On Self Test – bekapcsolás utáni önteszt). A POST a számítógép összes fontos áramkörét leteszteli. Ha minden rendben van, akkor megjeleníti a bejelentkező információkat. Ha hiba történik, akkor a hibának megfelelő kódot és a hozzá tartozó szöveget kiírja a képernyőre.

## 5.5 A CPU ÓRÁJA

A számítógép működése nem folytonos, az események diszkrét időpillanatokban mennek végbe. Ezeket az időpillanatok egy nagy pontosságú beépített óra szolgáltatja. Az óra egyenlő időközökben ún. *órajelet* bocsát ki, ekkor történhet műveletvégzés a gépben, két órajel között a gép áll.

Az órajelek sebessége határozza meg a gép sebességét. A mai számítógépek jellemző frekvenciája 100–500 MHz. Egy aritmetikai művelet elvégzése egy PC-n néhányszor tíz óraciklus.

## 5.6 A SZÁMÍTÓGÉP BELSŐ TÁROLÓJA

A gép tárolója kétállapotú fizikai elemekből épül fel. A két jól megkülönböztethető állapot mellett követelmény az állapotok gyors változtathatósága is.

A tár felépítését tekintve két alapvető részre oszlik. Az egyik az ún. állandó tár, amelynek tartalmát a felhasználó nem tudja megváltoztatni. Ezt ROM-nak szokás nevezni. A másik a tényleges operatív tár, amelyet a felhasználó tetszése szerint használ (RAM). A ROM tartalmát a gyártó vagy a felhasználó alakítja ki.

ROM alaptípusok:

- ☞ szabványos ROM, melynek tartalmát a gyártó alakítja ki,
- ☞ programozható ROM (PROM), amelynek tartalmát a felhasználó alakítja ki speciális eszközökkel, a tartalma nem változtatható,
- ☞ törölhető, programozható ROM (EPROM), amelyet a felhasználó programoz speciális berendezés segítségével, és amelynek tartalmát erős ultraibolya sugárzással törölni lehet,
- ☞ elektromosan törölhető, programozható ROM, amelynek a törléséhez változó elektromos mezőt használnak.

A ROM a tápfeszültség megszűnésekor nem veszíti el tartalmát, ebben eltér a RAM-tól, amely kikapcsoláskor nem őrzi meg állapotát.

A RAM építőelemei leggyakrabban az ún. *triggerek*, háromállapotú áramkörök. A három állapot: egy, zéró, lekapcsolt állapot. Aktív állapotban az egy és a zéró állapotot használja a gép. Aktív állapotban (bekapcsolva) a trigger megőrzi tartalmát, míg kikapcsoláskor elveszíti. Ezt a RAM-típust statikus tárnak hívják.

A másik RAM építőelem az elektromos *kapacitás*ok elvén működik. Ezt dinamikus tárnak is szokás nevezni. A dinamikus RAM tartalmát aktív állapotban is másodpercenként több százszor meg kell újítani (frissíteni kell).

A tár legkisebb címezhető egysége tipikusan a byte, minden byte-nak önálló címe van. Az adatok és az utasítások általában több byte hosszúságúak. Bármely két tárrekesz tartalmát a

processzor teljesen egyforma módon, egy lépésben találja meg a rekesz címének segítségével. Ezért a memóriát *közvetlen elérésű tárolónak* nevezzük.

Olvasásnak azt a műveletet nevezzük, amikor egy tárrekesz tartalmát a műveletvégző egységbe visszük. Az olvasás ellentett művelete az írás, melynek során a tár egy rekeszébe adatot rögzítünk. Írás során a rekesz megelőző tartalma elvész, olvasás során a rekesz tartalma nem változik.

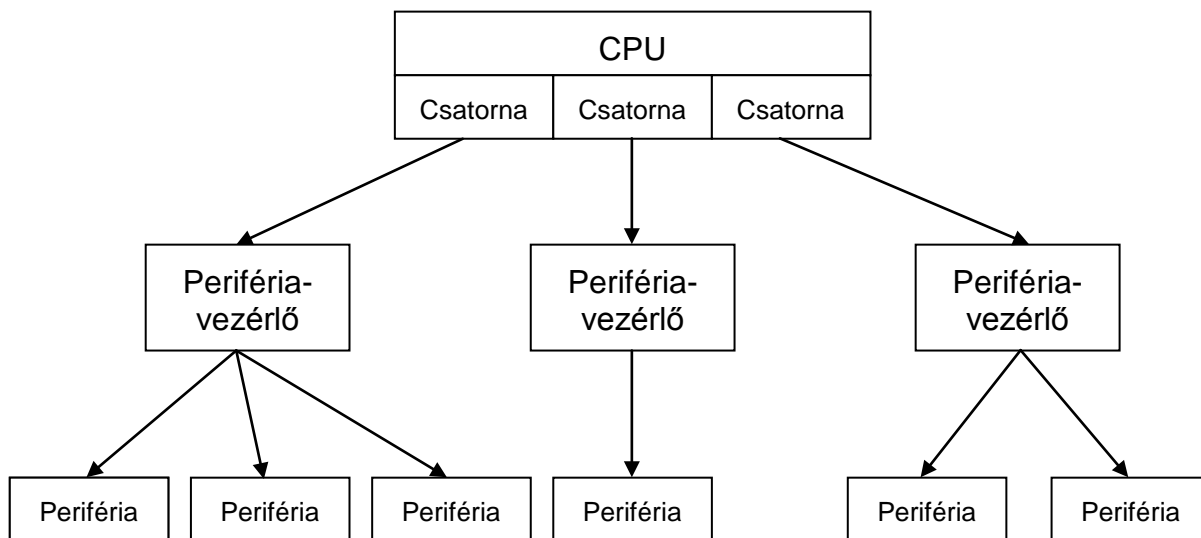
A tár minőségének jellemzésére két adatot használnak: a *kapacitását*, valamint a rekeszek *hozzáférési sebességét*, azt az időt, amely egy tetszőleges rekesz eléréséhez szükséges.

Az operatív tár sok esetben szűknek bizonyul, különösen akkor, ha a gépet több felhasználó egyidejűleg használja, és így osztozik a táron. Az operatív tárok minden határon túli fizikai bővítése egyrészt költséges, másrészt a címzési technika miatt sem valósítható meg. Ezért a tárat kibővítik gyors háttértárak egy adattároló területével. Ezt *virtuális tárnak* nevezik. Virtuális tár esetén a tárban mindig a programnak és az adatoknak csak éppen szükséges része tartózkodik, a többi a lemezen vár. Amikor a programrészre és adatokra már nincs szükség, azok kikerülnek a lemezre és onnan újabb program- és adatállomány-részletek kerülnek a tárba.

A virtuális-tár-kezelés technikailag úgy történik, hogy az operációs rendszer a programot és adatokat egyenlő nagyságú részekre, ún. *lapokra* bontja. Ezután a program működése során az operációs rendszer az aktuálisan szükséges lapokat olvassa be a tárba, a feleslegeseket pedig kiírja lemezre. Ez a művelet a lapozás.

## 5.7 ADATÁTVITELI RENDSZER

### 5.7.1 CSATORNARENDSZERŰ ADATÁTVITEL



5.9. ábra: A csatornák és a perifériák kapcsolata

Az információ átvitele csatornarendszer mellett mindig a központi egységben lévő operatív tár és a periféria között zajlik le. Az átvitel közben az információ többször átalakul (digitális jelekből analóggá és viszont). Ennek a folyamatnak a vezérlése sok művelettel jár, ezért a csatornarendszert különválasztották a mikroprocesszortól, ezzel megszabadítva a CPU-t egy sor nem közvetlenül belső művelettől. Az átviteli folyamat vezérlésére a gépekbe külön processzort építenek be, ez a csatorna.

A perifériák munkáját a *perifériavezérlő* szervezi és irányítja. A perifériavezérlő áll közvetlen kapcsolatban a csatornával, értelmezi annak utasításait és végrehajtja azokat a ráköött perifériákkal. Egy perifériavezérlő több periféria munkáját is irányíthatja.



A csatorna technikailag ugyanolyan programozható eszköz, mint a központi processzor. A különbség abban rejlik, hogy a felhasználónak a csatorna programozására nincs lehetősége, a csatornaprogram a gép része. A csatornaprogram adja át a központi processzornak szóló adatokat és kísérő információt, illetve veszi a processzortól a perifériáknak szóló üzeneteket.

A konkrét adatforgalom az operatív tár és a perifériák között megy végbe. A CPU miután az átvitelt kezdeményezte, felszabadul, és más műveleteket végezhet.

Amikor a periféria az egész adatátvitel tartamára leköti a csatornát, *monopol üzemmódban* dolgozik. A monopol üzemmód alatt a csatornára más periféria nem kapcsolódhat, egészen addig, amíg az átvitel be nem fejeződik. Az átvitt adatok mennyisége változó. A csatorna figyel, hogy ténylegesen folyik-e adatátvitel, vagy csak a program monopol üzemmódban lekötötte, ha a tényleges átvitel 30 másodpercig szünetel, a gép hibát jelez.

Ha a csatorna egyidejűleg több perifériát szolgál ki, *multiplex üzemmódban* működik. A multiplex üzemmódban egy periféria csak egy rövid intervallumra kapja meg a csatornát, ez alatt kell az átvitelre váró adatok egy csoportját továbbítani. Egy időpillanatban csak egy periféria van kapcsolatban a csatornával, de azt nem kötheti le tovább, mint az adott adatsoprot továbbításának idejére, ezután a csatorna tovább megy, függetlenül attól, hogy van-e az adott perifériának még „mondanivalója”. Ezután a csatorna ellátja a többi perifériát, majd ciklikusan visszatér.

A monopol üzemmód alkalmazása azoknál a perifériáknál szokásos, amelyeknél lehetőség van nagyobb mennyiségű adat gyors feldolgozására (pl. mágneslemez). Multiplex üzemmódot használnak a „lassú” perifériák, például néhány nyomtatótípus.

A csatornákat több osztályba sorolhatjuk:

- ☞ szelektor
- ☞ byte-multiplex
- ☞ blokkmultiplex

A *szelektorcsatorna* fizikailag csak egy perifériával tud kapcsolatot létesíteni, egyszerre egész adatblokkok vagy ezek sorozatainak átvitelére szolgál. Nem alkalmas a *multiplexelésre*, mindig monopol üzemmódban működik, egyidejűleg csak egy I/O műveletet hajt végre.

A *byte-multiplex csatornánál* mind a monopol, mind a multiplex üzemmód megengedett. Az üzemmód típusát a periféria határozza meg. A csatornán az átvitt információ egysége a byte.

A *blokkmultiplex csatorna* is képes mind monopol, mind multiplex üzemmódban működni, azonban a legkisebb átvitt adatsoprot az adatblokk, ami akár néhány ezer byte is lehet. Az adatblokk átvitelére a periféria mindenképpen monopolizálja a csatornát.

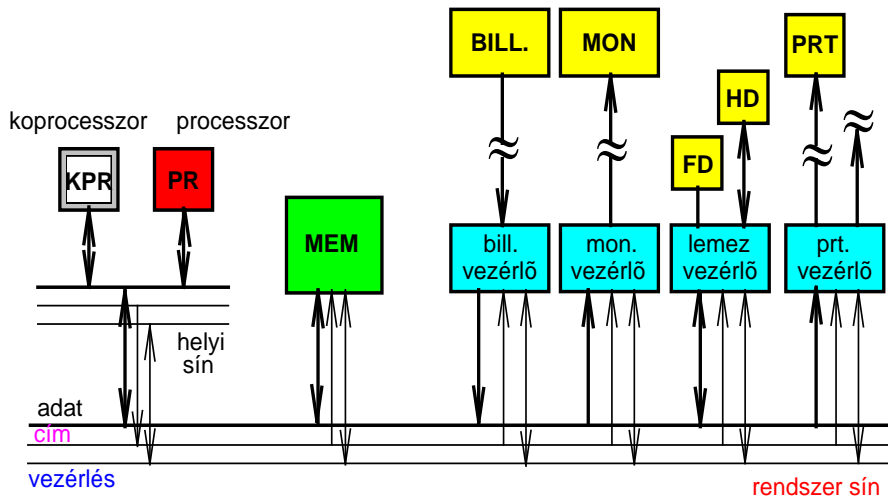
Az adatátvitel irányítása az ún. *felügyelőprogram* feladata. A felügyelőprogramot az operációs rendszer központi magja tartalmazza. Feladata a különböző programokban a különböző perifériákkal kezdeményezett adatátvitel megindítása, a beérkező megszakítások kezelése, ezen keresztül a műveletek koordinálása, az esetleges hibák jelzése, illetve javítása.

Amikor a felhasználói program be/kiviteli utasítást kezdeményez, ezt csak a felügyelőprogramon keresztül valósíthatja meg.

A processzor utasításkészletében a be/kiviteli utasítások 32 bit hosszúságúak, kétcímesek. A cím két összetevőből áll, a csatorna és a periféria címéből, mindkettő nyolcbites, így összesen 256 periféria címezhető.

### 5.7.2 ADATÁTVITEL SÍNRENDSZERREL

A sínrendszerű átvitel alapelveiben lényegesen különbözik a csatornarendszertől. Míg a csatornarendszerben központi, hierarchikus irányítás vezérli a műveleteket, a sínrendszer esetén bármely, a rendszerre kapcsolódó funkcionális egység vezérelheti a rendszert.



5.10. ábra: A mikroszámítógép sínrendszerre épülő struktúrája

A központi vezérlőegység, a perifériavezérlők és egyéb csatlakozó egységek egy közös adatátviteli berendezésre, a sínrendszerre kapcsolódnak. A sínrendszer címvonalak, adatvonalak, vezérlők összessége.

Funkcionálisan a sínrendszerre csatolt összes egység egyenrangú, bármelyik irányíthatja a teljes forgalmat. A vezérlés elve itt bonyolultabb, mint a csatornarendszer esetén, de számos előnye miatt korszerűbbnek mondható. Ezek az előnyök a következők:

- Lehetőség van a központi processzor és a perifériák közötti közvetlen átvitelre, az operatív tár igénybevétele nélkül. Ugyanígy megoldható a tár és a perifériák közötti közvetlen adatkapcsolat a processzor igénybevétele nélkül.
- Lehetőség van arra, hogy egyes perifériavezérlők közvetlenül cseréljenek adatokat, a CPU igénybevétele nélkül.
- AZ I/O utasítások megszűnnek a sínrendszerű átvitelnél, ugyanis a perifériavezérlők regiszterei ugyanúgy kezelhetők programból, mint az operatív tár rekeszei.

A sínrendszerben az átvitel koordinálása a megszakítási rendszerrel történik.

## 6 CSATOLÓ- ÉS BŐVÍTŐKÁRTYÁK

### 6.1 MONITORVEZÉRLŐ KÁRTYÁK

A monitorok meghajtásához szükséges áramkörök egy kártyán helyezkednek el. A monitorvezérlő teszi lehetővé, hogy a monitoron megjelenhessenek azok az információk, melyeket a számítógép közöl a felhasználókkal. A kártyákon található egy grafikus memória, mely azt a célt szolgálja, hogy a képképzés zökkenőmentesen történjen meg. A memóriát a PC felől lehet írni és olvasni egyaránt. Az újabb memóriákat már egyidőben lehet írni és olvasni is. Minden vezérlőn található egy céláramkör, mely a monitor vezérlését látja el.

#### 6.1.1 MDA KÁRTYA

1981-ben jelent meg az MDA (Monochrome Display Adapter) kártya. Csak szöveges megjelenítésre képes, és azt is csak 2 színben. A memória mérete 4 kbyte, mellyel a lehetséges felbontás 80x25 karakter. Egy karakter 9x14 képpontból áll.

Az MDA csatlakozóinak jele digitális és TTL jelek. A katódsugarat két frekvenciával vezérlik. Az egyik a képfrekvencia (függőleges szinkron) a másodpercenkénti képváltások számát adja meg. Minél nagyobb ez az érték, annál stabilabb a kép. A másik vezérlőfrekvencia a sorfrekvencia (vízszintes szinkron), ez egy képpont sor kiírásának idejét határozza meg. Minél gyorsabban történik egy sor kiírása, annál nagyobb lehet a kép felbontása.

#### 6.1.2 CGA KÁRTYA

Ez a típus az MDA kártya továbbfejlesztése. A CGA (Color Graphics Adapter) kártya már grafikus üzemmódra is képes. Ahhoz, hogy ez megoldható legyen, a memória méretét 16 kbyte-ra kellett növelni. A kártya képes karakteres üzemmódra is. A CGA karakterek 8x8 képpont méretűek, ezzel 80x25 vagy 40x25 karakteres mód válik elérhetővé.

A memória első byte-ja a képernyő bal felső sarkába kerülő karakter ASCII kódját tartalmazza. A következő címen a karakter megjelenítési módja van. A harmadik byte tartalmazza az első sor második oszlopába kerülő karaktert, a negyedik pedig a második karakter attribútumát. Ez így folytatódik, amíg a képernyőt ki nem töltik a karakterek. Így 80x25 karakter kirajzolásához 4000 byte memória kell. A páros címeken mindig a karakterkód, míg a páratlan címeken az attribútum található. Az attribútum byte értelmezése:

- 0-2. bit Előtér színe kódolva
- 3. bit Növelt fényerő engedélyezése
- 4-6 bit Háttér színe kódolva
- 7. bit Karaktervillogás engedélyezése

A CGA adapternek két különböző grafikus üzemmódja is van. A nagyobb felbontásban 640x200 képpont 2 színben használható. Alacsonyabb felbontásban 320x200 képpont és 4 különböző szín jeleníthető meg. Az első esetben minden képponthoz 1 bit tartozik, így ez 128000 bitet igényel. Ez valamivel kevesebb, mint 16 kbyte. Ha azonban négy színt használunk, akkor a pontok tárolásához két bitre van szükség. Így a felbontás csak fele lehet.

#### 6.1.3 HERCULES KÁRTYA

Megjelenése a CGA kártyával egyidőben történt. A monitorvezérlőn kívül a kártyára került egy párhuzamos port is. Felbontása grafikus üzemmódban 720x348 képpont. Szöveges módban a karakterek 9x14 képpont méretűek és 80x25 jeleníthető meg.

A képernyő memória 64 kbyte méretű. Két grafikus kép támogatására képes. Ehhez a memóriát két részre kell osztani. Egy lap 4 további bankra van felosztva a memória-elérés gyorsítása érdekében. Minden adatra egy adatmutató mutat. Ahhoz, hogy ezt keveset kelljen moz-

gatni, az adatok az egyes bankokba párhuzamosan kerülnek beírásra. Ez azt jelenti, hogy az első adat az első bankba, a második a másodikba, a negyedik a negyedikbe. Az adatmutatót csak ezután kell növelni. Így ezt csak 4 adat mozgatása után kell megtenni.

A Hercules kártya sajátossága, hogy egy VGA kártyával képes egyszerre működni.

#### 6.1.4 EGA KÁRTYA

Az EGA (Enhanced Graphics Adapter) a CGA továbbfejlesztése. Maximális felbontóképessége 640x350 képpont, amit 16 különböző színnel képes megjeleníteni. A 16 színt 64 színből választhatjuk ki. A képinformációk tárolásához 256 kbyte memória áll rendelkezésre. Karakteres üzemmódban a karakterek 8x14 képpont méretűek, a sorok számát 43-ra növelhetjük.

A kártya képes a programok által igényelt üzemmódokat automatikusan felismerni és átkapcsolni közöttük.

A képernyő memóriája 64 kbyte-os memóriablokkokra van felosztva. A karakteres módban az egyes blokkokat a következőképpen használja a kártya:

0. blokk: Megjelenítendő karakterek ASCII kódjai, melyek több oldal számára elegendőek.
1. blokk: Megjelenítendő karakterek attribútuma. Attribútumok lehetnek: szín, villogás, fényerő.
2. blokk: Letölthető karakterfontok (maximum 4 font). Ezzel lehetővé válik akár a magyar ékezetes karakterek megfelelő kiírása a képernyőre.
3. blokk: Nincs felhasználva.

A letölthető fontoknak 32 byte bittérkép áll rendelkezésre. Grafikus üzemmódban a memóriablokkok feladata egy kicsit más:

0. blokk: Kék szín, képpontonként 1 bit.
1. blokk: Zöld szín, képpontonként 1 bit.
2. blokk: Vörös szín, képpontonként 1 bit.
3. blokk: Fényerő, képpontonként 1 bit.

Ha megvizsgáljuk a memóriakiosztást, akkor látható, hogy minden blokkból egy bit tartozik minden képponthoz. Ez a négy bit a 16 palettaregiszter egyikét választja ki. Ennek a regiszternek a feladata a képpont megjelenítése. A palettaregiszterek 6 bitesek, minden színhez 2-2 bit tartozik. Mivel minden szín 4 fényerővel jeleníthető meg, ebből adódik a 64 különböző szín. Azonban egyszerre mindig csak 16 használható.

#### 6.1.5 VGA KÁRTYA

Az EGA kártya által megjeleníthető színek száma túl kevés volt. Több színt úgy lehet elérni, ha a színek eltérő fényerő-kombinációját keverjük össze. Az emberi szem felbontása olyan, hogy a legkisebb érzékelhető színárnyalat közötti különbség 24 biten tárolható. Itt minden szín fényerejét 8 biten tároljuk. Ezzel több mint 16 millió színárnyalat különböztethető meg. Az ehhez szükséges videócsatlakozónak már 27 lábba lenne szüksége. E megoldás helyett a kártya a színjeleket analóg kimeneteken továbbítja a monitor felé. A csatlakozón három színjel hordozza a színkeverés alapszíneit egyenfeszültséggel. A vonalon lévő 0 V sötét, míg a +0,7 V teljes fényerőt jelent. A közbenső fokozatok számára nincs megkötés.

A VGA kártya 640x480-as felbontásban tud megjeleníteni 16 színű grafikát. Minden szín egy 262144 színárnyalatot tartalmazó palettáról választható. A kártya a palettaregiszter tartalmát egy digitális-analóg átalakítóra küldi. A DAC 256 db 18 bites regisztert tartalmaz, melyből a palettaregiszter választ. A regiszterek tartalmát alakítja a konverter analóg feszültséggé.

A versenyhelyzet következtében az egyes gyártók próbálták felülmúlni egymás fejlesztéseit. Ennek végeredménye az SVGA (Super VGA) kártya. Itt sokféle felbontás és színmélység áll rendelkezésre:

Felbontás	Színmélység
640x480	16,7 millió
800x600	32768
1024x768	256
1280x1024	16

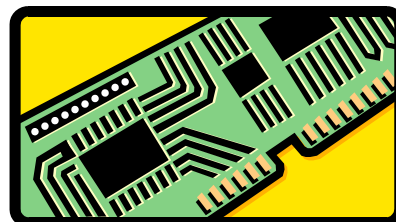
6-1. táblázat: SVGA felbontások és színmélységek

A felbontások igen nagyok. Ennek megfelelően a videómémória mérete igencsak megnőtt. A memóriaigényt kiszámíthatjuk, ha adott felbontásban a pixelek számát megszorozzuk a megjeleníteni kívánt színek számának ábrázolásához szükséges bitek számával.

Minden VGA kártyának van saját BIOS-a. Ez a bejelentkező üzenetet kiírja a képernyőre a számítógép indulása esetén. A BIOS olyan kialakítású, hogy a VGA kártya minden más kártya üzemmódjában beállítható.

A képernyő-memória a teljes videó címtartományt lefedi. Ebből különböző területeket használ a kártya attól függően, hogy milyen üzemmódban és milyen kártyaként üzemel.

A szöveges megjelenítés úgy történik, mint az EGA kártyánál. A palettaregiszterek kimenete nem közvetlenül a monitort vezérli, hanem a színkód regiszterekkel újra átkódolhatók. Ez a művelet kétféle módon történhet. Az első esetben a paletta regiszter kimenő 6 bitjét a szín-kiválasztó regiszter 2 bittel egészíti ki. Ezzel a megoldással 4 színkészletet tudunk tárolni és gyorsan változtatni. A másik esetben a paletta regiszter 4 bitjét egészíti ki a kártya szín-kiválasztó 4 bitjével. Ekkor már 16 színkészlet gyors váltását adhatjuk meg.



A VGA kártyával 512 különböző karakter egyidejű megjelenítésére van lehetőség. Ez azonban nem meríti ki a lehetőségeket, mert egyszerre 8 különböző programozható karakterkészletet használhatunk fel.

A VGA videokártyák nagyobb felbontásban is képesek nagy színmélységet elérni. Ekkor azonban a memória méretét meg kell növelni.

### 6.1.6 GYORSÍTÓKÁRTYÁK

A grafikus rendszerek elterjedésével a videokártyák sebessége már nem megfelelő. Ennek az a magyarázata, hogy az ablakok megnyitása, a menüpontok kiválasztása esetén mindig újra kell rajzolni a képet. A műveletet gyorsítani kell, melynek egyetlen hatékony módja a hardveres gyorsítás.

