

## 6. Memóriák

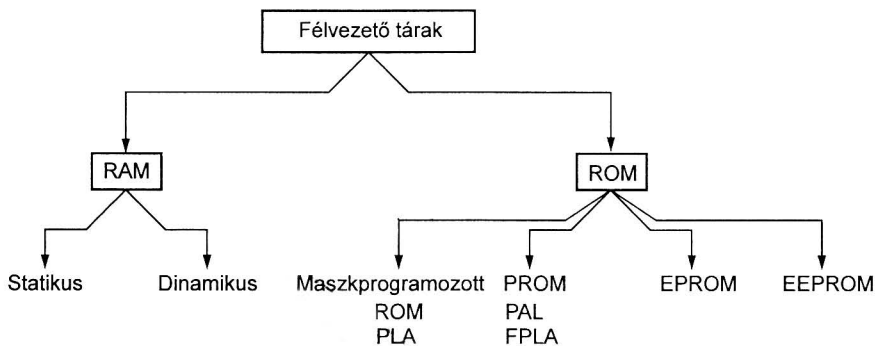
A memória vagy tár azon eszközök összessége, amelyek az információkat tetszés szerinti ideig megőrzik, és ahonnan azokat bármikor ki lehet olvasni. A számítógép utasításai által feldolgozandó bitek, bájtok és karakterek azon sorát, amely a memóriában is tárolható szóznak (*Word*) nevezik. A **szó hosszúsága** (amelyet rendszerint bitekben adnak meg), a számítógépek és egyben a memóriák egyik legfontosabb jellemzője.

A memóriában minden szó tárolására külön rekesz áll rendelkezésre. A rekeszt a cím (*Address*) azonosítja. A megcímezhető rekeszek száma és a rekeszben tárolható szó hosszúsága a memória szervezését mutatja. A **memória kapacitását**, amelyet bitben fejeznek ki, a rekeszek számának és a tárolható szó hosszúságának szorzata fejezi ki.

A címzés után csak egy bizonyos időintervallum eltelte után lehet a tárolt adatot (*Data*) kiolvasni, vagy újabbat beírni. Ez az intervallum az ún. **hozzáférési vagy elérési idő** (*Access time*). A kapacitás és a hozzáférési idő a memóriák két fontos jellemzője.

A memóriákat technológiai felépítésük és funkcionális jellemzőik szerint osztályozzák.

Technológiai felépítésük szerint a következő legelterjedtebb memóriatípusok különböztethetők meg: **félvezető, mágneses, optikai** és **magneto-optikai** memóriák. A mikroszámítógépek esetében a félvezető memóriáknak van a legnagyobb jelentőségük. Ezért a következőkben részletesebben csak a félvezető memóriákat fogjuk tárgyalni (6.1 ábra).



6.1. ábra. Félvezető táruk típusai

A memóriák egyik funkcionális jellemzője a megcímezett rekesz hozzáférési módja. Így megkülönböztethetők:

- ◆ **tetszőleges** (véletlenszerű) hozzáférésű memóriák,
- ◆ **soros** (szekvenciális) hozzáférésű memóriák,
- ◆ **asszociatív** memóriák.

A tetszőleges hozzáférésű memóriában bármely adat, függetlenül a címétől, ugyanolyan rövid idő alatt érhető el. Az ilyen típusú memória rövidített elnevezése **RAM** (az angol **R**andom **A**ccess **M**emory kifejezés rövidítése, azaz a véletlen hozzáférésű táré). A soros hozzáférésű memória esetében a keresett adat hozzáférési ideje különböző és függ a címétől, valamint a keresés kezdő címétől.

Az asszociatív memória bemenetén levő szót egyidejűleg összehasonlítja az egyes címeken tároltakkal, és azt a címet adja meg, amely által kijelölt rekeszben a tárolt szó megegyezik a bemeneti szóval. Az asszociatív memória használatos elnevezése **CAM** (az angol *Content Addressable Memory* kifejezés rövidítése).

Az információ beírhatósága szempontjából két típust lehet megkülönböztetni:

- ◆ *végleges* és
- ◆ *módosítható memóriát*.

A végleges beírás legtöbbször irreverzibilis szerkezeti változást hoz létre a memóriában, és utána a tartalma nem változtatható meg. Ebből a típusból az információt csak kiolvasni lehet. Rövidített elnevezése **ROM** (az angol *Read Only Memory* kifejezés rövidítése, azaz a permanens – csak olvasható – táré).

Az információ megőrzésének szempontjából:

- ◆ *statikus* és
- ◆ *dinamikus memóriákat*

különböztetünk meg.

A statikus memória az információt korlátlan ideig megőrizheti, feltéve, ha a tápfeszültsége nem szűnik meg. A dinamikus memória viszont időnként "felfrissítést" igényel, másként tartalma véglegesen törlődik.