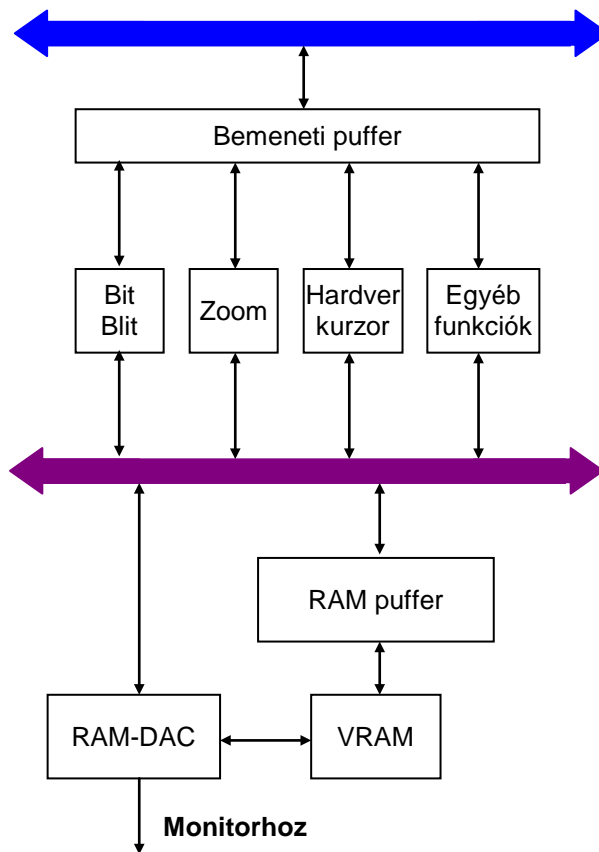
1992-ben jelent meg az első gyorsítókártya. Az elképzelés szerint egy intelligens processzor átveszi a számítógép CPU műveleteinek azon részét, mely a képalkotással foglalkozik. A grafikus operációs rendszerek gyakori eljárásait hardveresen oldják meg.

Ilyen eljárások a következők:

- ☞ *Bit-Blit*: Ablak eltolása. Egy téglalapot felvesz és egy másik pozícióban újrarajzolja.
- ☞ *Hardware Cursor*: Egérkurzor hardveres megjelenítése, a számítógép főprocesszora csupán a koordinátákat adja meg, a kártya önállóan kezeli a kurzort.
- ☞ *Line Drawing*: Vonalhúzás. A processzor csupán a kezdő- és a végpont koordinátáit adja át a kártyának.
- ☞ *Circle Drawing*: Kör rajzolása. Csupán a középpont koordinátáit és a sugár nagyságát adja át a CPU a kártyának.
- ☞ *Polygon Fill*: Terület kitöltése. Egy sokszöget a megadott mintával tölti fel.

Minden gyorsító kártya VGA kompatibilis. A gyorsító funkció DOS alatt szinte teljesen észrevehetetlen. Azonban egyes CAD és egyéb grafikus jellegű alkalmazásokban kihasználhatjuk a kártya által nyújtott előnyöket, de ehhez speciális, az adott programhoz készített szoftverre van szükség.

Minden kártya kimenete és a csatlakozó kiosztása a VGA kártyáéval teljes mértékig azonos.



6.1. ábra: VRAM-ot tartalmazó kártya felépítése

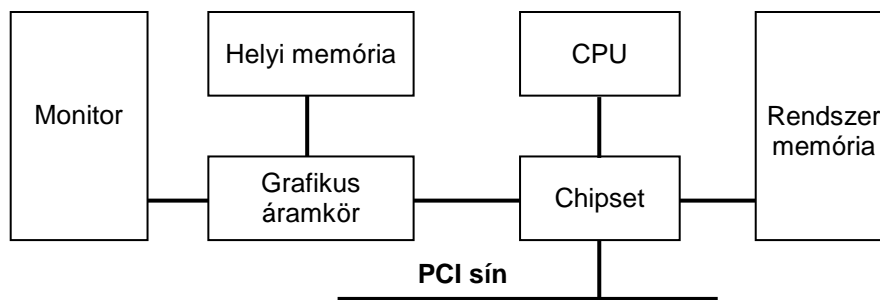
A kártya központi eleme a grafikus gyorsító áramkör. Ez egy speciálisan erre a célra optimalizált processzor. A monitorok analóg jelekkel működnek, viszont a számítógépek digitálisak. Ezt hidalják át a digitális-analóg átalakítók. Ezek a digitális jeleket RGB analóg jelekké alakítják. Ahhoz, hogy a kártyát képes legyen a CPU elérni, szükség van egy áramkörre, amely a címdekódolást elvégzi. Mivel a rajzolt képek igen nagy mennyiségű adathalmazok, szükség van egy átmeneti tárolóra. Ez az egyszerű CGA kártyákon DRAM.

A DRAM saját adatporttal rendelkezik, melyen keresztül a processzorral és a digitális-analóg átalakítóval kommunikál. A grafikus áramkör vezérli a DRAM írását úgy, hogy az adatot a PC processzora küldi, a DRAM pedig továbbküldi azt a DA átalakítónak. Az eljárás így meglehetősen lassú, legalábbis a grafikus rendszerek számára. A gyorsítás érdekében a grafikus kártyákon elhelyeztek ún. videó RAM-ot is (VRAM).

A VRAM és a DRAM között nincs túl nagy különbség, a működés elve azonos. Azonban a VRAM két adatporttal rendelkezik (Dual Port RAM). Ezzel a megoldással lehetővé válik, hogy egyidőben kapjon adatokat a PC processzortól és küldjön már korábban kapott adatokat a DA átalakító felé.

### 6.1.7 AGP

Az AGP (Accelerated Graphics Port, gyorsított grafikus port) nagy előnye, hogy nem kell csak a kártya memóriájával megelégednünk. Ha szükség van a nagy méretű 3D-s képek kirajzolásához memóriára, akkor a kártya képes a rendszer memóriáját használni. Másik mód a gyorsabb működés eléréséhez, hogy a kártya tartalmaz egy kitüntetett szerepű portot, melyen keresztül közvetlenül, késleltető elemek közbeiktatása nélkül képes elérni a memóriát és a processzort. Az adatátvitel sebessége a rendszer órajelével egyezik meg. Természetesen a memóriának nagyon gyorsnak kell lennie, ezért erre a célra csak az SDRAM megfelelő.



6.2. ábra: Az AGP rendszer felépítése

Az AGP port úgy képes elérni az 533 Mbit/s átviteli sebességet, hogy a memória eléréséhez egy új technikát használ. Ennek a DIME (Direct Memory Execute – direkt memória végrehajtás) nevet adták. A nagy képeket az AGP a rendszermemóriában tárolja. A gyors kiolvasáshoz szükséges a RAM közvetlen elérése. A kiolvasás során a képek digitális leképezései (textúrák) átkerülnek a kártyán lévő átmeneti tárolóba. Ezek szinte kivétel nélkül videó memóriák. Innen a képet a grafikus áramkör küldi el a monitornak. A chip azokat a képeket, amelyek már nem férnek el a saját memóriájában, a processzor megkerülésével közvetlenül a RAM-ban tárolja el. Innen bármikor előhívhatja, ha szükséges.

## 6.2 A MULTI IO KÁRTYA

A multi IO kártyát nevezik a szakzsargonban IDE kártyának. A kártya feladata a rendszer IO eszközeinek összekapcsolása a rendszersínnel és ezzel együtt a processzorral. Az IDE kártya az alábbi részegységeket tartalmazhatja:

- ☞ merevlemez vezérlő
- ☞ hajlékonylemez vezérlő
- ☞ soros port
- ☞ párhuzamos port
- ☞ játékport

### 6.2.1 MEREVLEMEZES VEZÉRLŐ

Az adatátviteli sebesség növelésének egyik módja, hogy a lemez és a meghajtó elektronika közötti távolságot lecsökkentjük. Ennek megfelelően akkor a legjobb a helyzet, ha magán a meghajtón van a vezérlő elektronika. Ebből az elgondolásból született meg az IDE (Integrated Device Equipment, integrált eszköz-elektronika) rendszer. Az ilyen meghajtóban az elektronika is megtalálható, ezzel a megoldással a nagysebességű jelvezetékek hossza minimális. Így csak a bővítő sín jeleit kell elvinni. Ezen a 16 bites párhuzamos interfészen a jelek már jóval kisebb sebességgel áramlanak. Az IDE meghajtók 1-4 Mbyte/s adatátviteli sebességre képesek.

Minden vezérlő rendelkezik egy önálló processzorral, mely a vezérlési műveleteket hivatott ellátni. Ez a processzor képes arra is, hogy a meghajtó paramétereit figyelemmel kísérje, és hiba esetén megpróbálja korrigálni azt.

Az IDE csatoló két IDE interfészes egység kezelését teszi lehetővé. Az egyik meghajtót MASTER-nek, míg a másikat SLAVE-nek kell beállítani.

Az illesztőkártya egy 40 eres szalagkábelrel van összekötve a merevlemez meghajtóval. A kábelben a bővítő sín egyes jeleit találjuk meg. Az egyetlen kiegészítő jel a meghajtóhoz való hozzáférést jelző LED vezérlővezetéke. Az interfész áramkörének a felépítése meglehetősen egyszerű, a címdekóderen és leválasztó áramkörön kívül nem is tartalmaz mást. Egyes illesztőkön lehetőség van egy gyorsítótár kialakítására.

### 6.2.2 FLOPPY ILLESZTŐ

A floppy illesztő a hajlékonylemez meghajtók illesztésére szolgál. A legelterjedtebb a 3,5", 1,44 MB tárolókapacitású „kislemez meghajtó”. A mai vezérlők már az összes funkciót megvalósító nagy integráltságú áramkörrel készülnek, így a kártya felépítése nagyon egyszerű.

A kártyákkal 120, 300 és 500 kbit/s adatátvitel valósítható meg. Egy kártya négy meghajtó csatlakoztatását teszi lehetővé.

### 6.2.3 SOROS VONALI ILLESZTŐ

A soros vonali illesztőt Amerikában az RS-232C, míg Európában a CCITT V24/V28 szabvány jelöli. A soros interfész soros működésű. Ez azt jelenti, hogy az adatbitek egy vezetéken keresztül egymás után megadott ütemben kerülnek átvitelre. Egy vezetéken egyirányú átvitel valósítható meg. Az adatátvitelnek szigorú szabályai, ún. protokollja van. Az interfész nem TTL szintű jeleket használ. Az RS-232C-ben a 0 logikai szintnek +3 és +15 V, a logikai 1 szintnek -3 és -15 V közötti értékek felelnek meg. A nagy tűrésű tartomány a zavarérzékenységet növeli. A PC-k esetében a feszültségszintek nem érik el a -15 és +15 V-ot. Az RS-232C interfész táplálása +5 V-ról történik, a -12 és +12 V-ot használja fel segéd feszültségnek. Minden interfész rendelkezik egy szintillesztő fokozattal, mely az említett segéd feszültségek-ből állítja elő a vonali feszültségeket. Így a gyakorlatban a kimeneteken 9 és 12 V közötti értékeket tudunk mérni.

A V24-es interfészen az adatátvitel szinkron vagy aszinkron módon történik. A szinkron adatcsere esetében az adó és a vevő különböző sebességgel működhet, mert külön vezetékkel (szinkron jellel) jelzik egymásnak az érvényes adat megjelenését az adatvezetéken. Aszinkron üzemmód esetében nincs ilyen szinkronjel, az adatvonalon lévő adatjelek maguk végzik el a szinkronizációt. Az átvitelnél az adó és a vevő közel azonos frekvenciájú órajelet használ. Az egyszerre átvitt adatmennyiség 5, 6, 7 vagy 8 bit lehet. Az adatfolyamot egy START bit előzi meg, mely egy 1 bitnyi ideig tartó alacsony szintű jel. Ezt követik az adatok, először a legelső helyiértékű bit kerül átvitelre. Az adatokat követi a paritásbit, mely el is hagyható. A paritásbit után egy vagy két STOP bit következik. Ezek tartoznak egy adatkeretbe (frame). A szinkronizálás az átvitel elején, a START bittel kezdődik, az adó és a vevő órájának annyira kell együtt futnia, hogy a szinkronból ne essenek ki egy keret átvitele alatt. A paritásbit az adatátvitel biztonságát növeli. A STOP bit vagy bitek feladatai a keret lezárása. Ezzel az egymás után küldött kereteket is szétválasztja.

Mikor adatokat viszünk át egyik helyről a másikra, nem mindegy annak sebessége. Az adatátvitel sebessége a bit/s vagy ennek logaritmikus formája a baud. Az átviteli sebesség a soros átvitelnél csak diszkrét értékeket vehet fel. Tipikus értékek:

56 75 110 300 600 1200 2400 4800 9600 19200 38400 bit/s

Az adatátvitel biztonságát sok tényező befolyásolja. Minél hosszabb a kábel, annál több zavarjelet képes összeszedni, ezért hosszabb kábel kisebb biztonságban képes az adatokat



átvinni. Nem mindegy a jelszinteket jelentő feszültségek értéke sem. A túl nagy vagy a túl kis érték csökkenti a biztonságot az átvitel során. Minél nagyobb sebességű az adatátvitel, annál érzékenyebb lesz az átvitel a környezeti zavarokra.

Az átvitel kezdetekor az adatátviteli paramétereket egyeztetni kell, pontosabban ugyanazokat kell beállítani mind az adónál, mind a vevőnél. A START bittel nem kell foglalkozni, mert az mindig egy bit hosszúságú. A STOP bitek száma 1, 1,5 és 2 lehet. Az adatbitek száma PC-k esetében szinte kivétel nélkül 8 bit.

A paritásbit helyzete nem ilyen egyszerű. Öt lehetséges kombinációja van a paritásbit használatának:

1. Nincs paritásbit.
2. A paritásbit mindig alacsony szintű (0).
3. A paritásbit mindig magas szintű (1).
4. Páros paritás. Akkor beszélhetünk páros paritásról, ha a paritásbit úgy egészíti ki az adatblokkban lévő 1-eseket, hogy páros számút tartalmazzon.
5. Páratlan paritás. Ebben az esetben a paritásbit olyan értéket vesz fel, hogy az adatblokkok 1-eseivel együtt páratlan számú 1-es legyen.

A vevő újra kiszámítja a paritásbitet és ha nem egyezik a vett és a számított érték, az azt jelenti, hogy hiba történt az átvitel során. Egy paritásbittel páratlan számú bithibát lehet felismerni.

A PC BIOS-a négy soros interfész használatát támogatja. Ezek neve rendre COM1-COM4.

#### 6.2.4 PÁRHUZAMOS ILLESZTŐ FOKOZAT

A párhuzamos interfész minden PC-ben megtalálható. A régebbi interfészek csak egyirányú adatátvitelre alkalmasak, az újabb párhuzamos csatolók már két irányban képesek kommunikációra.

A párhuzamos interfész párhuzamosan küldi az adatokat a vevő felé, az adatszéllesség 8 bit. A párhuzamos interfészek minden jele TTL kompatibilis. Az adatátvitel nyugtázása két módszer szerint történhet. Ennek megfelelően két- és háromvezetékes protokollt alkalmazhatunk.

Ha a vevő rendelkezik egy átmeneti tárral, addig az átvitel nagyon gyors, ha azonban ez megtelik, meg kell várni míg kiürül.

A BIOS négy párhuzamos port használatát támogatja. Ezek elnevezései LPT1-LPT4.

Induláskor a BIOS felismeri a párhuzamos interfészeket. A képernyőre kiírja a portok nevét, valamint a címét.

#### 6.2.5 BOTKORMÁNY ILLESZTŐ

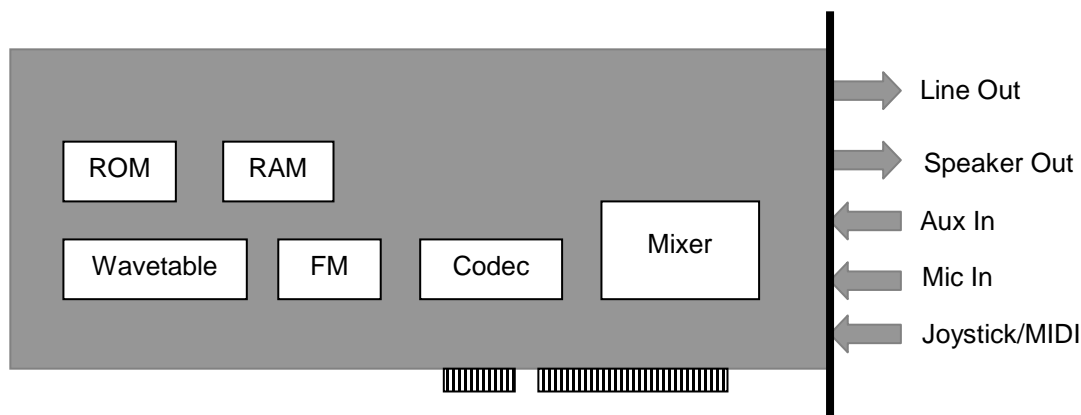
Az illesztőt nevezik játék portnak (game port) is.

A PC botkormányok csak olyanok lehetnek, melyek elmozdulását potenciométer érzékeli. A kártya négy kitérés és négy kapcsoló bemenettel rendelkezik. Ezzel a módszerrel két két-dimenziós vagy egy négydimenziós egység kezelését oldja meg. A kétdimenziós botkormány két kapcsolóval rendelkezik és egy-egy érzékelő bemenetet használ fel az X és az Y irányba történő elmozdulás érzékeléséhez. A kapcsoló bemenetekre érkező logikai szint közvetlenül kiolvasható. A potenciométerek pillanatnyi értékét egy multivibrátorral impulzussorozattá alakítja a kártya. A kiadott impulzus ideje arányos az ellenállás értékével. A kapcsoló bemenetek 1 k $\Omega$ -os ellenállásokkal vannak a tápfeszültségre kötve. Ha a kapcsolókat zárjuk, a bemenetek alacsony szintre kerülnek. A potenciométerek 100 k $\Omega$  értékű, középmegecsapolásos kialakításúak.

### 6.3 A HANGKÁRTYA

A számítógép digitális működéséből következik, hogy a hangot is digitális formában kell tárolni. Ehhez szükség van egy analóg-digitális átalakítóra. Ha viszont a számítógépen tárolt

vagy előállított hangot szeretnénk megszólaltatni, akkor egy digitális-analóg átalakítót kell használnunk.



6.3. ábra: A hangkártya vázlatos felépítése

A *Wavetable* (hullámtábla) elnevezésű egység végzi el a hangmintákkal kapcsolatos műveleteket és átadja a feldolgozott mintákat a CODEC-nek.

A *ROM* tartalmazza a hangkártyával adott hangmintákat gyárilag programozva. Mivel ezt csak olvasni tudjuk, ezért itt nincs lehetőségünk saját mintákat eltárolni.

Ha szeretnénk saját hangmintákat készíteni és azokat később a hangalkotásnál majd felhasználni, akkor el kell tárolnunk a hangkártya saját *RAM*-jában. Ennek mérete alapvetően meghatározza az eltárolható hangminták számát.

Az *FM* jelzésű blokk az FM szintézis megvalósításáért felelős.

Az analóg-digitális és a digitális-analóg átalakítást a *CODEC* végzi el. Ez integrálva tartalmazza a mintavevő és tartó áramkört, az analóg-digitális és a digitális-analóg áramköröket egyaránt. A szűrők rendszerint külső elemekkel vannak megoldva azért, hogy a felhasználhatóság univerzális legyen.

A *Mixer* elnevezésű egység feladata kettős. Egyrészt a különböző forrásból származó jeleket a CODEC-re kapcsolja, másrészt a CODEC kimeneti jelét a meghatározott kimenetre kapcsolja. Ez tulajdonképpen egy analóg, fokozat nélküli kapcsoló, melyet programból vezérelhetünk. A fokozatmentes azt jelenti, hogy a kimenetek vagy a bemenetek kapcsolása úgy lehetséges, hogy egymásba úsztatjuk őket. Ez úgy jelentkezik, hogy az egyik fokozatosan elhalkul, míg a másik fokozatosan felerősödik.

Rendszerint a hangkártyák tartalmaznak egy kis teljesítményű erősítőt, melyre közvetlenül hangszóró kapcsolható. Az erősítő jellemző teljesítménye 4 W.

A kimenetek és bemenetek valósítják meg a kapcsolatot a hangkártya és a külvilág között. Ezek a következők lehetnek:

<i>Line Out</i>	Vonali, tehát kis szintű kimenet. A kimeneti feszültség általában maximum 0,745 V, mely alkalmas erősítő eszközre csatlakoztatásához.
<i>Speaker Out</i>	Egy erősített kimenet, mely lehetővé teszi hangszórók közvetlen meghajtását.
<i>Aux In</i>	Nagyjelű bemenet, mely külső eszköztől származó analóg jelek feldolgozását teszi lehetővé.
<i>Mic In</i>	Mikrofon bemenet. Mivel a mikrofonnak a kimeneti jele meglehetősen alacsony, ezért a bemenet tartalmaz egy erősítőt.
<i>Joystick/MIDI</i>	Botkormány vagy elektronikus hangszer csatlakoztatását teszi lehetővé.

## 6.4 SCSI CSATOLÓ

Ez az interfész nem csupán merevlemezek számára készült, hanem minden olyan eszköz számára, melyet felkészítettek rá. Szabványos felületet biztosít. Ilyen eszköz lehet a merevlemez, a CD-ROM olvasó, a scanner stb. Ezeket az egységeket egyetlen vezérlő kezeli. A SCSI a Small Computer System Interface angol szavak rövidítése, magyarul egyszerű számítógép csatolót jelent. A vezérlő elektronika az egységen található, mivel így lehet elérni a legnagyobb adatátviteli sebességet. A periféria vezérlését a számítógép processzora helyett a kártyán lévő processzor végzi el. Ez újabb sebességnövekedést eredményez.

A szabványos SCSI csatoló 50 pólusú szalagsatlakozó, amelyben 9 vezérlővezeték és 9 adatvezeték fut. Azért 9 az adatbitek száma, mert el van látva egy paritásbittel. A paritásbit a típusától függően úgy egészíti ki az adatbyte-ot, hogy az abban lévő egyesek száma páros (páros paritás) vagy páratlan legyen (páratlan paritás). A páratlan számú vezetékek egy kivételével földelések. Erre azért van szükség, mert a nagy adatátviteli sebesség miatt a zavarjelekre sokkal nagyobb mértékben válna érzékenyebbé.

A SCSI egységek gyakran tartalmazznak saját gyorsító memóriát, amelyben az adatokat átmenetileg tárolni lehet. Az adatok tényleges felhasználása alatt, melyet már az eszköz végez el, a többi egység már kommunikálhat a számítógéppel.

Minden egységhez tartozik egy SCSI cím. Ezt az egységeken átkötésekkel vagy DIP kapcsolókkal lehet beállítani.

Az adatforgalom minden pillanatban két egység között zajlik, azonban időben az egységek változhatnak. A SCSI egységek kezdeményezőként (Initiator) és célként (Target) dolgozhatnak. A *kezdeményező* adja ki az utasításokat és vezérli a sín működését. A *target* általában egy intelligens eszköz, mely a tényleges adatátvitelt végzi el. Egy készülék szerepe nem rögzített előre, az az átvitel során alakul ki. Abban az esetben, ha a sínen több kezdeményező készülék van, akkor előfordulhat, hogy ezek egy időben szeretnék átvitelt kérni. Ez ütközéshez vezet. Ennek feloldására a szabvány szigorú, kötött, fix prioritású kapcsolat-felvételi protokollt tartalmaz.

Az adatforgalom a sínen megadott ütem szerint történik. Az egyes ütemeket fázisoknak nevezzük. Az egyes fázisok a következők:

- |                                     |  |
|-------------------------------------|--|
| <i>Szabad fázis</i>                 | A sín szabad, nincs kijelölt egység, nincs adatforgalom. Ez a sín alapállapota.  |
| <i>Sínhozzáférési fázis</i>         | Ez alatt veheti át egy egység a sínvezérlési jogot. Ha több kezdeményező egység is van a sínen és egyszerre többen szeretnék a sínre vezérelni, akkor ebben a fázisban folyik a versengés. Ha csupán egy kezdeményező egység van a rendszerben, ez a fázis kihagyható. A fázis a sín szabaddá válása után minimum 100 ns várakozással kezdődhet el. Minden eszköz kérheti az adatátvitelt – mely rendelkezik ilyen tulajdonsággal. Ez a fajta hozzárendelés már egy prioritási sorrendet is képez, ahol a legnagyobb prioritása annak a készüléknek van, mely a legmagasabb helyiértékű adatbittel rendelkezik. Mindegyik egység figyeli a sín adatvonalain kialakuló kombinációkat és a legmagasabb prioritású nyeri el a sínvezérlés jogát. A versenyben vesztes készülékeknek maximum 650 ns áll a rendelkezésükre, hogy a sínről lekapcsolódjanak, a győztesnek minimálisan 900 ns-ot várnia kell, mielőtt a vezérlést megkezdene. |
| <i>Választási fázis (szelekció)</i> | A győztes kezdeményező kiválasztja a rendszer eszközei közül egy célt. A kiválasztott eszköznek lehetősége van arra, hogy eldöntse, target funkciót kér tőle a vezérlő vagy egy már általa megkezdett átvitel folytatása történik.   |

<i>Újraválasztási fázis (reszelekció)</i>	A fázisban a célegység újra felveheti a kapcsolatot egy korábban megkezdett művelet befejezése céljából. Csak olyan eszközöknél működik, amelyek rendelkeznek iniciator funkcióval.
<i>Parancsfázis</i>	Ebben a fázisban kap a target eszköz parancsot a kezdeményezőtől. Összesen 40 parancs van különböző paraméterekkel.
<i>Adatfázis</i>	Tulajdonképpen ebben a fázisban történik az adatok továbbítása, vagy fogadása.
<i>Állapotinformáció fázis</i>	Az állapotfázisban a célegység állapotáról kap információt a kezdeményező. A lehetséges jelentések a következők: <ul style="list-style-type: none"><li>☞ GOOD STATUS (hibamentes állapot): A cél végrehajtotta az utolsó parancsot, készen áll a következő fogadására.</li><li>☞ CHECK CONDITION (állapot ellenőrzés): Valami hiba történt, amelyet a kezdeményezőnek meg kell határoznia.</li><li>☞ BUSY STATUS (foglalt állapot): A cél pillanatnyilag el van foglalva egy parancs végrehajtásával, még nem lehet neki újabbat küldeni.</li></ul>
<i>Üzenetfázis</i>	Az üzenetfázisban szintén a célok tudnak üzeneteket küldeni a kezdeményezőnek. Van azonban néhány olyan üzenet is, amelyet a kezdeményező küld a célnak.

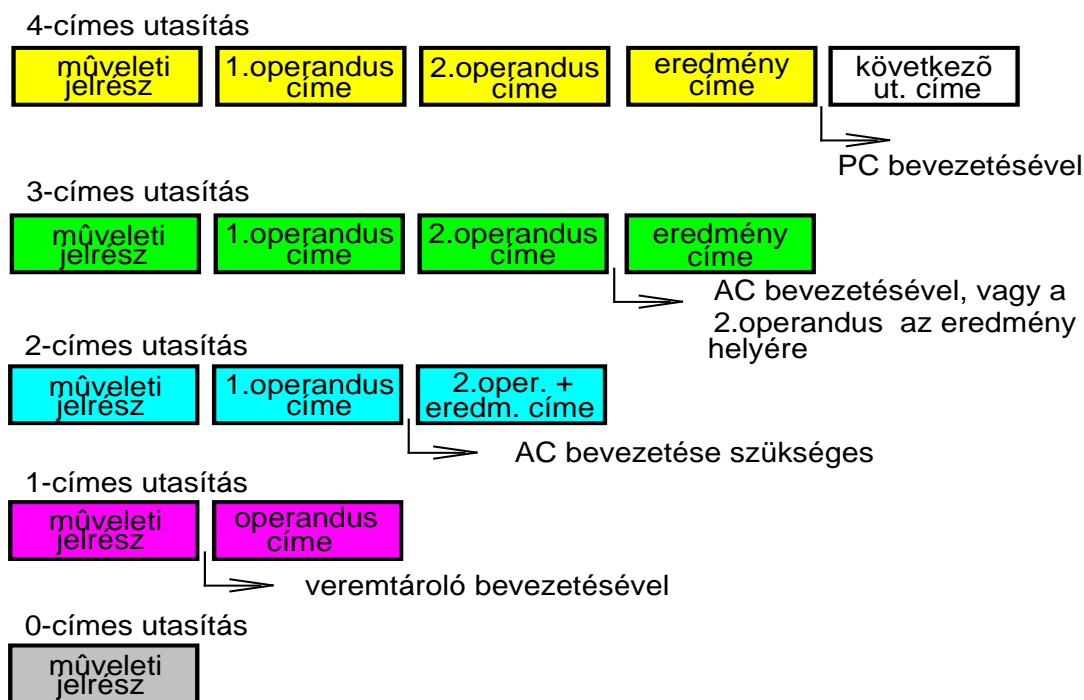
## 7 A PROGRAMOK VÉGREHAJTÁSA

A számítógépet a program működteti. A processzor közvetlenül csak a gépi kódú programokat képes értelmezni. A gépi kódú utasítás az elvégzendő *művelet kódjából* és a művelet *operandusaiból* áll. Maga az utasítás nem más, mint nullák és egyesek sorozata.

Az utasítások műveletikód-része adja meg a processzor számára, hogy milyen műveleteket kell elvégezni. Az operandusrész tartalmazza azt az információt, hogy a műveletet mivel kell elvégezni. A processzor által értelmezhető műveletek összességét a processzor *utasításkészletének* nevezzük.

### 7.1 GÉPI KÓDÚ UTASÍTÁSOK

A gépi kódú utasítások egy műveleti részből és egy operandusrészből állnak. Az operandusrész általában összetett.



7.1. ábra: Utasításszerkezetek

A legteljesebb utasításszerkezet a műveleti kódon kívül négy címet tartalmaz:

- ☞ első operandus címe
- ☞ második operandus címe
- ☞ eredmény címe
- ☞ következő végrehajtandó utasítás címe

Az utasítástól függően az első két cím helyett közvetlen adatok is szerepelhetnek, ekkor a műveletet a gép ezekkel végzi el. Általában nem célszerű közvetlen adatok használata a programokban, mert azokat nem tudjuk változtatni, míg a tárban az adatokat a programtól függően tetszés szerint változtathatjuk.

Az operatív táron kívül a vezérlőegységben van néhány speciális rendeltetésű tárolórekesz, ún. *regiszter*. A regisztereknek saját címük van, ezek is szerepelhetnek az utasítások operandusrészeiben.

A 4 címes és 3 címes utasításszerkezetek nagy hibája a helyfoglalás.

A korszerű számítógépek utasításai kétcímesek, azaz két operandushivatkozást tartalmaznak.

A kétcímes utasítások szerkezete:

- ☞ műveleti kód
- ☞ első operandus címe

A regisztereknek saját címük van, ezek is szerepelhetnek az utasítások operandusrészében.

Az eredmény címe lehet az operatív tár egyik rekeszének címe, de ha az eredményre a továbbiakban is szükség van, hagyhatjuk az ALU egy regiszterében is.

Ebben a rendszerben az utasítások nem jelölik ki a következő utasítást. Ehelyett a processzor egy speciális rekeszben (*utasításszámláló*) tárolja a következő végrehajtandó utasítás címét, és az aktuális művelet elvégzése után a következő utasítást az utasításszámlálóban lévő cím alapján választja ki. Ehhez egyrészt az szükséges, hogy egy program utasításai a memóriában egymás után, a végrehajtás sorrendjében helyezkedjenek el, másrészt a következő utasítás tárcímének kiszámításához a processzornak ismernie kell az aktuális utasítás hosszát.

Egyes utasításoknak az a célja, hogy megváltoztassák a programutasítások természetes, tárolási sorrend szerinti végrehajtási sorrendjét. Az erre a célra szolgáló utasításokat *vezérlésátadó* vagy *ugró utasítások*nak nevezzük. Ezek az utasítások a következő végrehajtandó utasítás címét töltik az utasításszámlálóba.

A kétcímes utasításokban nincs megadva az eredmény címe sem. A szokás az, hogy az eredmény az 1. operandus helyére kerül.

Egyes (pl. 8 bites) processzoroknál az utasítások egycímesek. Ilyenkor az egyik operandus az ALU egy speciális regiszterében (akkumulátor) található, a másik az operandusrészben megadott tárcímen, az eredmény pedig az akkumulátorba íródik.

A *címregiszter* és az *adatregiszter* is fontos szerepet játszik az utasítások végrehajtása során. Mindkét regiszter az ALU és az operatív tár közötti kommunikáció céljára szolgál. amikor a *vezérlőegység* egy adatot akar betölteni az aritmetikai és logikai egységbe, akkor a megfelelő tárrekesz címét előzetesen betölti a címregiszterbe, az olvasási művelet a címregiszter tartalmát használja a kívánt rekesz megkereséséhez.

Az olvasási művelet során az adat a tárrekeszből az adatregiszterbe kerül, majd a vezérlő innen írja át az aritmetikai és logikai egységbe. Amennyiben az ALU-ból kell adatot a tárba írni, a folyamat fordított irányban zajlik le, az adatregiszteren keresztül a címregiszter felhasználásával.

## 7.2 A CÍMZÉSI MÓDOK

### 7.2.1 A KÖZVETLEN ADATHIVATKOZÁS

Az operandus helyén nem egy adat címe, hanem maga az adat áll:

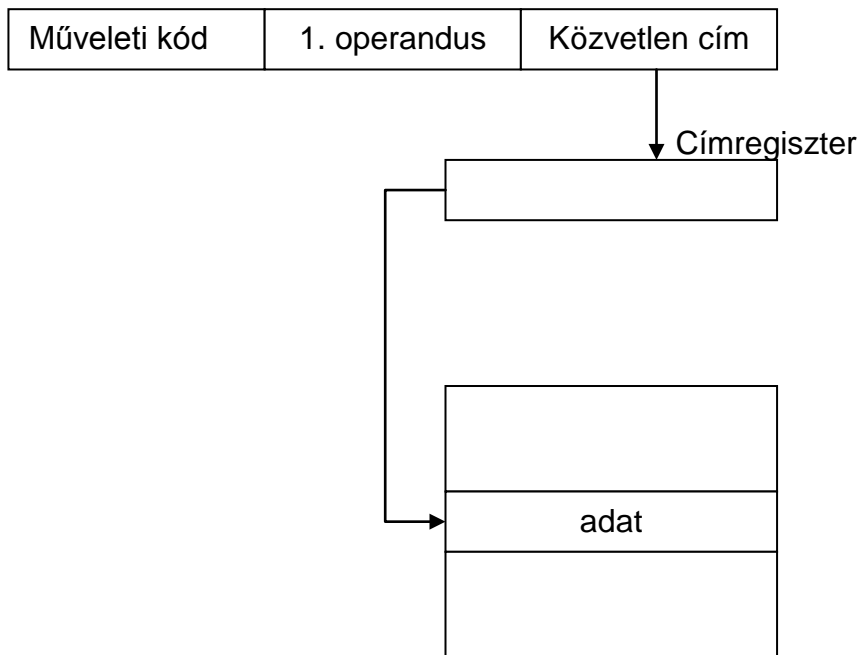
Műveleti kód	1. operandus	Adat
--------------	--------------	------

7.2. ábra: Utasításformátum közvetlen adathivatkozás esetén

Azt, hogy a 2. operandus helyén nem cím, hanem adat áll, a vezérlőegység a műveleti kódból tudja meg. Kétooperandusos műveletek esetén az első operandus nem lehet közvetlen adat, csak tárcím vagy regiszterhivatkozás.

### 7.2.2 A KÖZVETLEN CÍMZÉS

Az operandus itt egy közvetlen tárcím, azaz egy bináris szám.

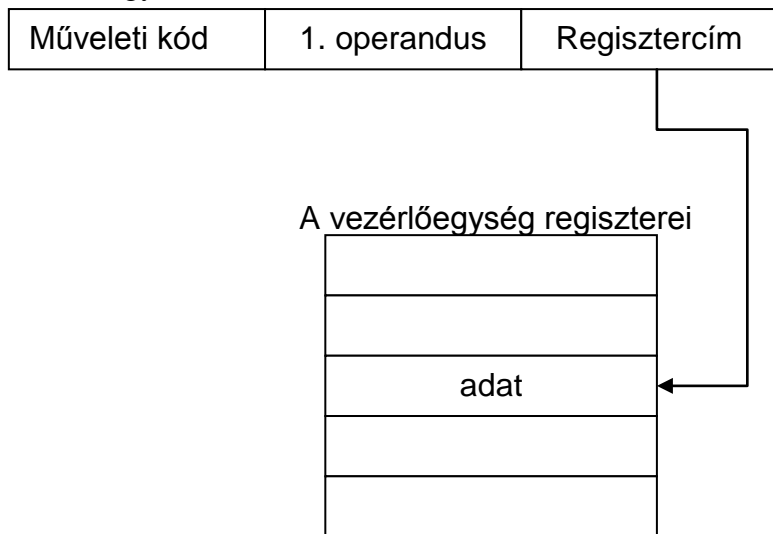


7.3. ábra: A közvetlen címzés

A közvetlen cím a regiszterbe kerül, a vezérlő megcímzi a kívánt rekeszt, a címzési mód hátránya, hogy a címrész nehezen módosítható, ezen kívül hosszú. Az utasítás mindig ugyanarra a tárrekeszre vonatkozik.

### 7.2.3 A REGISZTERES CÍMZÉS

A regiszteres címzésnél az operandus nem az operatív tárban van, hanem a processzor egyik regiszterében, így az adatért nem kell a tárhoz fordulni.

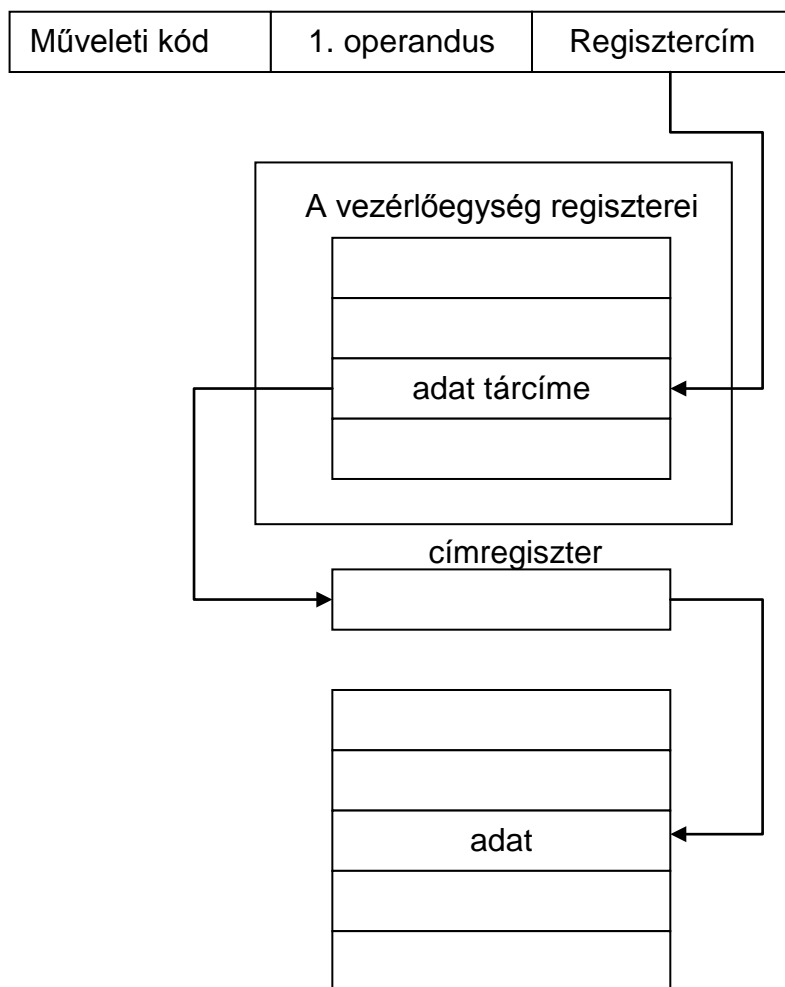


7.4. ábra: A regiszteres címzés

### 7.2.4 A REGISZTERES INDIREKT CÍMZÉS

A indirekt címzés azt jelenti, hogy az utasítás operandusa egy olyan címre való hivatkozás, amelyen a tényleges adat címe található.

A regiszteres indirekt címzésnél az operandus annak a regiszternek a címe, amelyben az adat tárbeli címe található.

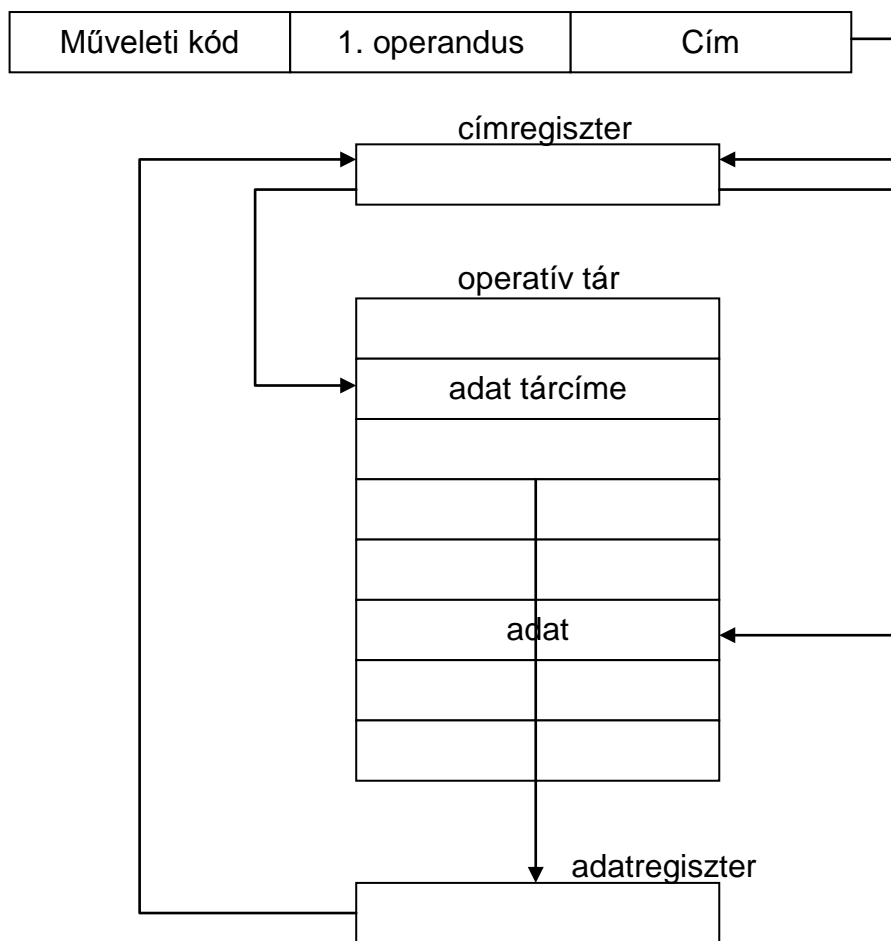


7.5. ábra: Regiszteres indirekt címzés

### 7.2.5 AZ INDIREKT CÍMZÉS

Ennél a címzési módnál az operandus olyan tárcímre mutat, amelyen az adat címe található. Itt egy adat eléréséhez kétszer kell a tárhoz fordulni. Ez az utasítástípus akkor használatos, ha a gép olyan, hogy az utasítás operandusrésze nem elég hosszú a teljes tár címzésére.



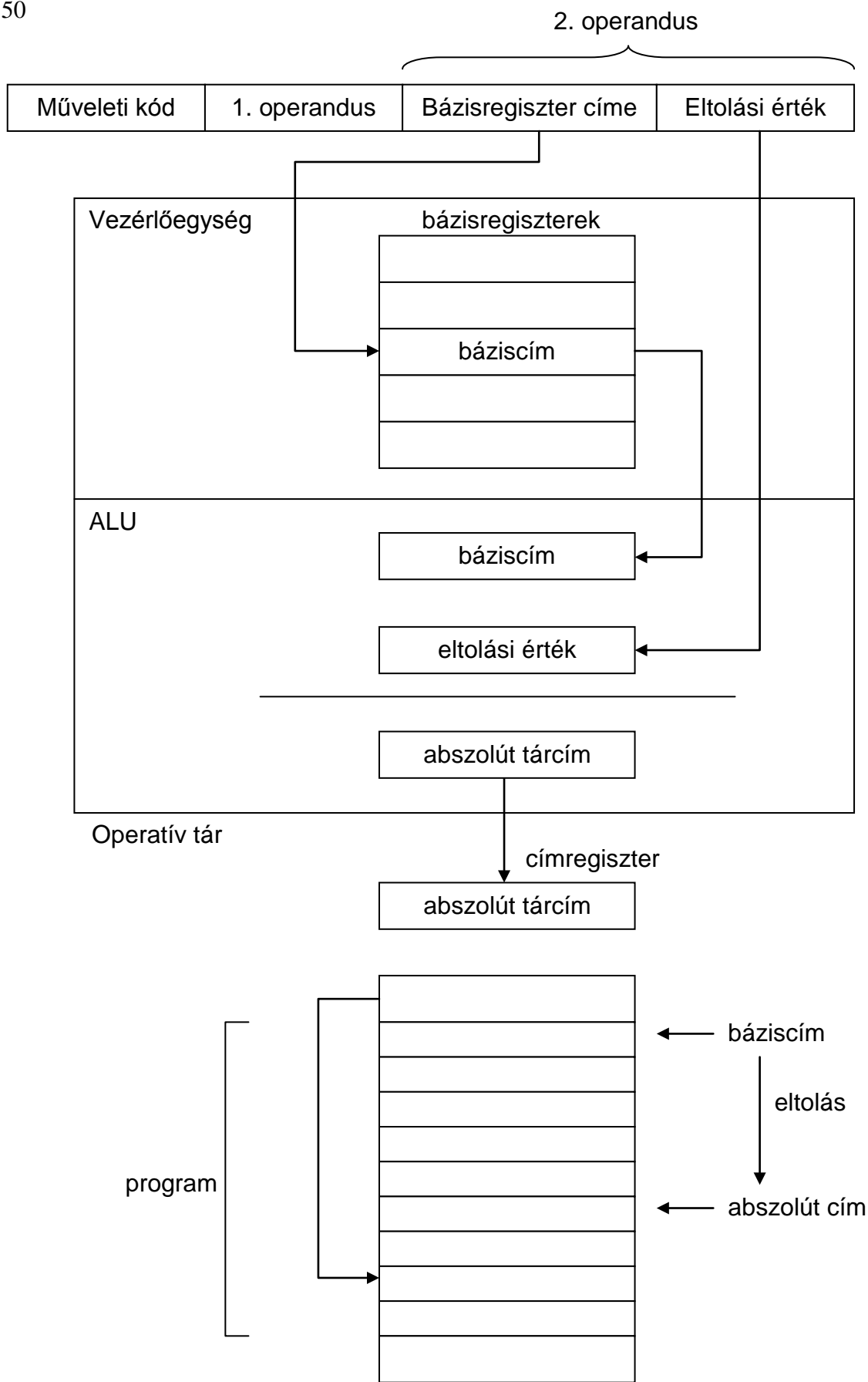


7.6. ábra: Indirekt címzés

### 7.2.6 A BÁZISREGISZTERES CÍMZÉS

A bázisregiszteres címzési módnál az adat tárcíme két címkomponens összege. Az egyik a vezérlőegység egy regiszterében, a bázisregiszterben található címkonstans, az ún. *báziscím*, a másik az ún. *eltolási cím*. Az utasításban a bázisregiszteres címzést alkalmazó operandus két részre oszlik. Az első részben található a bázisregiszter címe, a másodikban az eltolási érték. A tényleges cím kialakításához az ALU összeadja a báziscímet és az eltolási értéket.

A bázisregiszteres címzés az alapja a programok áthelyezhetőségének. A programok áthelyezhetősége, azaz a lehetőség, hogy a programok a tár bármely részében azonosan működhetnek, a feltétele annak, hogy a gépet egyszerre több program használhassa. A programozó minden utasításban egy bázisregiszterre hivatkozik, az operandusok címét pedig a program első utasításától való távolságuk adja meg (relatív cím). A program betöltésekor a gép a bázisregiszterbe tölti az első utasítás címét, ezután az *abszolút tárcím* minden bázisregiszteres címzési utasításnál a bázisregiszter tartalmának és a *relatív címnek* az összege.