A memóriákat még egyéb funkcionális jellemzőik szerint is meg lehet különböztetni, például a tápfeszültség kiesésekor megsemmisülő (RAM), illetve megmaradó tartalom (ROM) vagy a vezérlőjelek összetettsége alapján.

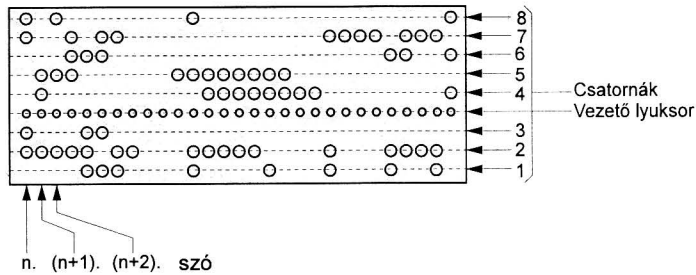
Megjegyzendő, hogy a **RAM** és **ROM** széles körben elterjedt elnevezések használata nem teljesen következetes. A **RAM** elnevezés rendszerint egy tetszőleges hozzáférésű memóriát jelöl, amely módosítható beírású. Tehát a **RAM** írható és olvasható is. A **ROM** elnevezés egy csak olvasható memóriát jelöl, amely tetszőleges hozzáférésű.

## 6.1. Soros hozzáférésű memóriák

A soros vagy szekvenciális hozzáférésű memória olyan típusú, amelyben az adatok csak egy adott lineáris sorrendben (*szekvenciában*) tárolhatók. Kiolvasás esetén az olvasóberendezés az elhaladó lineáris adatszekvenciából az éppen előtte levőt olvassa ki. Ezért a különböző címeken tárolt adatok hozzáférési ideje különbözik.

### 6.1.1. A lyukszalag

A soros hozzáférésű memóriák legklasszikusabb példája – bár jelenleg csak történeti jelentőséggel bír – a lyukszalag (*Paper tape*). Ennek az alapanyaga papír vagy műanyag. Az információ a szalag megfelelő kilyukasztásával tárolható (6.2. *ábra*:). A lyukszalag továbbítását az információhordozó lyukaknál kisebb vezető vagy továbbító lyukak biztosítják. A lyukszalagon tárolt szó a szalag hosszirányára merőleges lyuksorból áll. A szó minden egyes **1** vagy **0** értéket felvevő bitjét egy kilyukasztott, illetve lyukasztatlan hely képviseli. A kilyukasztott és a lyukasztatlan helyek kombinációja a szó kódolása szerint történik. A szalag hosszirányával párhuzamos lyuksort csatornának nevezik. Az alkalmazott kódrendszer függvényében 5-, 6-, 7- vagy 8-csatornás lyukszalag használatos.



6.2. ábra. Lyukszalag

A lyukszalagolvasó a  $t = 0$  időpontban az első szót olvassa ki. Ezután a szalagot egy szóval továbbítják, és  $t = 0 + T$  időpontban a lyukszalagolvasó a második szót olvassa ki. A  $T$  időköz a szalag továbbítására és a kiolvasó erősítők előkészítésére szolgál. A folyamat lépésenkénti megismétlése a szükséges információhalmaz kiolvasását eredményezi. A lyukszalagot a mágnesréteges memóriák teljesen kiszorították.

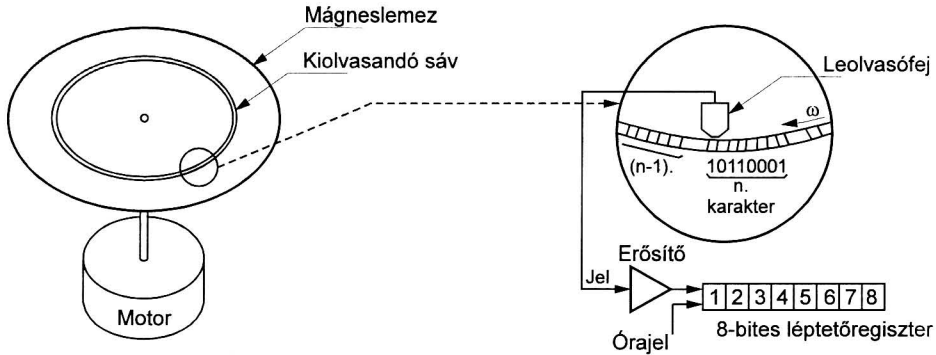
### 6.1.2. Mágnesréteges memóriák

A mágnesréteges memóriák tárolókapacitása messze felülmúlja a lyukszalagét (például egy hajlékony mágneslemez oldalainak egyikén tárolt 1 Mbájt egy 2,5 km hosszúságú lyukszalagot venne igénybe).