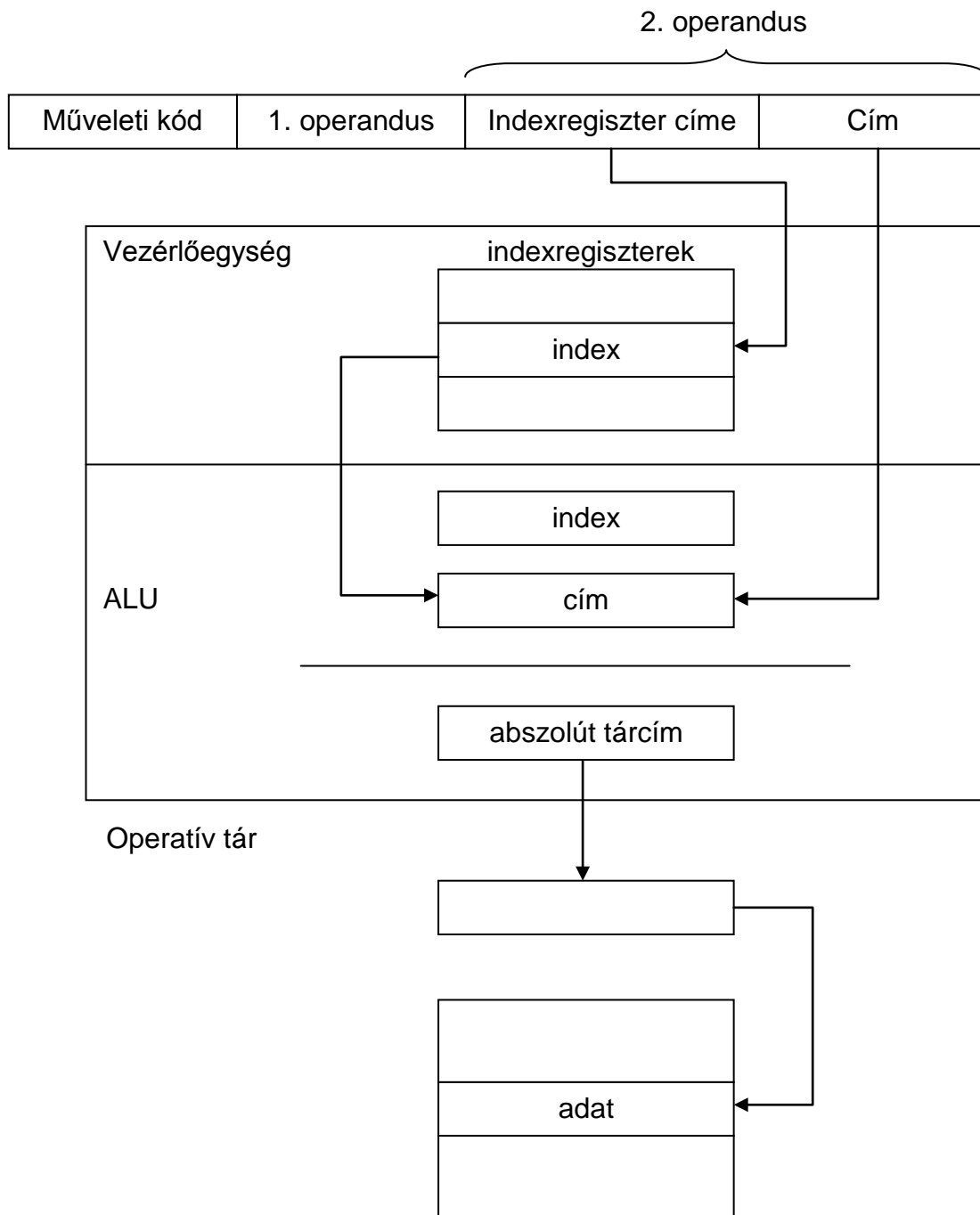


7.7. ábra: A bázisregiszteres címzés

### 7.2.7 AZ INDEXREGISZTERES CÍMZÉS

Az indexregiszteres címzés során az abszolút cím a vezérlő egy regiszterében, az ún. *indexregiszterben* lévő cím és az operandusban lévő cím összege. Az indexregiszter címe szintén ugyanabban az operandusban van, ahol az adat tárcíme.

A különbség a bázisregiszteres és az indexregiszteres címzés között az, hogy az indexregiszter tartalmát a program saját maga is megváltoztathatja. Az indexregiszteres címzésnél a relatív cím vagy index az adattömb elejétől számított távolság.



7.8. ábra: Az indexregiszteres címzés

### 7.2.8 UTASÍTÁSTÍPUSOK

Műveleti kód	R1	R2					RR
0	8	12 15					
Műveleti kód	R1	X1	B2	D2			RX
0	8	12	16	20	31		
Műveleti kód	R1	R3	B2	D2			RS
0	8	12	16	20	31		
Műveleti kód	I2		B1	D1			SI
0	8	12	16	20	31		
Műveleti kód	L		B1	D1	B2	D2	SS1
0	8	12	16	20	31	36 47	
Műveleti kód	L1	L2	B1	D1	B2	D2	SS2
0	8	12	16	20	31	36 47	

7.9. ábra: Utasítástípusok

- RR – mindkét operandus egy-egy regiszterben van. A gépen összesen 16 regiszter van, ezek 4 biten címezhetők. Az utasítás végrehajtása után az eredmény R1-be kerül.
- RX – az első operandus az R1 regiszterben van, a másik a tárban. A második operandus címzése bázisregiszteres, a címet indexregiszter módosítja. B tartalmazza a báziscímet, D az eltolást, X2 az indexet. Így a 2. operandus címe = (X2)+(B2)+D2, ahol (X2) az X2 tartalmát, (B2) a B2 tartalmát jelenti. Az X2 és a B2 általános célú regiszter, amelyből 16 van. Az eltolási mező 12 bites, azaz 4 kbyte-ot címezhet meg.
- RS – R1 és R3 általános célú regiszter, B2 és D2 a második operandus tárcímét állítja elő bázisregiszteres címmel. Ez három címes utasítás.
- SI – Az első operandus bázisregiszteres címmel megadott tárcím, a második pedig egy byte-os közvetlen adat.
- SS1 és SS2 – Mindkét operandus a tárban van, címüket bázisregiszteres címmel állítjuk elő. SS1-nél az érték az operandusok byte-ban kifejezett hossza mínusz 1. Az SS1 utasítást csak azonos hosszúságú operandusoknál használhatjuk. Az SS2 típusnál az operandusok eltérő hosszúságúak is lehetnek.

### 7.2.9 AZ IBM PC UTASÍTÁSSZERKEZETE

Az INTEL 8088 16 bites, 4 különböző típusú regiszterekkel rendelkezik.

Általános regiszterek:

- ☞ Akkumulátor (AX)
- ☞ Bázisregiszter (BX)
- ☞ Számolóregiszter (CX)
- ☞ Általános regiszter (DX)

Címzési regiszterek:

- ☞ Indexregiszter (SI)
- ☞ Indexregiszter (DI)
- ☞ Verem-bázisregiszter (BP)

Vezérlő regiszterek:

- ☞ Verem-mutató (SP)
- ☞ Utasításmutató (IP)
- ☞ Jelzőszó (flag)

Szegmensregiszterek:

- ☞ Kódszegmens-regiszter (CS)
- ☞ Adatszegmens-regiszter (DS)
- ☞ Második adatszegmens-regiszter (ES)
- ☞ Veremszegmens-regiszter (SS)

Az általános regiszterek különböző funkciók ellátására alkalmasak, általában adatok tárolására használjuk őket. A BX bázisregiszteres utasítások bázisregisztere lehet, a CX számolóként működhet.

Az SI és DI indexregiszterek használhatók indexregiszterként, illetve adattömbben mutatóként (indexként), a BP egy speciális rendeltetésű tárolórész, bázisregiszterként használható.

Az SP regiszter a verem elemeinek mutatója, IP az utasításszámláló. A flag 16 bitből áll, a bitek aritmetikai és egyéb műveletek végrehajtásánál használt jelzők.

Az IBM PC utasításformátuma:

Műveleti kód	d	w	mod	reg	r/m	
6	1	1	2	3	3	
1. byte			2. byte			közvetlen cím vagy adat (1 vagy 2 byte)

7.10. ábra: Az IBM PC utasításformátuma

Az utasításkészlet igen rugalmas, hiszen a második operandus címét a veremmutató, a bázisregiszter, valamint az indexregiszterek együttes felhasználásával lehet kialakítani. Azt, hogy a második operandus tartalmából hogyan kell kiszámítani a tárcímet, azt a „mod” két bitje és az r/m három bitje dönti el.

Az utasítás első operandusa csak regiszterben lehet. A regiszter címét adja meg a három „reg” bit. A „d” bit azt mutatja, hogy az operandust a regiszterből veszi-e ki az utasítás (d=0), vagy oda teszi be (d=1). A „w” bit az operandus hosszát jelöli, w=0 nyolcbites, w=1 pedig 16 bites operandust jelöl.

### 7.3 A SZÁMÍTÓGÉP MEGSZAKÍTÁSI RENDSZERE

A számítógép a program igényeinek megfelelően használja a különböző berendezéseket, a be/kiviteli eszközöket, a processzort, a tárolókat, a gép ún. *erőforrásait*. Ha a gépen egyszerre több program fut, az egyszerre egy erőforrást tud használni, a gép többi része kihasználatlanul áll. Ez a probléma csökkenthető, ha az egyszerre futó programok más és más erőforrásokat igényelnek ugyanabban az időpontban. A másik gond viszont az, hogy hogyan lehet megakadályozni, hogy a programok összeakadjanak, azaz egyszerre használják ugyanazt a berendezést. A megoldás: az éppen futó programokat megszakítják a processzoron, ha a processzort valamely okból egy fontosabb (erősebb) program igényli. Amikor a futó program visszakapja a vezérlést, ugyanonnan kell tudni indulnia, ahol befejezte működését. Tárolni kell tehát a program állapotának leírását. Ez a vezérlőegység egyik regiszterében történik meg.

Ez a regiszter az ún. *programállapotszó* (Program Status Word – PSW) regiszter.

Egy nagyobb számítógépen akár több száz program futását is koordinálnia kell a megszakítási rendszernek. A vezérlés visszamindig akkor történik, amikor az „erősebb” program processzorigénye megszűnt. A programok „erősségét” *prioritásnak* nevezzük.



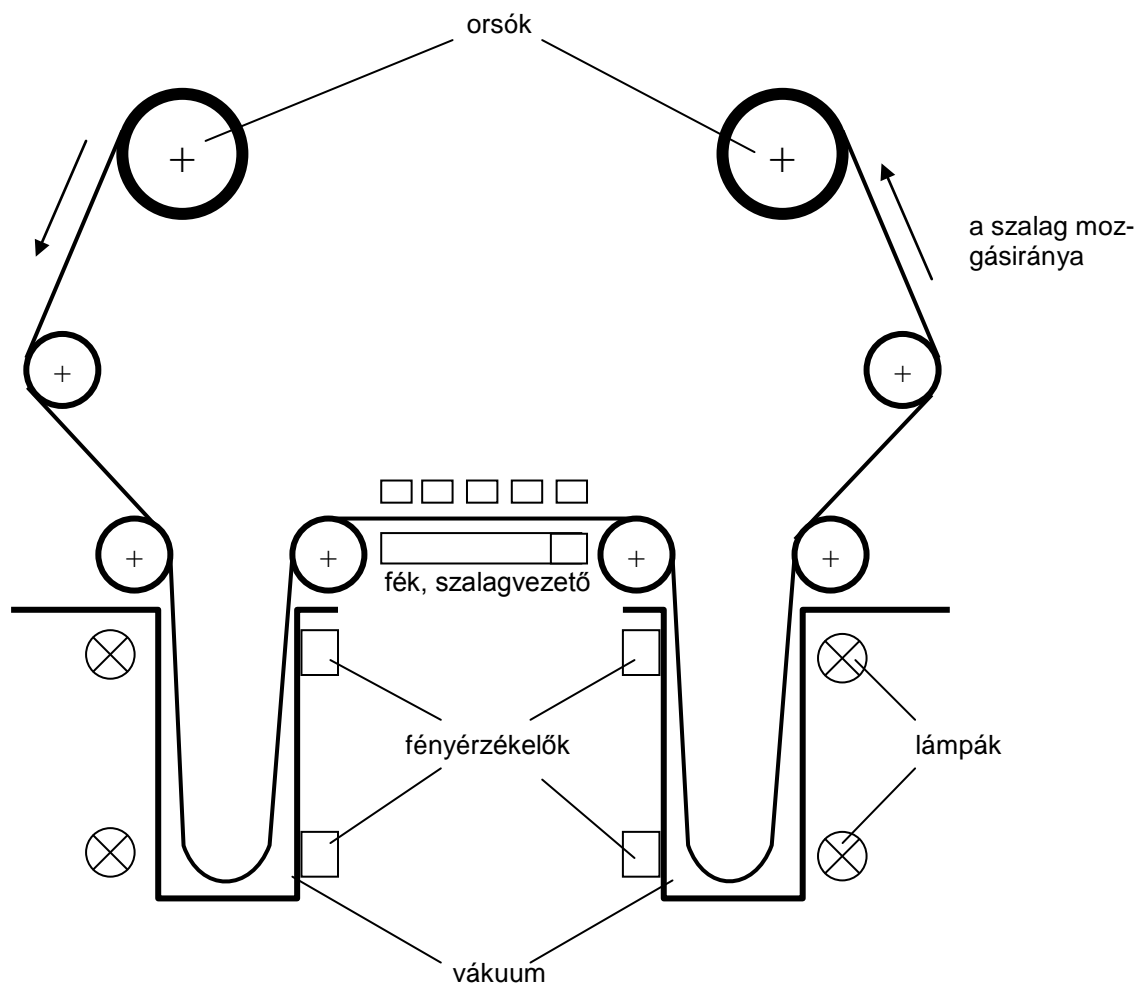
A megszakítást más célra is lehet használni, például a futó program a gép operációs rendszerének egy programját veszi igénybe, vagy áramkimaradás, esetleg hardver- vagy szoftverhiba miatt meg kell állnia, helyét a *hibakezelő program*nak kell átadnia.

## 8 HÁTTÉRTÁRAK

### 8.1 MÁGNESSZALAGOS ADATTÁROLÓ

A mágnesszalagos adattároló berendezés egy keménymágneses anyaggal bevont műanyag szalagra rögzíti az adatokat. A jelenleg használatos szalagok szélessége 0,5 inch, hossza 750 m, vagy 375 m. A szalagok tekercsben vagy zárt kazettában kaphatók.

A mágnesszalag – írás vagy olvasás közben – az egyik orsóról áttekeredik a másikra.



8.1. ábra: Vákuumkamrás mágnesszalagegység

A szalag elejétől néhány méterre egy szalagra ragasztott alumínium fóliacsík jelzi a hasznos szalagrész kezdetét, ugyanígy a szalag vége előtt egy másik fóliacsík a hasznos rész végét. A két fóliadarab közötti rész teljesen egységes.

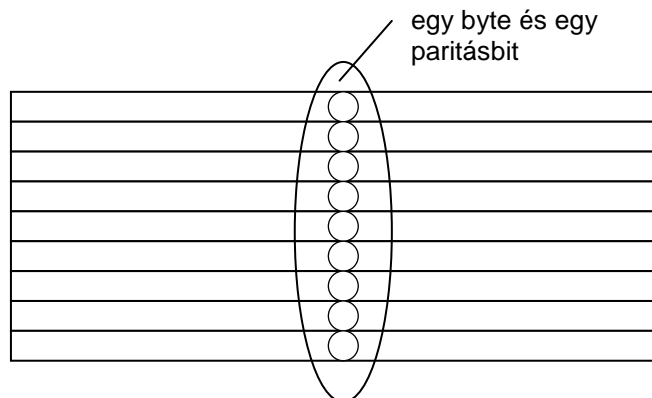
A szalag betöltésekor a szalagegység megkeresi a szalag elejét, ha megtalálta, az egység „kész” állapotba kerül, azaz a szalagra írni, vagy arról olvasni lehet.

#### 8.1.1 ADATTÁROLÁS A MÁGNESSZALAGON

Az adatok a szalagon kilenc hosszanti sávon, ún. *csatornán* kerülnek felírásra. A kilenc csatornán egymás mellett lévő kilenc bitből nyolc egy byte-ot alkot, a kilencedik az ellenőrzési célra használt paritásbit.

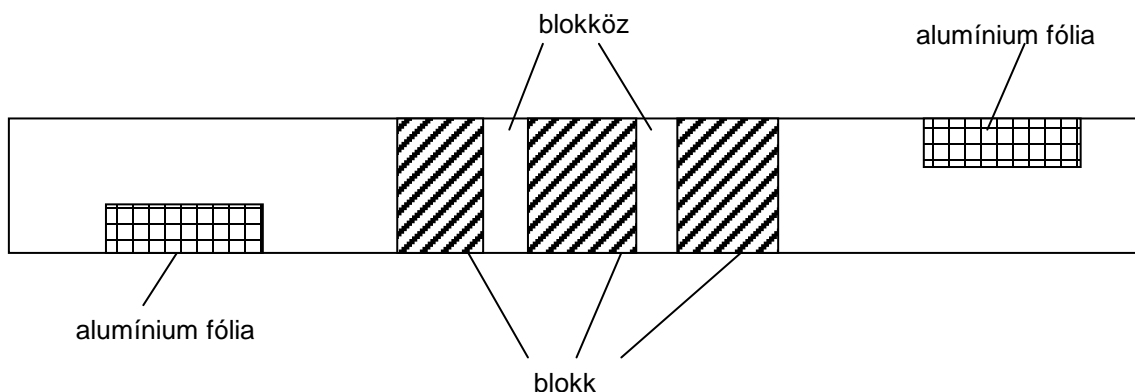
A mágnesszalag nem címezhető adathordozó, azaz magán a szalagon nincsenek olyan fizikai jelek, amelyeket a berendezés a felírt adattartalom olvasása nélkül megkereshetne. A sza-

lag mérete nem állandó, kisebb-nagyobb megnyúlás lehetséges, ezért a szalagon nem lehet fixen címeket rögzíteni. Az adatok megkeresésére úgy van mód, hogy azokat elolvassuk és bizonyos mértékig értelmezzük.



**8.2. ábra:** Kilencsatornás mágnesszalag egy darabja

A szalagon az adatokat ún. *adatblokkok*ban tároljuk, a blokkok között üres hézagot hagyunk, ezt *gap*nek nevezzük. Egy blokk több byte-ból áll, a blokk célszerű méretét a programozó határozza meg.



**8.3. ábra:** A mágnesszalag szerkezete

A blokkon belül a karaktereket a gép egymás után rögzíti. A mágnesszalagnál az írás és olvasás blokkonként történik, azaz a blokk az a legkisebb adatmennyiség, amelyet a szalagra a CPU-ból kivihetünk, vagy amelyet onnan a CPU-ba bevihetünk. A beolvasásra kijelölt blokk megtalálásakor a kereséskor gyorsan csévélődő szalagnak le kell lassulnia olvasási sebességre. Erre a célra szolgál a blokkok közötti hézag. A hézag nagysága nem állandó, mérete normál körülmények között kb. 15 mm.

### 8.1.2 A MÁGNESSZALAGEGYSÉG MŰKÖDÉSE

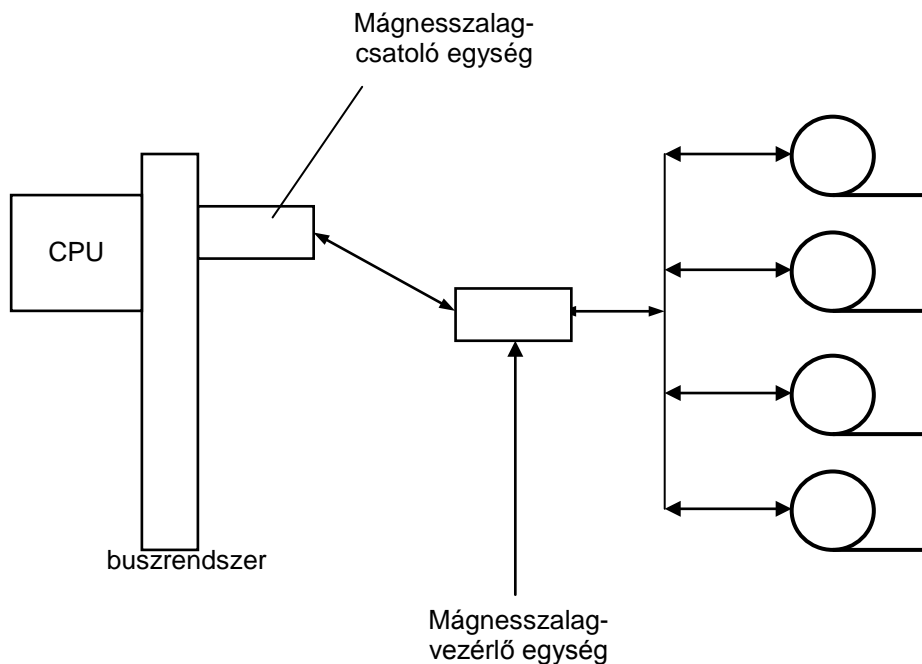
A szalag a gyakori gyorsítások, leállások miatt állandó rángatásnak van kitéve. Ezért a szalag két ún. kiegyenlítő pufferhurokkal különül el az orsóktól. A kiegyenlítést végezhetik rugós karok, de a jelenlegi szalagegységekben a vákuumkamrás megoldás a jellemző. Amikor az egység olvasási, írási vagy csévélési utasítást kap, az olvasó- és írófejek előtti kis szalagdarabka indul meg két, az orsóktól független mechanizmus hatására. A szalag a pufferből, azaz a vákuumkamrából jön elő, csak később kezd az orsóról letekeredni. Így a hirtelen rántás erőhatását sikerül időben széthúzni. A szalagtovábbító mechanizmus a szalagot az egyik vákuumkamrából a másikba továbbítja.



A szalag megfelelő utasítás hatására elhalad a törlő-, író- és olvasófej előtt, ahol a berendezés a kívánt műveletet elvégzi. Az adatok olvasási és továbbítási sebessége néhány százezer byte másodpercenként.

A nagy orsók a vákuumkamrából csévélik fel vagy oda engedik le a szalagot, a mozgásiránynak megfelelően. A vákuumkamrák bejáratánál és alján elhelyezett fotocellák érzékelik azt, hogy elegendő pufferszalag áll-e rendelkezésre, ha nem, a berendezés a hiányt pótolja, illetve, ha van, a felesleget felcsévéli.

A mágnesszalagegységek a mágnesszalagvezérlőn keresztül kapcsolódnak a gép I/O rendszeréhez. Egy vezérlőre több egység csatolható.



8.4. ábra: A mágnesszalagegységek csatlakoztatása a központi egységhez

### 8.1.3 MŰVELETVÉGZÉS A MÁGNESSZALAGON

A mágnesszalagegység a következő műveleteket végzi:

- ☞ blokk írása
- ☞ blokk törlése
- ☞ blokk olvasása
- ☞ szalagmozgatás

Az írás, törlés és olvasás műveletét az író-, olvasó- és törlőfej végzi, miközben a szalagtovábbítási mechanizmus a szalagot folyamatosan mozgatja. A szalag pozicionálása a blokkok szerint történik, tehát olvasni, írni, törölni blokkot lehet, a szalag előremozgatása is blokkokon át megy végbe.

A *blokk írása* a soron következő blokk-köz után kezdődik, azaz az előző blokk végétől kb. 1,5 cm-re. Az írást megelőzően a blokk teljes tartalmát elő kell állítani a főtárban, majd ezt egyszerre kell a sínrendszeren, csatolón és vezérlőegységen keresztül a megfelelő mágnesszalagegységre eljuttatni, ahol az rögzítődik. A rögzítés után azonnal megtörténik a visszaolvasás és az összehasonlításos ellenőrzés. A hibát már a vezérlőegység észleli, a szalagot a blokk elejére állítja, és megpróbálja újra felírni a blokkot.

A szalag megkülönböztetünk értékes adatokat és tájékozódást segítő, *szalagjel* (tape mark, TM) blokkokat. A szalagjelblokk speciális jelkombináció, ami elkülöníthető a tartalmi jelentéssel bíró adatoktól. Ezt a blokkot a vezérlő ismeri fel és e szerint állapít meg egyes sza-

lagpozíciókat. Szalagjeleket a programozó bárhova írhat a szalagra, majd a programból a szalagjelek megkeresésével állítja be a szalagot a megfelelő pozícióra. Szalagjeleket általában a logikailag összetartozó blokkok elválasztására használunk.

A mágnesszalagegység típusától függ, hogy adott hosszúságú szalagra mennyi adatot tud írni. Az egységnyi szalagra jutó bitek számát *írásűrűségnek* nevezzük. Mértékegysége a bit per inch (bpi). Értéke típustól függően 800-6400 bpi között változik.

A *blokkok törlése* nulla értékű jelek felvitelét jelenti a szalagra.

A *blokk olvasása* a blokk megkeresésével kezdődik. A programozó a szalagjelek segítségével rááll az olvasandó blokkra, majd a blokk tartalma egy program által megadott memóriaterületre kerül. Az olvasás alapegysége a blokk, egyszerre mindig egy blokk olvasása történik meg. Ha a vezérlőegység olvasási hibát fedez fel az ellenőrzések során, az olvasást megismétli. Az olvasás és átvitel sebessége a blokkhossztól függ.

A *szalag mozgatása* a szalagkezdet jel, a szalagjelek és az adatblokkok alapján történik. A mozgatás elemi műveletei az alábbiak:

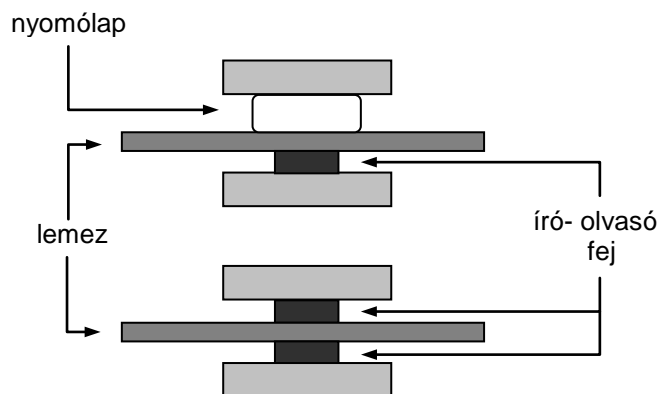
- ☞ szalagcsévévelés előre a következő TM utáni blokkig
- ☞ szalagcsévévelés hátra a megelőző TM előtti pozícióra
- ☞ szalagcsévévelés előre a következő blokk elejére
- ☞ szalagcsévévelés hátrafelé a megelőző blokk elejére
- ☞ szalagcsévévelés a szalag elejére (szalagkezdet jelig)
- ☞ szalag visszacsévévelése az eredeti orsóra

Ezekkel a műveletekkel lehet a szalagot a kívánt helyre pozícionálni.

A mágneskazettás adattárolásnál a tárolás elve megegyezik a mágnesszalagos adattárak működési elveivel, a különbség a méretekben, a meghajtómechanikában és a szalagformátumokban van. A kazettából a szalag nem vehető ki, azzal zárt egységet alkot.

## 8.2 MEREVLEMEZES TÁROLÓ

A merevlemez tároló nagy sebességű és nagy tárolókapacitású háttértároló. Fizikailag egy vagy több közös tengelyen elhelyezett lemezből áll, melynek bevonata mágneses mezőre érzékeny. Minden lemez mindkét oldala írható és olvasható, ez alól csak a két szélső lemez burkolat felőli oldala a kivétel. Minden oldalhoz tartozik egy író- és egy olvasófej. Ezeket általában egybeintegrálva készítik el. Az összes fej egyszerre mozdul el, a fejmeghajtó elektronika biztosítja, hogy az összes lemezoldalra egyszerre történjen meg az írás vagy olvasás.



8.5. ábra: A fej és a lemez fizikai kapcsolata

A lemez felülete speciális anyag, melynek elemi részecskéi a mágneses mezőtől függő irányba állnak be. A fej típusa nyitott lággyvas, melyen egy tekercset helyezünk el. Ha a tekercsre váltakozó feszültséget kapcsolunk, akkor a vasmag két vége közti nyílásban mágneses mező indukálódik. A mágneses erővonalak „kigyűrődnek” a vasmag síkjából. Ezek az erővo-

nalak érintkeznek tulajdonképpen a lemezzel. Az erővonalak iránya a lemezbevonat részecskéit a megfelelő irányba állítják. Mivel a számítógép digitális működésű és bináris kódolást használ, ezért a jeleknek csak két állapota lehet, 0 és 1.

### 8.2.1 FIZIKAI LEMEZKEZELÉS

Amikor írunk a lemezre, mindig csak egy fej dolgozik egy időben. Ha a lemez forgási sebessége állandó, akkor a lemez külső és belső széle nem egyforma kerületi sebességgel forog. Ezen lehet segíteni, hogy ha az elektronika a forgási sebességet úgy változtatja, hogy a fej alatti sáv sebessége állandó maradjon. A lemezeket forgató motor fordulatszáma azonban nem állandó, bizonyos tűrési tartományon belül mozog. Ennek az eredménye, hogy a lemez forgási sebessége is ingadozni fog. Ez a beolvasásnál is és az írásnál is komoly probléma, mert így a szektorhatárokat nehéz megtalálni. Ennek kiküszöbölésére az információt kódoljuk és a kódolásba beépítünk olyan információt, mely megoldja a szinkronizációt. A legismertebb kódolási eljárások:

- ☞ *FM* (Frequency Modulation, frekvencia moduláció): Ennél a kódolásnál az alacsony és a magas szinteket eltérő frekvenciájú jelek jelzik. Általában a 0 szinthez egy nagyságrenddel alacsonyabb frekvenciájú jel tartozik, mint az 1 szinthez.
- ☞ *MFM* (Modify Frequency Modulation, módosított frekvencia moduláció): Az FM kódolás hiányosságait hivatott kiküszöbölni az MFM eljárás. A jeleket itt felfutó élek jelölik. A két szintet a felfutó él elhelyezkedése határozza meg. 0 szintet a bithatár elején lévő, míg az 1 állapotot a bitközépen elhelyezkedő felfutó él jelöli. Ha az 1-est 0 követi, akkor ott nincs impulzus a kódolt jelben.
- ☞ *Manchester kódolás*: Ez a kódolás nullára szimmetrikus feszültségszinteket használ. Itt a biteket jelátmenet jelképezi, az ugrás irányának is szerepe van. Például 0→1 átmenet 1-est, az 1→0 átmenet 0-át jelöl. Akkor, ha több azonos bit követi egymást, akkor a jelnek a két bit között vissza kell térnie az eredeti szintre azért, hogy a következő bitidőben ugyanolyan átmenet következhesen.
- ☞ *RLL kódolás* (Run Length Limited, futási hossz korlátozás): Két 1 állapot között meghatározott számú 0 állapotnak kell lennie. Több változata létezik.

A lemez kezelésében különböző részeket különböztetünk meg. A lemezeket koncentrikus körökre osztjuk, ezeket nevezzük sávoknak. A sávok a fejek sugárirányú elmozdulásával érhetők el. Minden sávot megadott számú, egyenlő hosszúságú részre osztunk, ennek az elnevezése szektor. Az egymás fölött elhelyezkedő sávokat nevezzük cilindereknek. A szektor a kezelhető legkisebb adategység. A szektorokat felépítésük alapján két részre oszthatjuk, a szektorfejre és az adatblokkra. Ez utóbbi a lemezre írt adatokat tartalmazza. A szektorfej adminisztratív célokat szolgál. Tartalmazza a sáv számát, a fej sorszámát, a szektor számát és a hosszának kódját. Mindkét részt egy speciális jel előzi meg, hogy a különválasztás megbízható legyen. Az adatblokkot az adatjel (Data Mark, DM), míg a szektorfejet az azonosító jel (identifier Mark, IM) vezeti be. A DM értéke normál és törölt állapotot tartalmazhat. Ha írunk a lemezre, a vezérlő ezt a jelzést felülírja.

A merevlemez tárolóba beépítettek egy hibafelismerő és egy hibajavító áramkört. Ezek használatával felismerhetők az adatátvitel során bekövetkező hibák, valamint ki is javíthatók azok.

Ahhoz, hogy a lemezen létrejöjjenek a sávok és a szektorok, egy speciális műveletet kell végrehajtanunk, amit formázásnak hívunk. A formázásnak két része van: az alacsony szintű (Low Level Format) és a logikai formázás.

Az *alacsony szintű formázás*nál a merevlemez vezérlő a szektorokat megszámozza 1-től kezdve folytonosan egyesével növelve. Az ideális az lenne, ha a vezérlő a szektorokat folyamatosan növekvő sorszám szerint olvasná be. Az információ olvasásához és feldolgozásához azonban idő kell, ami alatt a lemez tovább fordul. Ha a következő szektor eleje már elhaladt a

fej előtt, az nem tudja kiolvasni a benne található adatot, ezért kell várni egy teljes körülfordulást. Ez jelentős idővesztést jelent. A megoldást a *logikai formázás* jelenti, melyet az operációs rendszer végez el.

A merevlemez fizikai kezelése tulajdonképpen sáv- és szektorszintű műveletek. A szektorszintű műveletek a következők:

- ☞ szektorok írása és olvasása
- ☞ törölt jelzésű szektorok írása és olvasása
- ☞ azonosító olvasása

Minden fizikai műveletet elő kell készíteni a fej beállításával. Ez azt jelenti, hogy a fejet a kívánt sáv fölé kell helyezni. A merevlemezeken az adatátvitel olyan gyors, hogy megfelelő sebességgel csak közvetlen memóriáhozáféréssel lehet kezelni.

A sávszintű műveleteket adatátvitelre nem használják, elsősorban a lemez formázására és vizsgálatára alkalmasak. A sávszintű művelet végrehajtása az indexjelnél kezdődik és a sáv végéig tart. A végrehajtáshoz meg kell adni a sáv, a fej és a szektor számát. A művelet után a vezérlő sorban elkezd olvasni a lemez szektorfejeit és megvizsgálja a sorszámot. Ha egy körbefordulás alatt nem talál olyan számút, mint amit mi előírtunk, akkor a Sector Not Found hibüzenetet kapjuk. A szektorokat 1-től, míg a fejeket és a sávokat 0-tól kezdik el számozni. Lehetőségünk van azonban ettől eltérő formátum létrehozására is.

### 8.2.2 A MEREVLEMEZ LOGIKAI KEZELÉSE

A logikai lemezkezelés alapja a több szektorból álló szektorcsoport, idegen szóval cluster. Ebben a blokkcsoportban tartja nyilván az operációs rendszer a lemezterületek foglaltságát, a file-ok elhelyezkedését. Annak érdekében, hogy kezelni tudjuk a clustereket, azok számozására van szükség. Ez a számozás azonban független a lemezegység fizikai paramétereitől. Az átvitel gyorsaságának növeléséhez a szektorokat nem folyamatosan számozzuk, hanem két egymást követő sáv között annyi távolságot hagyunk, hogy ennyi idő alatt a vezérlő fel tudja dolgozni az adatokat. Így az olvasás és az írás folyamatosan történik. A fizikai és a logikai sorszámok közötti kapcsolatot az átlapolási tényező (Interleave Faktor) adja meg. Az átlapolási tényezőt az alacsony szintű formázásnál lehet beállítani.

Minél nagyobb a lemez mérete, annál nagyobb a cluster mérete is. A merevlemez vezérlő elektronikája írni és olvasni a merevlemez csak clusterenként tudja. Ez azt jelenti, hogy a legkisebb írható vagy olvasható méret a clustermérettel egyenlő.

A helyvesztés csökkentése érdekében annál jobb, minél kisebb a merevlemez mérete. Hogy ez ne okozzon problémát, a lemezt logikai részekre, partíciókra kell felosztani. Minden merevlemez egy fizikai partícióból áll, ennek mérete a lemez teljes területével egyenlő. A fizikai partíciót osztjuk fel logikai partíciókra. Kétféle logikai partíciót különböztetünk meg, az elsődleges és a másodlagos logikai partíciót. A logikai partíciókat úgy látjuk, mintha külön merevlemez volna, külön-külön tudjuk írni és olvasni őket. A partíciók felhasználásával több operációs rendszer futtatására is lehetőségünk nyílik. A partíciók információit a merevlemez el kell tárolni, hogy azokat az operációs rendszer képes legyen felismerni és kezelni. Erre a célra szolgál a merevlemez partíciós táblája.

### 8.2.3 PARTÍCIÓS TÁBLA

A partíciós tábla mindig a 0. számú logikai szektor, a 0. sávon a 0. fejjel elérhető 1. fizikai sorszámú szektor. A partíciós tábla a szektor végén található.

A szektor első részében egy programot találunk, ami az igazi partíciós táblát értelmezi és a betöltésre kijelölt partíció betöltő szektorát beolvassa. A program neve betöltő rekord (Master Boot Record, MBR). Ezt a szektort ellenőrzi, és ha hibátlan, el is indítja. A szektornak körülbelül a fele üres terület. Ide kerülnek azok a programok, melyek az MBR-rel kapcsolatosan végeznek műveleteket.

A ROM-BIOS program az önteszt és a kezdeti paraméterek beállítása után az első fizikai szektort (MBR és partíciós tábla) betölti, majd átadja a vezérlést. Ezután elindul az MBR-ben található program futása.

#### **8.2.4 ÁLLOMÁNYELHELYEZÉSI TÁBLA**

Az állományelhelyezési tábla (FAT, File Allocation Table) az állományok rekordjainak lemezen történő elhelyezkedését tárolja. A lemez legkisebb egysége a szektor, azonban az operációs rendszer által kezelt legkisebb egység a cluster. Az ebben található szektorok száma mindig 2-nek valamelyik hatványa, valamint összefüggésben van a partíció méretével. A FAT ezen clustereknek a foglaltságát tárolja, vagyis innen tudjuk meg, hogy hol van szabad hely a lemezen.

Az állományelhelyezési tábla egy táblázat, melynek bejegyzései vannak. Minden bejegyzésnek van száma, amivel azonosítjuk és természetesen van tartalma is. A bejegyzések száma függ a bejegyzések méretétől.

A bejegyzések száma tulajdonképpen egy clusterszám, vagyis a bejegyzések száma és a hozzá tartozó lemezterület között közvetlen a kapcsolat.

Az állománykezelési táblát két példányban tároljuk egymás mögött.

#### **8.2.5 KATALÓGUS**

Az állományokat a lemezen katalógusokban tároljuk. A katalógusrendszert mi hozzuk létre, és fa struktúrájú a felépítése. Ez azt jelenti, hogy a file-ok katalógusokban vannak, melyek még egymásba is ágyazhatók. Egy állomány vagy katalógus helyét megkapjuk, ha azokat a könyvtárakat, amelyeken át kell haladni, hogy elérjünk oda, ahová szeretnénk, leírjuk egymás mögé \ jellel elválasztva.

A főkatalógus a két FAT után következik a lemezen, és még a FAT szerkezetnek a része.

### **8.3 HAJLÉKONYLEMEZ MEGHAJTÓ**

A magnetofonok használata után a PC-kben megjelentek az 5,25” átmérőjű mágneslemezzel dolgozó floppy disk meghajtók. A lemezek kapacitása kezdetben 160 kbyte volt, ami a következőképpen adódott: egy oldalon 40 sáv, sávonként 8 szektor és szektoronként 512 byte.

A következő generáció már kétoldalas, így annak kapacitása is kétszeres, 320 kB. Újabb változást a sávonkénti 8 helyett 9 szektor felírása jelentett. Ezzel a lemezkapacitás 360 kB lett.

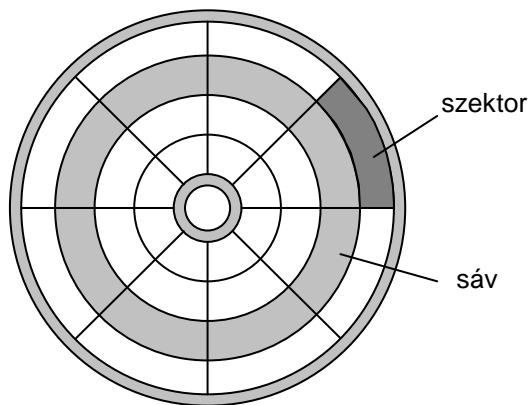
Az AT-hez egy új floppyegységet fejlesztettek ki. Erre a lemezre már nem 40 hanem 80 sávot lehet felírni, így az elérhető kapacitás 720 kbyte. Megfelelő floppy lemezeknél ez az egység képes sávonként 15 szektort felírni a lemezre, ekkor a kapacitás 1,2 MB.

Ezek mellett megjelent egy új floppyegység is. Ez 3,5” lemezzel dolgozik. Ezek a lemezek rendelkeznek a legnagyobb kapacitással. Sávonként 18 szektort írnak a lemezre, az így elérhető kapacitás 720 kB, illetve 1,44 MB attól függően, hogy DD (Double Density) vagy HD (High Density) technikát alkalmaznak.



**8.6. ábra:** A 3,5" lemez felépítése

A 3,5" lemezt alul és felül egy-egy kemény műanyag borítólemez védi. A borítólemezek és a mágneses bevonatú lemez között mindkét oldalon egy-egy textil korong helyezkedik el. Ennek rendeltetése az, hogy a használat közben a lemez felületére kerülő port megkösse, így a porban lévő apró, kemény szemcsék se a lemezt, se a mágneses író-olvasó fejet ne karcolják össze. A lemezen található zárólemeznek is ez a feladata. Használat közben a floppy meghajtó ezt a kis lemezt félrehúzza és így az író-olvasófejek az alsó – felső borítólemezen lévő részen át érik el a mágneses lemezt. Használaton kívül a zárólemezt egy kis rugó visszatolja alaphelyzetébe. Műveletvégzéskor a floppyegység a mágneses lemezt a tengelyen keresztül forgatja meg.



**8.7. ábra:** A lemez logikai felépítése

A hajlékonylemezes csatolóártya négy meghajtó csatlakoztatását teszi lehetővé. A meghajtó sorszámát a vezeték határozza meg.

A csatoló általában nem önálló kártyaként kerül beszerelésre, hanem összeintegrálják egyéb perifériákkal, mint például a merevlemez vezérlővel, a soros vonali illesztővel, stb.

## 8.4 A CD-ROM

A CD-ROM (Compact Disc – Read Only Memory) manapság nagyon elterjedt adattárolási eszköz. Ez főként alacsony gyártási költségének és nagy tárolókapacitásának köszönhető.

### 8.4.1 A LÉZERTECHNOLÓGIA

A LASER rövidítés a Light Amplification by Stimulated Emission of Radiation kifejezésből származik.

A fehér fény a szivárvány minden színét tartalmazza. Minden szín különböző frekvenciájú fény. Az alacsonyabb frekvenciájú jelek a sötétvörösbe, míg a magasabb frekvenciájúak a viola színbe mennek át. A közönséges fénysugarak szétszóródnak a tér minden irányába, szakszóval nem koherensek. Ez azt eredményezi, hogy nem tudjuk semmilyen eszközzel egy megadott pontba fókuszálni. A lézerefény alapján különböző a látható fehér fénytől. Ennek a fénynek egyetlen színe van, ami koherens. Így lehet fókuszálni és erősíteni is, tehát nagyon jól irányítható.

A lézersugár többféle gázzal és anyaggal előállítható. A CD-ROM-ok többsége olyan fényt használ, ami a színspektrum kisebb frekvenciájához tartozik. Ilyen például a piros és a sárga.

#### **8.4.2 A LEMEZ FELÉPÍTÉSE**

A szabványos CD-ROM lemez 4,75" (120 mm) átmérőjű és 1,2 mm vastag. Egy átlátszó polikarbonát műanyag hordozóra alumínium (vagy arany) réteget, majd egy lakkréteget visznek fel. A polikarbonát hordozót öntési eljárással készítik, melynek során az egyik oldalán apró bemélyedéseket hoznak létre. Erre az oldalra viszik fel a jól tükröződő alumíniumréteget, majd a lakkréteget, ami véd az oxidációtól és a mechanikus sérülésektől.

A tükröző réteg felületét land-nek, a bemélyedéseket pedig pit-nek hívják. A pitek spirál alakban sorakoznak a lemezen belülről kifelé.

A lemez olvasásakor a CD lejátszó vagy meghajtó lézersugarakat bocsát a polikarbonát hordozón keresztül az alumínium felületre. A sugár a pitekről és land-ekről különböző módon verődik vissza, ami gyakorlatilag megfelel az „egyeseeknek” és „nulláknak”.

A lézernyaláb a lemezről visszaverődik, melyet érzékelni kell. Problémát jelent, hogy a lemezek nem teljesen kör alakúak, tehát a sávok nem mindig ugyanazon a sugáron találhatók. A másik probléma, hogy a lemezek síkja nem teljesen tökéletes. Ez azt eredményezi, hogy a fej hol távolabb, hol közelebb lesz a lemezhez. Ezeket a hibákat fel kell ismerni és még az olvasás előtt megfelelően korrigálni kell. Ez az érzékelő speciális kialakításával lehetséges. Nem egy, hanem négy érzékelőt integrálnak egy közös tokba. ezek elhelyezkedése négyzetes. Ha a lézernyaláb megfelelően verődik vissza, akkor ez mind a négy érzékelőre pontosan azonos mértékben vetődik. Ha valamely érzékelő nagyobb fénymennyiséget érzékel, akkor az elektronika abból meg tudja állapítani, hogy mi a hiba oka és el tudja végezni a korrekciót. A fejeket apró motorok mozgatják, a pozicionálásuk nagyon gyors és pontos a beállítás.

Igazából a pit nem 1-est és a land nem 0-t jelent. Valójában a pit és a land is 0-t jelent, míg a pit és a land közötti átmenet jelent 1-est. Amikor a letapogató lézersugár pit és land közötti átmenetre ér, a visszaverődött sugár állapotában bekövetkezett változás jelenti az 1-est. A különböző hosszúságú pit-ek és land-ek különböző számú 0-át jelentenek. Műszaki okok miatt a pit-ek és a land-ek nem hordozhatnak 3 bitnél rövidebb és 11 bitnél hosszabb információt, beleértve a pit és land közötti átmenetet is. Ez egyszerűen azt jelenti, hogy a bináris számábrázolást a CD-technika nem teszi lehetővé. Emiatt egy 14 bites kódolást alkalmaznak, és a kiolvasáskor a 14 bites szavakat visszakódolják a megszokott 8 bitesekké.

A mágneslemezek állandó szögsebességgel forognak. Azaz a lemez fordulatszama állandó, aminek következtében a belső sávok sebessége kisebb, mint a külsőké. Következésképp a belső sávokban nagyobb az adatsűrűség, mint a külső sávokban.

A CD-knél ilyen gond nincs, mert minden szektor azonos hosszúságú. Ha a forgás állandó lineáris sebességgel történne, az azt jelentené, hogy amikor a lézersugár a belső sávokat olvassa le, a szögsebesség nagyobb, mint amikor a külsőket. A CD-ROM olyan technikát használ, amely képes változtatni a forgási sebességet attól függően, hogy éppen a lemez mely részét olvassa. Így biztosítani lehet, hogy a lemez és a lézerfej egymáshoz viszonyított sebessége állandó maradjon. Mivel az adatsűrűség állandó, a tárolható információ mennyisége sokkal nagyobb, mint a mágneslemezeknél. Ugyanakkor a hozzáférési idő valamelyest nő, mivel a lemez szögsebességét a lézersugár helyzetének függvényében állandóan változtatni kell.

Míg a tömegben gyártott CD-ken a pit-eket a gyártás során hozzák létre, a CD-R lemezek üresek. A CD-R lemezek polikarbonát hordozója hasonló a CD-ékéhez, de a gyártás során felvisznek a lemezre egy spirális sávot, ami az íráskor a lézersugár pozicionálását segíti. Az alumíniumréteg helyett egy szerves festékréteget használnak, majd erre egy vékony aranyréteg kerül. Az aranyréteg jó visszaverő képességű és nem korrodál a festékréteggel érintkezve. Ezután védőlakkréteget visznek fel.

A CD-ROM-ok fejlődésével egyre gyorsabb és gyorsabb meghajtókat hoztak forgalomba. Alapsebességnek az audio CD sebességét tekintjük, a CD-ROM-ok sebességét az ehhez viszonyított szorzószámmal fejezzük ki. Az audio CD sebessége 150 kbyte/s, ez az 1x sebesség.

A 8x-nál nagyobb sebességű CD-ROM-ok egy speciális tulajdonsággal rendelkeznek. Ahhoz, hogy ezt a sebességet nyújtani tudják, a fordulatszámot nagyon meg kellett növelni. Sok energiapazarlással járna, ha a lemez mindig forogna. Azért, hogy ez ne legyen így, ha nincs lemezművelet, a forgást leállítják a meghajtók. A másik sajátosság, hogy a meghajtóra ráírt sebesség a maximálisan elérhető adatátvitelt jelzi. Ez csupán az elméleti maximum.

### 8.4.3 CD ÍRÁSA

A CD író a lézersugarat a festékrétegre fókuszálja és az írás során beégeti azt. Ilyen módon a pit-hez hasonló elváltozás lép fel a rétegen. Ennek megfelelően gyakorlatilag az összes szabványos CD-meghajtó képes az ilyen módon elkészített lemezeket olvasni. A pit-ek sűrűn sorakoznak a sávokon, amelyek 1,6 mikron távolságra vannak egymástól.

### 8.4.4 CD LEMEZEK NAGYÜZEMI GYÁRTÁSA

Első lépés, hogy a lemezre rögzítendő adatokat összeállítják és rendszerezik. Ezeket felmásolják egy megfelelő méretű és nagyon gyors merevlemezre. Az adatokhoz tartoznia kell egy tartalomjegyzék listának, egy index, egy hibafelismerő és javító, valamint egy információ-visszakereső szoftvernek.

Az adatokról elkészítik az első ún. one-off lemezt. Ez nem más, mint egy tesztváltozat. Ha ez minden kívánalomnak megfelel, akkor erről készítenek egy üvegből készült mesterlemezt. Ebbe lézerrel vágják bele az adatoknak megfelelő pit-eket. Ezek után minden további változat erről a lemezről készül.

A CD lemezeket préselik, ezért amit a mesterlemezzel készítenek, az az eredetinek pontos tükörképe lesz. Ahhoz, hogy az eredetinek megfelelő lemezt kapjunk, ismét egy másolatot kell készíteni. A mesterlemez első másolatát nevezik anya-lemeznek.

Ezek után az anyalemezről elkészítik az első működőképes változatot, melyet apa-lemeznek neveznek.

Az üres lemezek átmérője 120 mm. Préselésük után egy visszatükröződő alumínium réteggel vonják be. Erre egy védőlak réteg kerül, hogy ne oxidálódjon az alumínium réteg.

### 8.4.5 CD-ROM-OK VEZÉRLŐI

Ahhoz, hogy a CD meghajtót a számítógép kezelni tudja, valamilyen kapcsolatot kell létesítenie a processzorral. Alapvetően 4 módon csatlakoztatható az egység a géphez:

1. Saját vezérlőkártyával. A vezérlőkártya általában nagyon egyszerű felépítésű. A címdekóderen és az illesztő-leválasztó fokozaton kívül nem is tartalmaz mást. A csatlakoztatást egy 40 pólusú csatlakozó teszi lehetővé. Ez az eljárás csak régi 1x és 2x CD meghajtók esetén volt használatos.
2. EIDE interfészre csatlakozik. erre négy egység köthető rá, melyek lehetnek CD-ROM-ok is. A CD-ROM tartalmazza a meghajtó elektronikát, a kábel csak a rendszerjelek meghosszabbítását végzi el.
3. Hangkártya megfelelő interfésze végzi az illesztést. A hangkártyák is tartalmaznak olyan meghajtó fokozatokat, melyek a CD-ROM meghajtását teszik lehetővé.



4. SCSI interfészre kapcsolható egység. Léteznek ilyen meghajtók, de az árak miatt nem terjedtek el. A CD írók esetében viszont szinte kizárólagosnak mondható ez a csatlakoztatási mód.

#### 8.4.6 CD TÍPUSOK

- ☞ (I)CVD – Integrated Compact Video Disc: a CD-V versenytársa analóg videó és digitális audio vagy adat tárolására képes.
- ☞ CD-3: 20 perc jó minőségű sztereó zene tárolására alkalmas, 3,5” átmérőjű.
- ☞ CD-DA (CD-A) – CD-Digital Audio: 60-80 perc jó minőségű sztereó zene tárolására.
- ☞ CD Electronic PROM: Elektronikus úton programozható CD.
- ☞ CD-I – CD-Interactive: nagy teljesítményű audio, videó tároló.
- ☞ CD-PAC – CD-Public Access Catalogues of library: könyvtári katalógusokra.
- ☞ CD-R – CD-Recordable: Egyszer írható CD-ROM.
- ☞ CD-ROM XA – CD-ROM eXtended Architecture: átmenet a CD-ROM és a CD-I között.
- ☞ CD-V – CD-Video: 5-6 perc videókép+hang és maximum 20 perc digitalizált zene tárolására képes.
- ☞ CD-V/EP: 8”/20 cm 20perc videókép+hang tárolására képes oldalanként.
- ☞ CD-V/LP: 12”/30 cm 1 óra videókép+hang tárolására képes oldalanként.
- ☞ CD-Write Once/CD-Programmable ROM: írható lemez szintén.
- ☞ DVD – Digital Versatile Disc: akár 17 GB információ tárolását is lehetővé teszi.
- ☞ DVI – Digital Video Interactive: mindenféle vizuális anyag tárolására és lejátszására képes eszköz.
- ☞ Hybrid videodisc: analóg és digitális adatok együttes tárolására. képes lemez.
- ☞ IV – Interactive Video: videó lemezen alapuló rendszer.
- ☞ LV – Laser Vision: reflektív optikai videolemez.
- ☞ LV-ROM – Laser Vision ROM: 54000 kép/oldal vagy 324 Mbyte digitális adat tárolására alkalmas.
- ☞ WORM – Write Once Read Many: egyszer írható, többször olvasható lemez.

## 9 BE/KIVITELI EGYSÉGEK

A számítógép be/kiviteli művelete mindig egy beviteli/kiviteli utasítás hatására indul meg. Az egyszerű I/O perifériáknál a bemenő adat – ami egyszerre egy byte – az I/O regiszterbe kerül, kivételnél pedig innen indul a megfelelő perifériára. A bevittelt vagy kivittelt kezdeményező gépi utasítás meghatározza a periféria jelét vagy számát (címét) és az elvégzendő funkciót.

Tipikus funkciók:

READ – a perifériaregiszter tartalmának beolvasása az akkumulátorregiszterbe

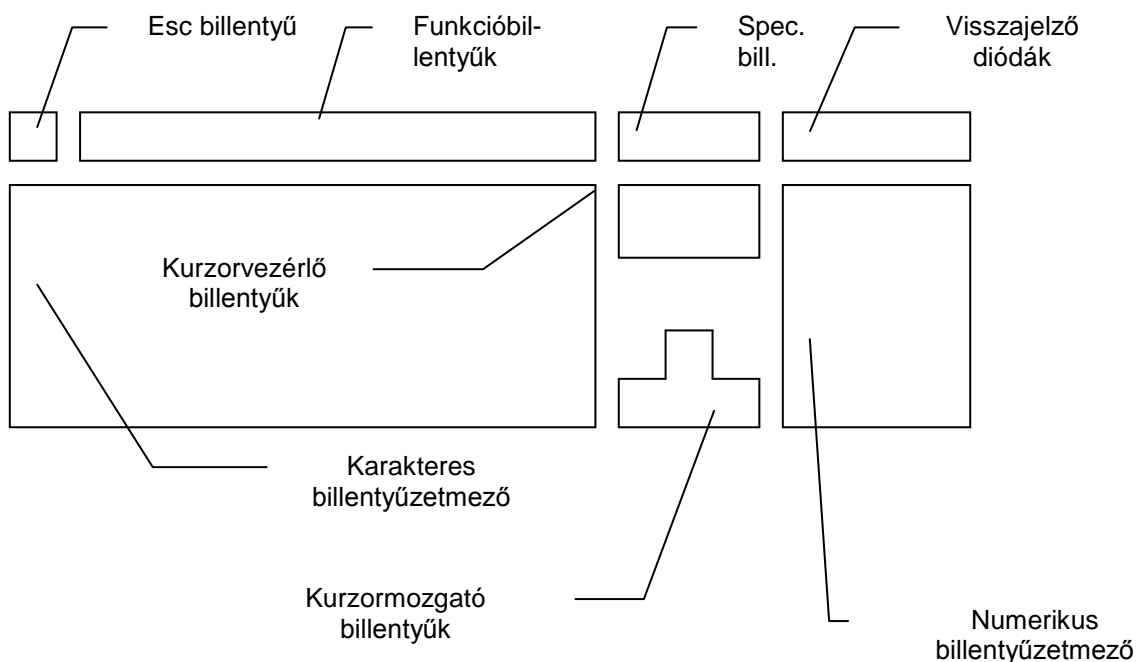
WRITE – az akkumulátor tartalmának kivitele a perifériaregiszterbe

SKIP – a következő utasítás átugrása, ha a periféria foglalt

CLEAR – a perifériaregiszter és a periféria foglaltságát jelző bit törlése

A gép minden utasítás végrehajtása előtt ellenőrzi az összes periférijelzőt, hogy nincs-e változás. Ha van, megszakításkérést bocsát ki. A kérés leállítja az éppen futó programot, tárolja a folytatáshoz szükséges adatokat, majd rátér annak a perifériának a kiszolgálására, amelyik éppen felszabadult.

### 9.1 A BILLENTYŰZET



9.1. ábra: A billentyűzet felépítése

A billentyűzet egy billentyűjének lenyomása egy héttites kódot állít elő. Ez rendszerint ASCII kód. A billentyűzetben a számok, betűk és írásjelek mellett általában találhatunk ún. *funkcionális billentyűket*, amelyek lenyomása egy előre beprogramozott funkció végrehajtását váltja ki. A funkcionális billentyűkhöz kapcsolódó programot a felhasználó adhatja meg.

A billentyűzeten lévő *SHIFT*, *CTRL* és *ALT* billentyűk lenyomása módosítja a többi billentyű által generált kódot. A *CTRL* általában arra szolgál, hogy visszavegyük a vezérlést a programtól annak befejezése előtt, a *SHIFT* a billentyűkön lévő „alsó” és „felső” jel közötti választást teszi lehetővé, az *ALT*-ot általában további módosításokra használjuk. Ezen kívül a

billentyűzetben még találunk a program vezérlésére használt *ESC* billentyűt, a képernyőn való szerkesztésre használt nyílakat, *DEL* és *INS* billentyűt valamint *kapcsolókat*.

Több fizikai megoldást dolgoztak ki és használnak arra, hogy a billentyű benyomása elektromos jelet váltson ki. Igen megbízható a mágneses kapcsoló, aminél a billentyű lenyomásával egy mágneset közelítünk két fémlapocskához, amelyek ennek hatására összeérnek és zárnak egy áramkört.

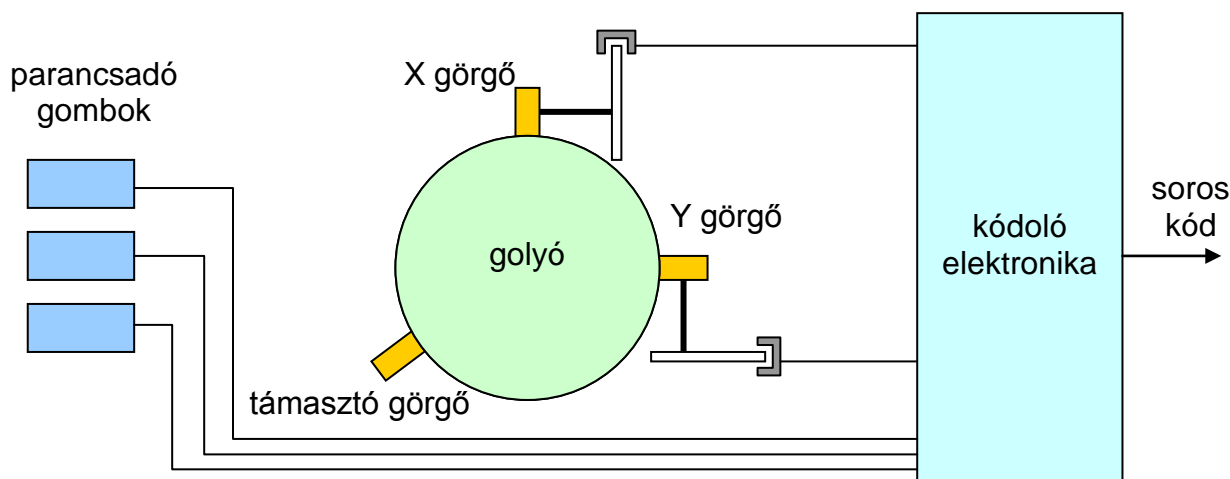
Más megoldások mágneses effektuson, kapacitásváltozáson és elasztikus ellenállások mérésén alapulnak. A billentyű lenyomásával kapott hét bithez a billentyűzet csatol egy ellenőrző bitet, majd a biteket sorosan vagy párhuzamosan a központi egységbe küldi. A nyolcbites kódot egy ún. *startbit* előzi meg, és egy *stopbit* követ, erről tudja a hardver, hogy mikor fogadjon egy kódot, és az meddig tart.

A billentyűzet áramkörei megakadályozzák azt, hogy ha például két billentyűt egyszerre nyomunk le, felismerhetetlen kód képződjék.

## 9.2 POZÍCIONÁLÓ ESZKÖZÖK

A pozícionáló eszközök kényelmes, gyors, és viszonylag pontos mozgást tesznek lehetővé a képernyőn nagy felbontású grafikus módban is.

### 9.2.1 EGÉR



9.2. ábra: Egér felépítése

Az egyik legelterjedtebb pozícionáló eszköz az egér. Az asztal felületén való elmozdulását a házban lévő golyó forgása követi, ezt látjuk a képernyőn is az ún. *egérkurzor* segítségével. A golyó két, egymásra merőleges hengerrel érintkezik, és a mozgás során forgatja őket, az elfordulása megfelel a golyó vízszintes és függőleges irányú mozgásának. A két henger forgását külön-külön érzékelők detektálják, és a megfelelő kódokat továbbítják az alaplapra.

Az egéren van két-három nyomógomb, melyek lenyomása általában kiválasztási funkciót indít el.

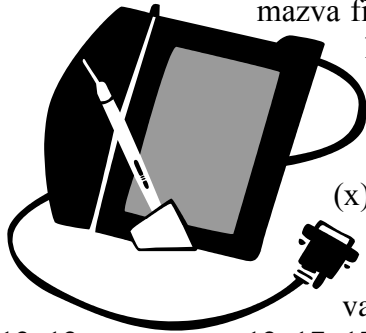
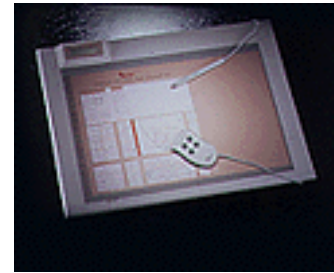
A kiváltott információk továbbítása általában soros formában történik egy kábelon keresztül a számítógép soros portja segítségével. Létezik infravörös fény segítségével működő változata is, ebben az esetben az egér tápellátását akkumulátor biztosítja.

### 9.2.2 TRACK BALL (HANYATTEGÉR)

Működése megegyezik az egérével. A pozícionáló golyó lehet a házhoz rögzített különálló egységben, vagy gépházba beépítve. A képernyőn az egérkurzort a golyó kézzel való forgatásával pozícionálhatjuk, a golyó mellett két vagy három nyomógombot is találunk.

### 9.2.3 DIGITALIZÁLÓ TÁBLA

Ha nagyobb pontosságot szeretnénk elérni, akkor van szükségünk a *digitalizáló táblára*, vagy *tabletre*. A tablet a számítógéphez csatlakozó érzékeny felület, amelyen egy tollszerű eszközzel vagy célkereszttel ellátott speciális egerrel lehet mozogni. A felület érzékelése különböző megoldású lehet, pl. a tollban egy apró mágnes van, és ezt érzékeli a tábla, vagy nyomásérzékelőt alkalmazva figyelik a toll helyzetét. Műszaki rajzhoz, vagy grafikához használjuk. A



tábla abszolút pozícionálást, míg az eger relatív pozícionálást tesz lehetővé.

Az abszolút pozícionálás azt jelenti, hogy a tábla vízszintes (x) és függőleges (y) méretei hozzárendelhetők a képernyőn megjelenő, illetve a készíteni kívánt alkalmazás valódi méreteihez. A digitalizáló táblák legfontosabb paramétere az x és y méret, valamint az érzékenység. Tipikus méretek inchben megadva: 5x5,

12x12, 12x17, 17x17 stb. Érzékenységen azt a legkisebb elmozdulást értjük, amit a tábla érzéklni képes. A korszerű táblák rendelkeznek nyomásérzékeny funkcióval is, ami azt jelenti, hogy a tábla nem csak a toll pozícióját érzékeli, hanem a nyomás erősségét is, általában 256 fokozatban. A tabletnek is van vezeték nélküli változata.

### 9.2.4 BOTKORMÁNY

A botkormányban az elmozdulás irányának érzékelésére két, egymásra merőleges potenciométer szolgál. A *joystick* nyomásérzékeny, azaz nem csak az elmozdulást, hanem annak sebességét is érzékeli. A botkormány ezen kívül egy vagy több kapcsolót is tartalmaz.

## 9.3 FÉNYCERUZA

A fényceruza (light pen) optoelektronikai eszköz. Ha egy ponton hozzáérintjük a képernyőhöz, akkor segítségével megállapíthatjuk, hogy megvilágítja-e az elektronsugár az adott pontot, vagy sem. Segítségével bemenő jelet állíthatunk elő. Kiválasztásra, kézi rajzok megjelenítésére használhatjuk.

## 9.4 SCANNER

A scanner képbeviteli eszköz. A szkennerek csoportosítása történhet a dokumentum kezelése, a dokumentum típusa szerint, valamint a felbontás nagysága szerint.

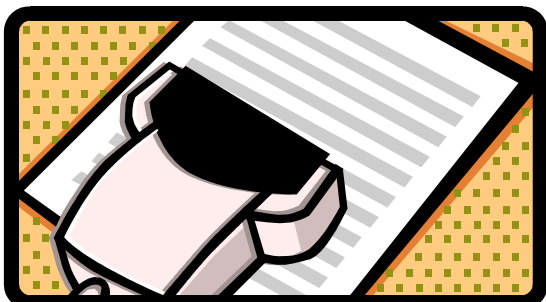
A scannerek fontos jellemzője a felbontás, mely megmutatja, hogy egy adott távolságról hány elemi pontot (pixel) tud behozni. Az optikai felbontás a teljes optikai úton érvényes felbontást jelenti, ahol minden egyes pixelhez egy CCD elem tartozik. A logikai felbontás számítási művelet eredménye, ilyenkor az alkalmazott szoftver a CCD két egymás melletti cellájának értékéből következtet a kettő közötti értékre.

A scannerek fő egységei:

- ☞ érzékelő
- ☞ optika
- ☞ megvilágító egység
- ☞ mozgató mechanika
- ☞ elektronika

☞ interface

Az érzékelő a dokumentumról érkező fényt elektronikus jellé alakítja, melyből az eredeti



másolata előállítható. Általában az érzékelő CCD (Charge Coupled Device) elem, ez egy töltéscsatolt félvezető eszköz, ahol egy sorban fényre érzékeny cellák helyezkednek el, és ezek a cellák a megvilágítással arányos feszültséget szolgáltatnak.

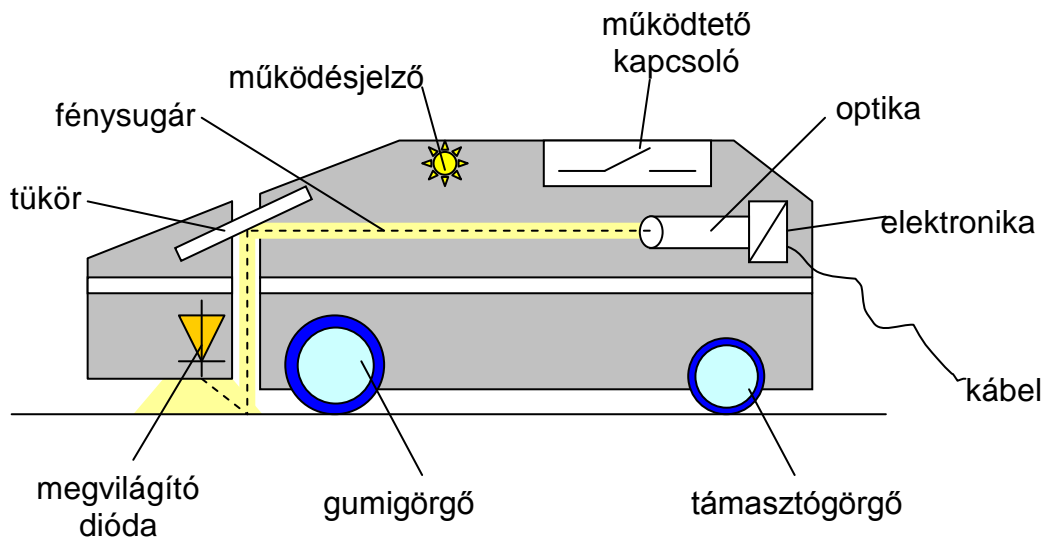
Az optika feladata, hogy a képet megfelelő minőségben az érzékelőre juttassa. Általában több lencséből álló lencserendszert alkalmaznak.

A megvilágító egység feladata a bevinni kívánt dokumentum egyenletes fényerővel történő megvilágítása, színes scannereknél fontos paraméter a fény spektruma is. Az egyenletes és kellő intenzitású fényt nagyfrekvenciás fénycsövekkel érik el.

A letapogatáshoz az érzékelőt vagy a dokumentumot kell mozgatni, s ezt a feladatot a mozgó mechanika végzi el.

Az elektronika feladata az egységek vezérlése, és a részegységek megfelelő tápellátása. Az analóg jelet digitális jellé kell alakítani, mert az interface-n keresztül már digitális információ halad. Az átalakítást ADC (Analog to Digital Converter) végzi, mely a CCD maximális kimeneti jelét 256 vagy 1024 elemi egységre osztja, és 8, ill. 10 biten ábrázolja. Az ADC kimenete egy ún. átmeneti memóriához csatlakozik, és így jutnak az adatok az interface-n keresztül a számítógépbe.

### 9.4.1 KÉZI SCANNER



9.3. ábra: Kézi scanner felépítése

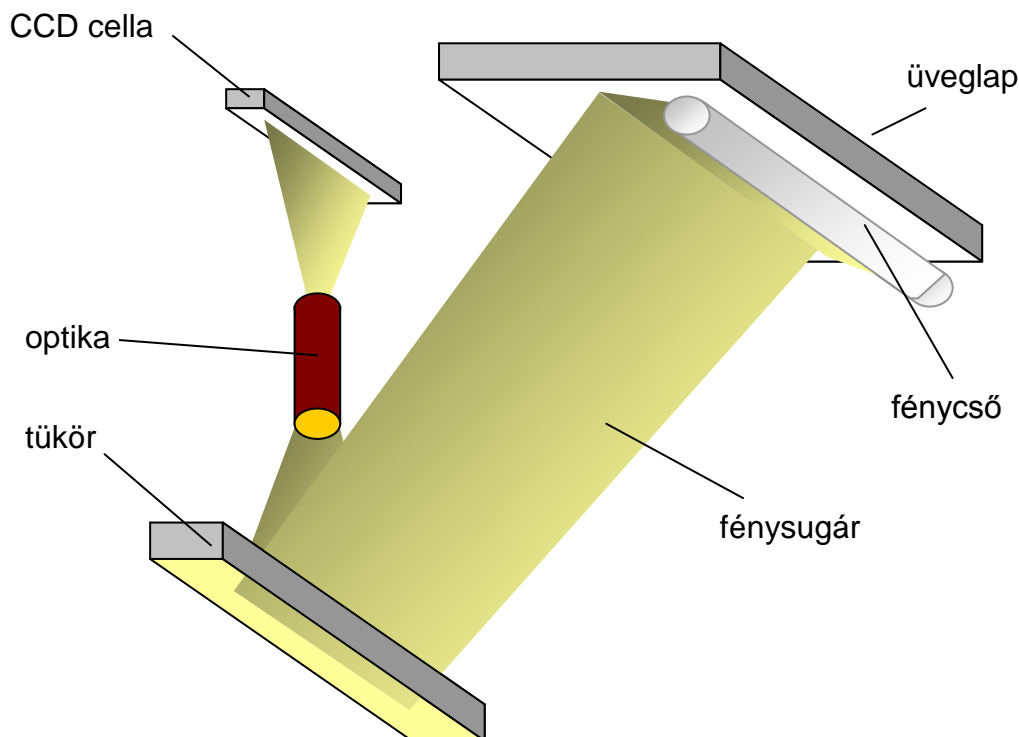
A kézi scannereket kézzel kell mozgatni a dokumentum fölött, a mozgatást két görgő és egy szinkronizáló görgő segíti. Az egyenes húzás nincs biztosítva, ezért gyakori a görbe vonalú mozgás. A műveletet sík felületen kell végezni.

A megvilágítás LED-ekkel történik. az érzékelés 105 mm szélességben megy végbe, vagyis egy A/4-es méretű dokumentum csak két lépésben digitalizálható, azaz két képet kapunk, melyeket egy szoftver egyesít.

A felbontást 100-400 dpi között kapcsolóval kell állítani, a fényerő pedig egy potenciométerrel állítható tetszés szerint.

A kézi scannerek soros interface-szel rendelkeznek, és saját illesztőkártya tartozik hozzájuk.

### 9.4.2 SÍKÁGYAS SCANNER



9.4. ábra: A síkúgyas scanner felépítése

A síkúgyas scanner esetén egy kocsi van felszerelve az érzékelő, az optika és a fénycső. A kocsi egy üveglap alatt mozog, a vezetést biztosító két acéltengelyen.

A dokumentumot képpel lefelé az üveglapra kell helyezni. A scannelés folyamata alatt a dokumentum áll. A mozgó kocsi fénye végigpásztázza a dokumentumot, a visszavert fény az optikán keresztül az érzékelőbe jut, az elektronika feldolgozza a jeleket, és az interface-n keresztül a számítógépbe juttatja. Több dokumentum folyamatos feldolgozására is van lehetőség automatikus lapadagoló alkalmazásával.

#### 9.4.3 LAPÁTHÚZÓS SCANNER

A lapáthúzóscannerekben a dokumentum mozog, gumigörgők segítségével biztosítják az egyeneses továbbítást. Van olyan típusú scanner, ahol több lap is behelyezhető egyszerre, ilyenkor a felszedő görgő garantálja az egymás utáni lapbehúzást.

Könyv bevitele nem lehetséges, vastagabb vagy fényesebb lapok továbbítása is problémás lehet. A dokumentum mérete is korlátozott. A továbbító rendszer miatt átlátszó fóliák sem használhatók.

#### 9.4.4 ÁLLVÁNYOS SCANNER

Az állványos scanner esetén a vízszintes dokumentumasztalból függőleges rúd áll ki, melyre állandó, vagy változtatható magasságban van rögzítve az érzékelőből és optikából kialakított egység. A dokumentumot (ez lehet három dimenziós tárgy is) az asztalra kell helyezni, majd megfelelő megvilágítás mellett a rúdon lévő érzékelő (mely léptetőmotorral mozgatott CCD) bescanneli a képet. A megvilágítást külső lámpasor biztosítja, ezért a kép minőségét a környezeti fény befolyásolhatja.

#### 9.4.5 DOBSCANNER

A dobscannernek foto-sokszorozó érzékelője van, melynek a jellemzője, hogy egyszerre csak egyetlen pontot érzékel, így minden egyes képpontot azonos jellemzőkkel tapogatnak le. A dokumentum egy forgódob palástján helyezik el, és visszavert fényel, vagy áteső fényel

tapogatja le. A fej a letapogatás során a palást mentén folyamatosan mozog, és a teljes dokumentum felett elhalad. A dob átlátszó anyagból készül, átmérője általában 50–150 mm.

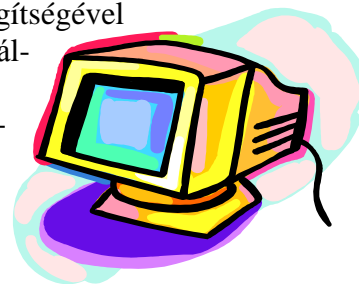
A dobscannereket a nyomdaipar alkalmazza, nagy felbontással, jó színérzékeléssel rendelkeznek, az elektronika általában tartalmazza a színrebotó egységeket is.

## 9.5 MONITOR

A monitor feladatai:

- ☞ megjeleníti az adott alkalmazás által generált információkat
- ☞ kontrollfunkciókat lát el, azaz a felhasználó a monitor segítségével nyomon követi és ellenőrzi a különböző bemeneti eszközök által kiadott parancsokat és azok eredményét.

A monitor a karaktereket és a képeket a képernyőn lévő képpontok, ún. pixelek kigyújtásával jeleníti meg. A színes monitoroknál a képet egy vörös, egy zöld és egy kék összetevő alkotja, e három elem különböző intenzitással történő kivilágításával a monitor képes színének megváltoztatására (additív színkeverés).



A monitorok fontos jellemzője a méret. A képcsőátló mérete határozza meg a megjelenítő méreteit. A méreteket collban adják meg. Elterjedt képcsőméret napjainkban a 14". A nagyobb méretű monitorok élvezhetőbb képet biztosítanak.

A monitorok többsége ma már digitális vezérlést tartalmaz, mely a különböző felbontásokat automatikusan választja ki és optimalizálja a képméretet. A 15"-os és a nagyobb monitoroktól elvárható az 1024x768-as felbontás, és hogy legalább 70-72 Hz-es non-interlaced módban jelenítse meg a képet.

A másik fontos jellemző a monitor frekvenciája. A teljes képernyőnyi kép megjelenítéséhez elektronsugárral kell ismétlődően pásztázni a képernyőn. Mindegyik pásztázás során megvilágítja a vörös, a zöld és a kék képpontokat. Az állókép tartásához is folyamatosan kell frissíteni a képpontokat. Ezért a monitornak nagyon gyorsan kell mozgatnia az elektronsugarakat. Ha azt szeretnénk, hogy villogásmentes képet kapjunk, akkor legalább 60 képet kell kirajzolnia az elektronsugárnak másodpercenként, ezt hívjuk *sorfrekvenciának*. Ez VGA módban pl.  $640 \times 480 = 28800$  sor másodpercenként. Az elektronsugárnak  $28800 \times 640 = 18432000$  képpontot kell kigyújtani másodpercenként, ezt *pontfrekvenciának* hívjuk. A monitorok a pixelsorokat balról jobbra frissítik, amikor az elektronsugarak elérik a képernyő jobb szélét, a monitor kikapcsolja az elektronágyúkat, és rááll a következő sor elejére. Ha az elektronsugár eléri a képernyő alját, a monitor kikapcsolja az elektronágyúkat, és visszaállítja őket a képernyő bal felső sarkába. Azt a számot, amellyel a monitor a teljes képernyőt frissíti, *független frissítési rátának* nevezik. A szokásos független frissítési ráta a másodpercenkénti 50-től 100-ig terjedő tartományban van.

A felbontás a monitor által megjeleníthető pixelek számának leírására szolgál, szorzatként adják meg, az egy képernyősorban található képpontok számának és a képernyősorok számának szorzataként. Minél nagyobb a felbontás, annál élesebb a kép. A monitor élességének jellemzője a képponttávolság, amit a két szomszédos színű képpont közötti távolság határoz meg. Ez a távolság általában 0,21–0,31 mm közötti érték.

A kép megjelenítéséhez állandóan jeleket kell továbbítani a videokártyáról a monitorhoz. Nagyobb felbontás esetén a monitor által frissítendő pixelek száma több, és vannak olyan monitorok, amelyek erre nem képesek. Az ilyen monitorok egy *interlacing*nek nevezett technika segítségével frissítik a képernyőt, melynek lényege, hogy az elektronsugár először a páratlan sorokat rajzolja ki, majd a páros sorokat, ami azt jelenti, hogy a kép félképekből áll össze, a hagyományos TV-készülékeknél is ezt alkalmazzák. A non-interlaced monitoroknál az elektronsugár teljes képet rajzol a képernyőre.



A gyártás során az alkatrészek értékei csak bizonyos tűréseken belül felelnek meg az előírt értéknek, ezért *torzítás* jön létre.

A monitoroknál jelentkező geometriai torzítások:

- ☞ trapéztorzítás
- ☞ paralelogrammatorzítás
- ☞ párnatorzítás
- ☞ hordótorzítás
- ☞ forgatás
- ☞ linearitás

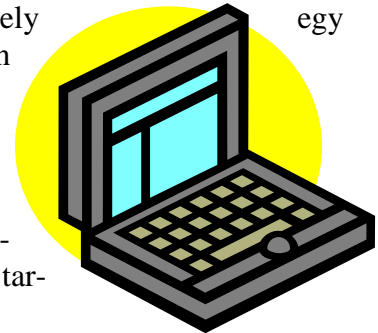
A gyártók törekednek olyan eszközök létrehozására, amelyek energiatakarékosan működnek, ergonomikus kialakításúak. Az ún. GREEN monitort alacsony sugárzás jellemzi, kicsi a reflexiója és villogásmentes a kép. A káros sugárzás minél kisebb szinten tartása a megfelelő árnyékolással és az eltérítő rendszer megfelelő kialakításával lehetséges.

## 9.6 LCD KIJELZŐK

Az LCD kijelzők a hordozható számítógépek képmegjelenítő eszközei (ma már létezik asztali változata is). Szerkezeti felépítésük szerint van aktív mátrix felépítésű (TFT – Thin Film Transistor). A megjelenített kép kontrasztja jó minőségű, a látószög jó, a feléledési idő 20 ms körüli, az előállítása bonyolult, az ára magas. A másik típus a passzív mátrix felépítésű (DSTN – Dual Super Twisted Neumatic). A megjelenített kép kontrasztja alacsony, a látószög kicsi, a feléledési idő 250 ms körüli, előállítása egyszerűbb, ára alacsonyabb.

A *monokróm kijelző* képpontjainak száma 640x480, valamint 64 árnyalat megjelenítésére képes. A képpontok mátrix elrendezésűek. A szemcsék átrendeződése időt vesz igénybe, ezért általában a kijelző vezérlés szempontjából két részre van osztva azért, hogy még elfogadható képfrissítést tudjanak biztosítani. Mind a monokróm, mind a színes LCD kijelzőnél segédfényforrásra van szükség ahhoz, hogy a megjelenített kép jól látható legyen, és a kontraszt is megfelelő legyen. A segédfényforrást a kijelző alatt helyezik el, mely egy fluoreszcens lámpa (Cathod Fluorescent Tube). Ez tulajdonképpen egy kis neoncső, 4-5 mm átmérőjű, 150 mm hosszú, teljesítményfelvétele 3-6 W, feszültségigénye 350-450 V váltakozófeszültség.

A *színes LCD kijelző*knél képponthármasokból alakítják ki az LCD-t, egy képpontban megtalálhatjuk az R, G, B szemcséket. Az LCD belsejében színszűrők vannak elhelyezve, mely a fényt felbontja a három alapszínre. Minden képponthoz három tranzistor tartozik, így a tranzistorok száma 3x640x480.



## 9.7 NYOMTATÓK

A nyomtatók feladata, hogy valamilyen forrásból származó információt papíron vagy más hordozón megjelenítsen.

A nyomtatók csoportosítása többféleképpen történhet:

Az *ütő nyomtatók* esetén egy festéket tartalmazó szalagból az ütőerő hatására festék préselődik a papírra. Előnye, hogy egyszerű, olcsó és több példányos nyomtatás is lehetséges. Hátránya, hogy zajos, lassú és gyenge minőségű a nyomtatás.

A *nem ütő nyomtatók* esetében a tinta vagy a festék felvitele és rögzítése fűjással, olvasztással, nagy nyomású hengerléssel történik. Az ilyen nyomtató csendes működésű, de egyidőben csak egy példányt tud nyomtatni. Általában a nyomtatás gyorsabb, a nyomtatási minőség jobb.

A *nyomtatási minőség* szempontjából (elsősorban a tús nyomtatóknál) három csoportot lehet megkülönböztetni. A leggyengébb minőséget az elnagyolt (draft) nyomtatás adja, ahol látszanak a karaktereket kialakító pontok. A következő fokozat az NLQ, csaknem levélminő-



séget jelent. A legjobb minőséget a levél minőségű (LQ) nyomtatás adja, itt a karakterek folyamatos vonalból állnak, szabad szemmel nem fedezhető fel folytonossági hiány.

A tintasugaras- és a lézernyomatók nyomtatási minőségét alapvetően a nyomtató felbontása határozza meg. A grafikus nyomtatás alapja az elemi pont, ilyen pontokból kell összerakni a karaktereket és a grafikus ábrákat. Az elhelyezhető elemi pont átmérője határozza meg a felbontást, mely jelzi a távolságegységenként elhelyezhető elemi pontok számát. A felbontás mértékegysége a dpi (dot per inch). A mai korszerű nyomtatók felbontóképességét 300, 600 és 1200 dpi tartományon belül állíthatjuk.

A nyomtatók funkcionális egységei:

- ☞ a karaktert megformáló és papírra juttató rész
- ☞ a papírtovábbító egység
- ☞ az elektronika
- ☞ az interface
- ☞ és a burkolat

### 9.7.1 MÁTRIXNYOMTATÓ

A mátrixnyomtató karakterformáló egysége a tűket tartalmazó mátrixfej. A betűket több pontból állítják össze. Régen igen elterjedtek voltak a 9x6-os mátrixú nyomtatók, ahol 9 függőleges pontból lehet a karakterek szarát megjeleníteni. Az alapvonal a második és a harmadik pont között húzódik. A 9 tűs mátrixnyomtatóhoz kifejlesztett nyomtatófejben 9 darab tű helyezkedik el egy függőleges oszlopban, rubin tűvezetőben. A tűket 9 darab elektromágnes vezérli. A nyomtatás úgy történik, hogy a fej vízszintes mozgásával egyidőben a tűk a festékszalagon keresztül a papírnak nyomódnak. A 9 tűs fej nem ad jó minőségű nyomtatási eredményt, ezért a későbbiekben több kisebb átmérőjű tűt tartalmazó nyomtatófejeket alkalmaztak.

A mátrixnyomtatók kellékanyaga a festékszalag, amely a leggyakrabban egy festékkal átitatott textilszalag, amit műanyag kazettában helyeznek el. A nyomtatás alatt a szalagot folyamatosan egy irányba kell mozgatni, a hajtás többnyire a fej mozgásából ered, de egyes gyorsnyomtatók esetében önálló motort alkalmaznak. A szalag a kazetta belsejében lazán van összehajtogatva, és néhányszor tíz méter hosszú, szélessége 8-25 mm között változik.

### 9.7.2 TINTASUGARAS NYOMTATÓ

A tintasugaras nyomtatók jellemzője a viszonylag jó minőségű (300-600 dpi) és a csendes nyomtatás, az alacsony ár, valamint az aránylag magas nyomtatási költség. A tintasugaras nyomtatókban a nyomtatófej fűvókákat tartalmaz. A fűvóka kis átmérőjű (gyakran 0,05 mm) lyukkal készül, melyen keresztül a tintacsepp kirepül a papírra. A csepp leválasztására több-fajta módszer létezik, így megkülönböztetünk folyamatos áramú vagy tintacseppes fejeket. A tintacseppes fej lehet piezokristályos vagy buborék működésű.



A *folyamatos áramú* fejnél a tinta folyamatosan áramlik a fűvókákon keresztül a papír felé és egy elektrosztatikus eltérítő rendszer téríti el, amikor nem szükséges, hogy a papírra kerüljön. A konstrukció előnye, hogy a tinta nem tud beszáradni a fűvókákba. Ezt a megoldást elsősorban színes nyomtatóknál alkalmazzák.

A *piezokristályos* nyomtatófej felépítése egy piezoelektromos kristályból, a fűvókából, a hozzá vezető kapillárisból és a tintatartályból áll. Működése során a kristályra adott feszültség hatására megváltoztatja méretét, és egy adott mennyiségű tintát lök ki a fűvókán keresztül a papírra. Ha a feszültséget megszüntetjük, akkor a tinta a tartályból a kapillárison keresztül a

nyomástérbe áramlik.

A *buborékfűvóka* hőt alkalmaz a csepp leválasztására. A csőben, melyben a tinta van, elhelyeznek egy kis fűtőszálat, és ezt a csepp leválasztásakor felfűtik. A felfűtés hatására a tinta egy része elgőzölög, buborékot hoz létre a csőben, ez a nyomás pedig kilöki a tintát a fűvókán keresztül.

A nyomtatófej 24, 36 vagy 48 fűvókát tartalmaz, a színes nyomtatonál nem ritka a 96 darab fűvóka sem. A fűvókákat függőleges vagy ferde vonalban helyezik el, a vezérlést szalagkábel segítségével viszik a fejhez.

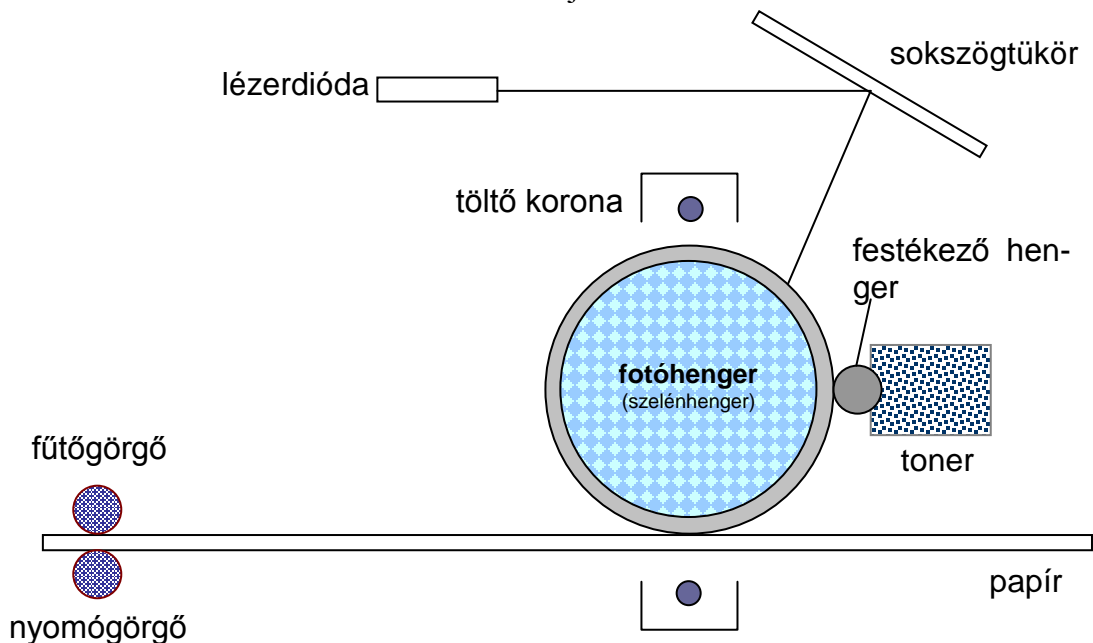
A *színes tintasugaras nyomtatók* ugyanazt a nyomtatófejet alkalmazzák, amit a fekete-fehér nyomtatók, de a négy színnek megfelelően négy példányban.

### 9.7.3 LÉZERNYOMTATÓ

A lézernyomtatóknál általában infravörös (800-1200 nm), szabad szemmel nem látható tartományban sugárzó félvezető lézert alkalmaznak. A félvezető lézer a ráadott tápfeszültség hatására sugároz, és könnyen modulálható. A moduláció a képpontok függvényében való villogtatást jelent. Néhány nyomtatonál hélium-neon lézert alkalmaznak.

A nyomtatási szélesség A/4-es papírméretnél 8 inch, ezért a lézersugarat ilyen szélességben kell az idő függvényében eltéríteni, hogy minden képpont a megfelelő pozícióba kerüljön. Az eltérítő egység egy nyolc oldalú hasáb alakú forgó tükör, mely 8 inch szélességű pásztázó sugárnyalábot állít elő. A tükröt általában üvegből készítik, a nyolc oldalát jó fényvisszaverő anyaggal vonják be, a forgatást egyenáramú motor végzi elektronikus fordulatszám-szabályozással. A tükör nyolc oldalának az eltérítés síkjában tökéletesen egybe kell esnie, mert különben a két sor egymásra rajzolódna, vagy túl nagy lenne a távolság a két sor között. Az eltérített sugár egyik végállásában helyezik el a sugárérzékelőt, mely jelet ad az elektronikának az eltérítés szinkronizálásához és a motor fordulatszámának szabályozásához. Az eltérítés után létrejön a vízszintes felbontás, így a képpontok vízszintesen helyes pozícióba kerülnek.

Az optika feladata a lézersugár fókuszálása és a torzulások kiküszöbölése. Az optikától adott távolságra helyezkedik el a fényérzékeny henger, itt kell a sugárnak fókuszban lenni. A henger az eltérítés középső pozíciójában közelebb van az optikához, mint a két szélső helyzetben, azért közepén a fényt lassabban kell a lencsén átengedni, mint a széleken, hogy mindenütt azonos fázisban érkezzon a hengerre. Az optika további feladata a felbontás által megkövetelt foltátmérő biztosítása az eltérítés teljes vonalában.

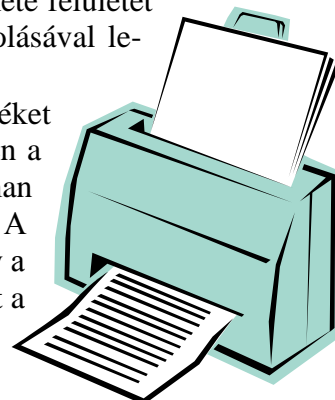


9.5. ábra: A lézernyomtató elvi felépítése

A fotóhenger 4-10 cm átmérőjű alumínium csőre felvitt szerves fotóvezető anyag (pl. szelén), az elektromos töltést jól megtartja, megvilágítás hatására azonban a megvilágított helyen vezetővé válik, és a töltését elveszíti. Erre a hengerre vetíti rá a lézersugár a nyomtatandó szöveget vagy képet. A hengert a nyomtatás közben egyenletes sebességgel kell forgatni, a fordulatszám szinkronban van a tükrörforgatással, és gondoskodni kell elektromos töltöttségéről. Az alkalmazott átmérő általában nem teszi lehetővé, hogy egy teljes A/4-es oldal a hengeren maradjon, ezért egy hengerfordulaton belül meg kell kezdeni a papírra nyomtatást. Van olyan nyomtatók, ahol a henger helyett végtelenített szalagot használnak.

A töltő koronák vékony huzalok, melyek adott távolságra helyezkednek el a henger palástjától, feszültségük néhány száz és néhány ezer volt között változik. A töltő korona feladata, hogy a megvilágítás előtt a hengert egyenletesen feltöltse. A lézersugár hatására a megvilágított helyen a töltés elveszik, tehát a megvilágítás után a henger egy alkotójának a feszültsége vagy nulla, vagy néhány ezer volt. Ez a sok kicsi kondenzátor elhalad a festékező henger előtt, és a töltött helyeken festék tapad rá. Ezzel a módszerrel fekete felületet a lézer sugárforrás kikapcsolásával, fehér felületet annak bekapcsolásával lehet elérni.

A festékező egység nagyon finomra őrölt (néhány  $\mu\text{m}$ ) porfestéket (toner) tartalmaz, ezt a finom porfestéket kell egyenletes rétegben a képtartalomnak megfelelően felvinni a papírra. A festékbe finoman őrölt vasport kevernek, ezáltal a festék mágnesezhetővé válik. A henger palástja mentén elhelyezett állandó mágnesű forgó tengely a festéktárolóból mindig adott mennyiségű festéket választ le, és azt a hengertől adott távolságra tartja. A nyomtatás hatására a festék az ún. mágneses keférről a hengerre, majd a papírra vándorol, és a vaspár visszamarad a keféren.



A papírt a hengerről leválasztva a *beégető egység*be kell vezetni, mely 150 °C körüli hőmérsékleten a festéket a papírba égeti. Az egység egy fűthető hengert tartalmaz, amelynek hőmérséklete pontosan szabályozható. A festékezett papír egyenletes sebességgel áthalad a beégető henger és egy gumihenger között, a festék pedig rögzítődik.

A *színes lézernyomtatót* négy xerografikus egységgel (négy henger és négy különböző színű toner) építik fel. Elsőként a lézer felírja az első színhez tartozó információt a hengerre, a festékező egység festékkel látja el, és a festék átkerül a papírra, majd a beégetéssel fixálódik. Ezután a folyamat a második, a harmadik majd végül a negyedik szín feldolgozásával folytatódik. A papír a négy egység előtt elhaladva folyamatos pályán mozog, a lézerforrás egymás után világítja meg a négy hengert.

#### 9.7.4 HÖNYOMTATÓ

A legősibb eljárásokat alkalmazó hagyományos hőnyomtató nem használ sem festéket, sem szalagot, csupán a nyomtatófejbe épített fűtőellenállásokkal operál. A fej és a hőre érzékeny speciális papír közti fizikai érintkezés során alakulnak ki a karakterek. A nyomtatási minőség közepes, mert a hőérzékeny papír nem elég stabil, fény hatására lassan tönkremegy. Ezért ezt az eljárást csak ritkán, igénytelen helyeken alkalmazzák.

Minőségi nyomtatás előállítására jóval alkalmasabb a hőátvitelen alapuló eljárás. Itt is megtaláljuk a nyomtatófejbe épített fűtőellenállásokat, ezek azonban nem a papír megpörkölésével, hanem egy festék-viasz keverékből készült szalag felfűtésével rajzolják ki a kívánt karaktereket. A szelektíven alkalmazott fűtés a megfelelő helyeken olvasztja meg a viaszt, így a festék szabaddá válik és rátapad a papírra.

Az olcsóbb típusok mozgó fejjel, míg a kimondottan drága kategóriába tartozó berendezések több ezer ellenállásból álló, rögzített fejjel rendelkeznek. A hőátviteli technológiával működő nyomtatókat elsősorban három vagy négy színnyomással alkalmazzák. A különböző színű rétegeket egymás után készíti el a nyomtatófej.

#### 9.7.5 SZUBLIMÁCIÓS NYOMTATÓ

A csúcsot árban és minőségben egyaránt a szublimációs hőnyomtatók tartják. A szublimációs jelző arra utal, hogy a nyomtatókban valamilyen szilárd test az átmeneti folyékony halmozott állapot kihagyásával alakul át légneművé. Az eljárásnál tehát csak speciális festék használható. A nyomtatópapírt egy polimerizált bevonat fedi, amikor pedig a fej melegíti a filmet, a két réteg közti kontaktusban a festék átrendeződik a papír irányába. Az így készült nyomatok fényképminőségűek. A hőnyomtatók hátránya a magas ár, a lassúság, a hatalmas méret és tömeg, valamint a speciális anyagok használata miatt kialakuló magas nyomtatási költség.

### 9.8 PLOTTEREK

A plotter a számítógép által előállított rajzok megjelenítésére szolgál. A készülék elsősorban tervek, műszaki rajzok készítésére alkalmas.

A plotterek működési elv szerint többfélék lehetnek, de közülük a tollas és a tintasugaras plotterek a legelterjedtebbek.

A tollas plotterek, mint ahogy ez nevükből is kitűnik tollakkal rajzolnak. Csőtollakkal, filctollakkal, golyóstollakkal, egyesek ceruzákkal. A tollak lehetnek egyszer használatosak, vagy utántölthetőek, valamint megkülönböztetjük őket szín, vastagság és aszerint is, hogy milyen médiához lehet őket használni. Ezeket a tollakat egy speciális befogószerkezet fogadja és működteti. Vannak 4, 6 és 8 tollat működtető befogók. Részben a tollak problémái miatt (folyton beszáradnak, elkopnak, nehézkes az utánpótlás) részben pedig a rajzolási sebessége, szolgáltatásai miatt ma már a tollas plottereket elavultnak mondhatjuk.

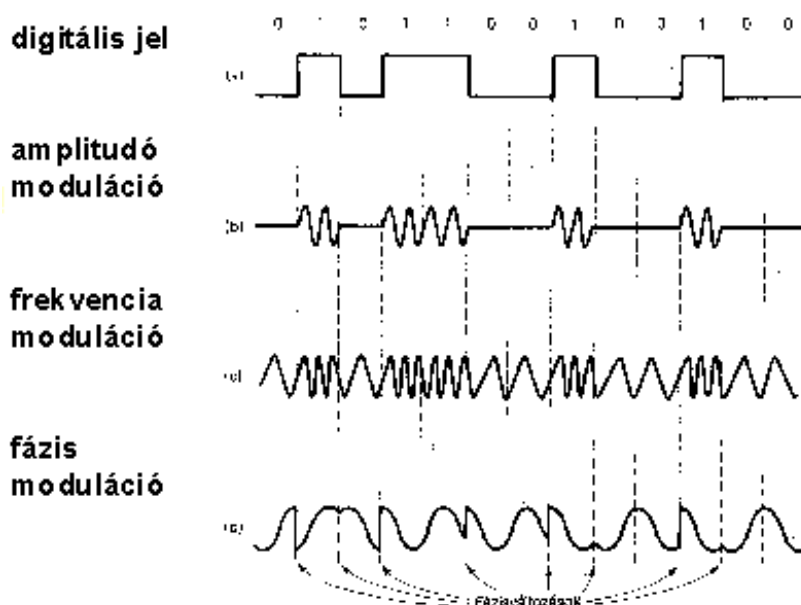
Az utóbbi időben a tintasugaras plotterek teljesen egyeduralmukodóvá váltak és ezáltal a tollas plotterek szinte teljesen eltűntek. Ez főképp annak köszönhető, hogy nagyságrenddel gyors-

sabban működnek előző társaiknál, valamint nemcsak vonalas ábrákat hanem részletgazdag színes posztereket is készíthetünk vele, ami a tollas technológiánál elképzelhetetlen volt.

## 9.9 MODEMEK

Telefonvonalon keresztül az elméleti információátviteli határ 1-2 Mbit/s. Azonban a telefonközpontokban a vonalak szűrőkön mennek keresztül, amelyek a váltakozóáramú jelek nem használt részeit levágják. Ennek a műveletnek az eredménye egy tiszta jel, mely feldolgozható. A telefonközpontok csak a 300 Hz és 3,3 kHz közé eső frekvenciájú jeleket engedik át. Az információt célszerű tehát ebbe a tartományba eső szinuszos jel valamely jellemzőjéhez rendelni. Ezt a szinusz hullám modulálásával tudjuk megvalósítani. Azt az eszközt, mely a bemenetére adott digitális jel vezérlésével a modulációt elvégzi, illetve a modulált, analóg jelet visszaalakítja digitális bitsorozattá (demodulálja), *modemnek* nevezzük (modulátor – demodulátor).

Minden szinuszos jelnek három jellemzője van, melyet a moduláció során felhasználhatunk:



9.6. ábra: Modulációs módok összehasonlítása

### 9.9.1 AMPLITÚDÓ MODULÁCIÓ

Egy adott amplitúdójú jel megléte vagy a hiánya hordozza az információt, vagyis az átvinni kívánt bitsorozat (moduláló jel) függvényében változtatjuk meg a hangfrekvenciás jel (vivőjel) amplitúdóját. Így keletkezik az amplitúdó-modulált (AM) jel. A megvalósítás jól működik, de csak lassú átvitel oldható meg.

A jelek visszaállítására komparátorokat használnak. Ezek olyan eszközök, melyek két bemenettel és egy kimenettel rendelkeznek. A bemenetre kapcsolt feszültségek szintjétől függ a kimenet állapota. Az egyik bemenet a nem invertáló, a másik az invertáló. A bemenetek kapcsolása szerint a komparátor lehet invertáló és nem invertáló. A két bemenet négy különböző állapotot vehet fel.

Az egyik bemenetre minden esetben egy referencia-feszültség kerül. A bemenetre adott jel amplitúdóját hasonlítja össze ezzel a referencia-feszültséggel. Lehetséges esetek:

1. A referencia-feszültség a nem invertáló bemeneten van. A vizsgálandó jelet az invertáló bemenetre kapcsoljuk. Ha a jel amplitúdója nagyobb, mint  $U_{ref}$ , akkor a kimenet alacsony szintű lesz.

2. A referencia-feszültség a nem invertáló bemeneten van. A vizsgálandó jelet az invertáló bemenetre kapcsoljuk. Ha a jel amplitúdója kisebb, mint  $U_{ref}$ , akkor a kimenet magas állapotot vesz fel.
3. A referencia-feszültség az invertáló bemeneten van. A vizsgálandó jelet a nem invertáló bemenetre vezetjük. Ha a jel amplitúdója nagyobb, mint  $U_{ref}$ , akkor a kimenet magas szintű lesz.
4. A referencia-feszültség az invertáló bemeneten van. A vizsgálandó jelet a nem invertáló bemenetre kapcsoljuk. Ha a jel amplitúdója kisebb, mint  $U_{ref}$ , akkor a kimenet alacsony szintű lesz.

### 9.9.2 FREKVENCIA MODULÁCIÓ

Ezt a modulációt használták először a modemeknél. Nagyon jó a zajtűrése és a biteket hordozó frekvenciákat egyszerű szűrőkkel nagyon könnyű szétválasztani.

A digitális információban lévő 1 és 0 állapotokat eltérő frekvenciájú szinuszos jelek valósítják meg. A frekvenciakülönbségnek elég nagynek kell lennie ahhoz, hogy biztonságosan szét lehessen választani.

A frekvenciamoduláció egyik lehetséges megvalósítása: egy oszcillátor jelét egy 10-es osztóval leosztjuk. A nagyobb frekvencia felel meg az 1-eseknek, míg ennek egytizede lesz a 0 állapot. Mind a két kimenet egy kapun halad keresztül, melynek állapotát a vezérlő bemenetére adott logikai jelekkel lehet meghatározni. A digitális információ kapcsolgatja ezeket a kapukat. Ha a továbbítandó információ 1-es, akkor az egyik kaput kell kinyitni és a másikat lezárni. Ha az információ 0, akkor a másik kapu lesz nyitva és az első zárva. A két kapu kimenetét már csak egy vonalra kell kapcsolni. Ezt egy analóg VAGY kapu segítségével tudjuk megoldani.

### 9.9.3 FÁZIS MODULÁCIÓ

A modulációs módok közül ez a legbonyolultabb. A fázismódosítás nem a biteket valósítja meg, hanem az átmeneteket. Ha a  $360^\circ$ -os fázisstartományt felosztjuk négy részre, akkor a sík négy irányába mutató, egymással  $90^\circ$ -os szöget bezáró fázisvektorral lesz leírható. Mivel itt négy különböző állapot van, ezért négy fázisvektorral összesen két bitet lehet kódolni. A négy állapot a következő:

1. 00: Ennek az átmenetnek a  $0^\circ$ -os fázistolás felel meg, tehát nem történik fáziseltolódás.
2. 01: Ez  $90^\circ$ -os fázistolást jelent.
3. 10: Az ilyen átmenethez a szinuszos jelet  $180^\circ$ -kal kell eltolni a kezdőfázishoz képest.
4. 11: Az átmenetet  $270^\circ$ -os fáziseltolás fogja jelölni.

Ha szeretnénk az adatátviteli sebességet növelni, az két módon lehetséges. Vagy a fázishelyzetek vagy az amplitúdó értékek számát megnöveljük.

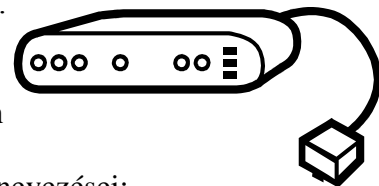
### 9.9.4 A MODEMEK FELÉPÍTÉSE

A modemek önállóan működni képes számítógép perifériák. Mielőtt működni kezdenének, a számítógépnek fel kell programozni, különböző szabványos és csak az adott modemre érvényes parancsokkal vezérelni és az állapotát folyamatosan ellenőrizni.

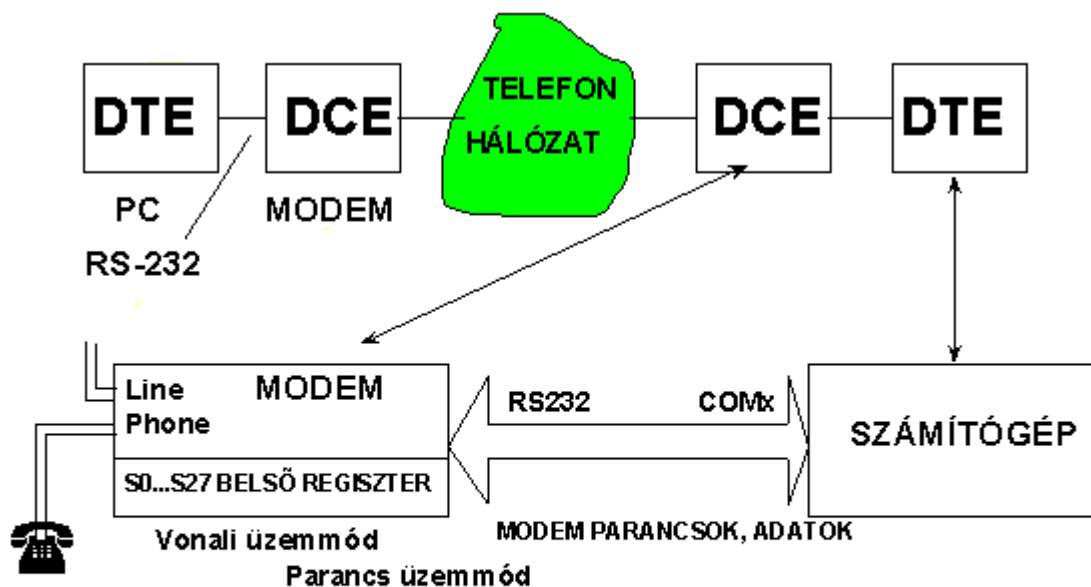
A kapcsolatban két egységet kell megkülönböztetni, ezek elnevezései:

DTE (Data Terminal Equipment): adatvég-berendezés, a számítógép vagy a terminál neve.

DCE (Data Circuit-Terminating Equipment): adatáramkörü-végberendezés, a modem hivatalos elnevezése.



A két modem a telefonhálózaton keresztül kerül összeköttetésbe.



**DTE=Data Terminal Equipment - Adatvég berendezés**

**DCE=Data Circuit-terminating Equipment - Adatáramkör végberendezés**

9.7. ábra: Modem-számítógép kapcsolat

A számítógép és a modem összekapcsolása kétféle módon történhet meg. A klasszikus mód a soros porton történő összeköttetés. Azonban ma már olyan modemek is elterjedtek, melyek a számítógép alaplapjába vannak dugaszolva, tehát a szabványos ISA vagy PCI sínbe illeszkenek. A belső felépítés néhány különbségtől eltekintve azonos, az eltérés csupán a csatlófelület kialakításában van. Továbbá a külső modem rendelkezik a kapcsolat ellenőrzését megvalósító LED-ekkel.

A modemek működése egy céláramkörön alapul, minden fontos feladatot ez végez el. Ezek a következők:

- Elvégzi a soros-párhuzamos, párhuzamos-soros átalakítást.
- Kezeli a megszakításokat.
- A programtól kapott adatok alapján előállítja a tárcsakódot vagy tárcsaimpulzusokat.
- Detektálja a csengető feszültséget.
- Kétirányú összeköttetést képes teremteni a szabványos távbeszélővonallal.
- Előállítja a működéshez szükséges órajelet.

Az integrált áramkör programját egy EEPROM tartalmazza. A címdekóder nagyon egyszerű kialakítású, logikai áramkörökkel van megvalósítva. A kártya a COM1 vagy a COM2 port címen tud kommunikálni a számítógéppel, jumperekkel lehet a két cím közül választani. A címnek megfelelően a megszakítást is be kell állítanunk, melyre ugyancsak jumper szolgál. A processzor által előállított, modulált analóg feszültséget egy transzformátoron keresztül küldjük a telefonvonalra. Külső modem esetén szükség van még egy külső hálózati tápegységre is.

Mielőtt a modemet használni tudnánk, fel kell programozni. Ezt általában a használt program végzi el helyettünk. A számítógép a szöveges formájú parancsokat a soros vonalon keresztül küldi el a modemnek. Az parancs üzemmódban értelmezi azokat, majd általában OK üzenettel válaszol.

A modemek többségében 28 regiszter található. Ezek határozzák meg a modem üzemmódjait és működési paramétereit. Egyes jellemzők egy nem felejtő memóriában tárolódnak, ezek bekapcsolás után az alapbeállítását jelentik.

Ha egy modemmel adatokat viszünk át a telefonvonalon, akkor lefoglaljuk a hang átvitelére használt csatornát. Ebből következik, hogy vagy adatokat, vagy hangot viszünk át. Előfordulhat igény arra, hogy a modem felismerje a hangokat és lehetővé tegye az azokkal történő munkát. Erre találták ki az ún. voice-modemeket. Ezek képesek a hangot felismerni és egy analóg-digitális átalakítóval digitálissá, illetve a digitális jelet analóggá alakítani és a számítógépen tárolni.

A ma kapható modemek szinte kivétel nélkül képesek FAX készülékként is működni. Ez a speciális üzemmód lehetővé teszi képanyagok gyors továbbítását két pont között. Azonban nem szükséges mindkét oldalon számítógépnek lenni, elegendő egy FAX készülék is. Ezek egy modem, egy lapolvasó és egy nyomtató összeintegrálásából születtek.



# TARTALOMJEGYZÉK

<b>1</b>	<b>INFORMÁCIÓ, INFORMÁCIÓS RENDSZEREK</b>	<b>1</b>
<b>2</b>	<b>A SZÁMÍTÁSTECHNIKA TÖRTÉNETE</b>	<b>3</b>
2.1	A SZÁMRENDSZEREK KIALAKULÁSA	3
2.2	A SZÁMÍTÓGÉPEK FEJLŐDÉSE	3
2.3	SZÁMÍTÓGÉP-GENERÁCIÓK	5
<b>3</b>	<b>A SZÁMÍTÓGÉP MŰKÖDÉSÉNEK ALAPJAI</b>	<b>6</b>
3.1	NEUMANN ELVEI	6
3.2	A SZÁMRENDSZEREK	6
3.2.1	KETTES (BINÁRIS) SZÁMRENDSZER	7
3.2.2	MIÉRT JÓ A KETTES SZÁMRENDSZER	7
3.2.3	TÍZES-KETTES (DECIMÁLIS-BINÁRIS) SZÁMRENDSZERBELI SZÁMKONVERZIÓK	7
3.2.4	BINÁRIS SZÁMOK ÁTALAKÍTÁSA DECIMÁLIS SZÁMOKKÁ	8
3.2.5	HEXADECIMÁLIS SZÁMOK	9
3.2.6	DECIMÁLIS SZÁMOK KONVERTÁLÁSA TETSZŐLEGES ALAPÚ SZÁMRENDSZERBE	10
3.2.7	ARITMETIKAI MŰVELETEK BINÁRIS SZÁMRENDSZERBEN	10
3.3	ADATÁBRÁZOLÁS A SZÁMÍTÓGÉPEN	12
3.3.1	A FIXPONTOS GÉPI SZÁMÁBRÁZOLÁS	12
3.3.2	LEBEGŐPONTOS SZÁMÁBRÁZOLÁS	13
3.4	DECIMÁLIS SZÁMOK, KÓDRENDSZEREK	15
3.4.1	DECIMÁLIS SZÁMOK A SZÁMÍTÓGÉPEN	15
3.4.2	KÓDRENDSZEREK	16
3.5	RELÁCIÓS MŰVELETEK A SZÁMÍTÓGÉPEN	17
3.6	A SZÁMÍTÓGÉPES LOGIKA	17
<b>4</b>	<b>A SZÁMÍTÓGÉPEK ARCHITEKTÚRÁJA</b>	<b>20</b>
4.1	A SZÁMÍTÓGÉPES HARDVER SZERKEZETE	20
4.2	A SZÁMÍTÓGÉPES SZOFTVER SZERKEZETE	21
<b>5</b>	<b>A KÖZPONTI EGYSÉG</b>	<b>22</b>
5.1	A SZÁMÍTÓGÉPHÁZ ÉS A TÁPEGYSÉG	22
5.1.1	TÁPEGYSÉG	22
5.1.2	NYOMÓGOMBOK, KAPCSOLÓK, KIJELEZŐK	24
5.1.3	HANGSZÓRÓ	24
5.1.4	BŐVÍTŐ HELYEK	24
5.1.5	HÁZTÍPUSOK	24
5.2	AZ ALAPLAP	26
5.3	A MIKROPROCESSZOR	27
5.3.1	A MIKROPROCESSZOROK TECHNOLÓGIÁJA	27
5.3.2	A MIKROPROCESSZOROK CSOPORTOSÍTÁSA	27
5.3.3	A MIKROPROCESSZOROK FELÉPÍTÉSE	28

5.3.4	REGISZTEREK	28
5.3.5	ARITMETIKAI ÉS LOGIKAI EGYSÉG	28
5.3.6	A VEZÉRLŐEGYSÉG (CU)	29
<b>5.4</b>	<b>BIOS</b>	<b>30</b>
<b>5.5</b>	<b>A CPU ÓRÁJA</b>	<b>31</b>
<b>5.6</b>	<b>A SZÁMÍTÓGÉP BELSŐ TÁROLÓJA</b>	<b>31</b>
<b>5.7</b>	<b>ADATÁTVITELI RENDSZER</b>	<b>32</b>
5.7.1	CSATORNARENDszerű ADATÁTVITEL	32
5.7.2	ADATÁTVITEL SÍNRENDSZERREL	34

## **6 CSATOLÓ- ÉS BŐVÍTŐKÁRTYÁK** **35**

---

<b>6.1</b>	<b>MONITORVEZÉRLŐ KÁRTYÁK</b>	<b>35</b>
6.1.1	MDA KÁRTYA	35
6.1.2	CGA KÁRTYA	35
6.1.3	HERCULES KÁRTYA	35
6.1.4	EGA KÁRTYA	36
6.1.5	VGA KÁRTYA	36
6.1.6	GYORSÍTÓKÁRTYÁK	37
6.1.7	AGP	39
<b>6.2</b>	<b>A MULTI IO KÁRTYA</b>	<b>39</b>
6.2.1	MEREVLEMEZES VEZÉRLŐ	39
6.2.2	FLOPPY ILLESZTŐ	40
6.2.3	SOROS VONALI ILLESZTŐ	40
6.2.4	PÁRHUZAMOS ILLESZTŐ FOKOZAT	41
6.2.5	BOTKORMÁNY ILLESZTŐ	41
<b>6.3</b>	<b>A HANGKÁRTYA</b>	<b>41</b>
<b>6.4</b>	<b>SCSI CSATOLÓ</b>	<b>43</b>

## **7 A PROGRAMOK VÉGREHAJTÁSA** **45**

---

<b>7.1</b>	<b>GÉPI KÓDÚ UTASÍTÁSOK</b>	<b>45</b>
<b>7.2</b>	<b>A CÍMZÉSI MÓDOK</b>	<b>46</b>
7.2.1	A KÖZVETLEN ADATHIVATKOZÁS	46
7.2.2	A KÖZVETLEN CÍMZÉS	47
7.2.3	A REGISZTERES CÍMZÉS	47
7.2.4	A REGISZTERES INDIREKT CÍMZÉS	48
7.2.5	AZ INDIREKT CÍMZÉS	48
7.2.6	A BÁZISREGISZTERES CÍMZÉS	49
7.2.7	AZ INDEXREGISZTERES CÍMZÉS	51
7.2.8	UTASÍTÁSTÍPUSOK	52
7.2.9	AZ IBM PC UTASÍTÁSSZERKEZETE	52
<b>7.3</b>	<b>A SZÁMÍTÓGÉP MEGSZAKÍTÁSI RENDSZERE</b>	<b>53</b>

## **8 HÁTTÉRTÁRAK** **55**

---

<b>8.1</b>	<b>MÁGNESSZALAGOS ADATTÁROLÓ</b>	<b>55</b>
8.1.1	ADATTÁROLÁS A MÁGNESSZALAGON	55
8.1.2	A MÁGNESSZALAGEGYSÉG MŰKÖDÉSE	56
8.1.3	MŰVELETVÉGGZÉS A MÁGNESSZALAGON	57
<b>8.2</b>	<b>MEREVLEMEZES TÁROLÓ</b>	<b>58</b>
8.2.1	FIZIKAI LEMEZKEZELÉS	59

	83	
8.2.2	A MEREVLEMEZ LOGIKAI KEZELÉSE	60
8.2.3	PARTÍCIÓS TÁBLA	60
8.2.4	ÁLLOMÁNYELHELYEZÉSI TÁBLA	61
8.2.5	KATALÓGUS	61
<b>8.3</b>	<b>HAJLÉKONYLEMEZ MEGHAJTÓ</b>	<b>61</b>
<b>8.4</b>	<b>A CD-ROM</b>	<b>62</b>
8.4.1	A LÉZERTECHNOLÓGIA	62
8.4.2	A LEMEZ FELÉPÍTÉSE	63
8.4.3	CD ÍRÁSA	64
8.4.4	CD LEMEZEK NAGYÜZEMI GYÁRTÁSA	64
8.4.5	CD-ROM-OK VEZÉRLŐI	64
8.4.6	CD TÍPUSOK	65
<b>9</b>	<b>BE/KIVITELI EGYSÉGEK</b>	<b>66</b>
<b>9.1</b>	<b>A BILLENTYŰZET</b>	<b>66</b>
<b>9.2</b>	<b>POZÍCIONÁLÓ ESZKÖZÖK</b>	<b>67</b>
9.2.1	EGÉR	67
9.2.2	TRACK BALL (HANYATTEGÉR)	67
9.2.3	DIGITALIZÁLÓ TÁBLA	68
9.2.4	BOTKORMÁNY	68
<b>9.3</b>	<b>FÉNYCERUZA</b>	<b>68</b>
<b>9.4</b>	<b>SCANNER</b>	<b>68</b>
9.4.1	KÉZI SCANNER	69
9.4.2	SÍKÁGYAS SCANNER	69
9.4.3	LAPÁTHÚZÓS SCANNER	70
9.4.4	ÁLLVÁNYOS SCANNER	70
9.4.5	DOBSCANNER	70
<b>9.5</b>	<b>MONITOR</b>	<b>71</b>
<b>9.6</b>	<b>LCD KIJELZŐK</b>	<b>72</b>
<b>9.7</b>	<b>NYOMTATÓK</b>	<b>72</b>
9.7.1	MÁTRIXNYOMTATÓ	73
9.7.2	TINTASUGARAS NYOMTATÓ	74
9.7.3	LÉZERNYOMTATÓ	74
9.7.4	HŐNYOMTATÓ	76
9.7.5	SZUBLIMÁCIÓS NYOMTATÓ	76
<b>9.8</b>	<b>PLOTTEREK</b>	<b>76</b>
<b>9.9</b>	<b>MODEMEK</b>	<b>77</b>
9.9.1	AMPLITÚDÓ MODULÁCIÓ	77
9.9.2	FREKVENCIA MODULÁCIÓ	78
9.9.3	FÁZIS MODULÁCIÓ	78
9.9.4	A MODEMEK FELÉPÍTÉSE	78
<b>TARTALOMJEGYZÉK</b>		<b>81</b>