A mágnesréteges memóriák hozzáférési ideje egy-két nagyságrenddel kisebb, mint a lyukszalagé. A mágnesréteges memóriák működési elve hasonlít a magnetofonéhoz. Az írófej pólusait az információtól függő íróáram mágnesezi, amely a fej tekercsén folyik keresztül. A fej légrésében keletkező mágneses tér a mozgó tárközeg mágnesezhető rétegén maradandó mágneszettséget alakít ki. Ez a beírt adattól függően változó irányú, és ha elhalad az olvasófej előtt, akkor ennek a tekercsében áramimpulzust indukál. A felerősített impulzusokból alakítják ki a kiolvasott szót. A mágneses réteg hordozóanyagától függően megkülönböztethetők: mágnesszalagos, mágneskártyás, mágneslemezes és mágnesdobos memóriák. A mikroszámítógépek esetén a kazettás kivételű mágnesszalagos (angolul: *Cassette tape*), hajlékony mágneslemezes (angolul: *Floppy-disk*) és a mágneskártyás memóriák (angolul: *Magnetic card*) használatosak.

A kazettás mágnesszalagos memória hasonló felépítésű és méretű kazettás szalagot használ, mint a közismert kazettás magnetofon. A kettő között csak minőségi és tárterületi különbség van. Egy zenei felvételnél a szalag minősége miatt keletkező zavarokat alig vesszük észre. Ezek viszont a tárolt információkat megghamisíthatják. Ennek elkerülésére a tárolási célokat szolgáló kazetták minősége jobb. Egyes kazettás mágnesszalagos memóriák az információt hang formájában tárolják. Vagyis egy bizonyos tónus logikai **0**-t, míg egy másik logikai **1**-et jelent. Ennek a módszernek a hátránya, hogy egy szalagon kevesebb információt lehet tárolni, mint az előbbiekben felvázolt digitális felvételi módszer alkalmazásával.

A *hajlékony mágneslemezes memória* esetén a mágnesezhető réteg egy gyorsan forgó műanyag lemez mindkét felére van felvive. A mágneslemez felületén koncentrikus sávok mentén tárolják az információt. Az olvasófej úgy mozgatható, hogy akármelyik sávon tárolt információt leolvashassa.



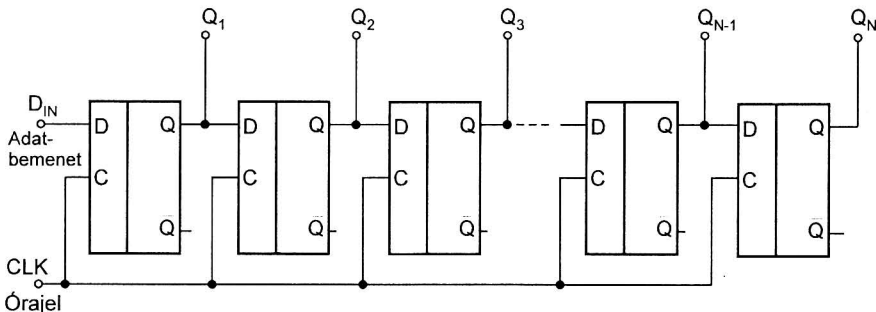
6.3 ábra. Mágneslemezes memóriaegység vázlatos felépítése

A hozzáférési idő két művelettől függ, elsősorban a leolvasófejet kell a kívánt sávra beállítani, ezután a forgó lemezen a sáv kívánt részének kell az olvasófej elé kerülnie és elhaladnia (6.3 ábra). A hajlékony mágneslemez kapacitásának növelését és hozzáférési idejének csökkentését főleg a mágnesezhető réteg jellemzőinek javításával, a fej és a lemez között levő távolság csökkentésével, valamint a lemez fordulatszámának növelésével lehet elérni. Az ún. Winchester-technológiájú mágneslemez-memóriák esetén a lemezt, a fejeket, valamint az ezeket beállító mechanizmust egy közös, zárt, cserélhető kazettába építették be. Így sikerült kiküszöbölni a lemezcsere-nél fennálló fejbeállítási eltérést, valamint a levegőben levő por és egyéb szennyeződés lerakódását a lemezre.

A **mágneskártyás memória** adathordozója, amint a neve is mutatja, egy mágneses réteggel ellátott hajlékony kártya. Az információ tárolása hasonlóképpen történik, mint az előbbi mágnesréteges memóriák esetében. A tárolandó információ felvételére a mágneskártya hosszanti irányban több sávra van felosztva. A mágnesréteges memóriák nagy kapacitásuk, valamint az ehhez viszonyított alacsony árak miatt a számítógépek háttérmemóriájának alapvető eszközei.

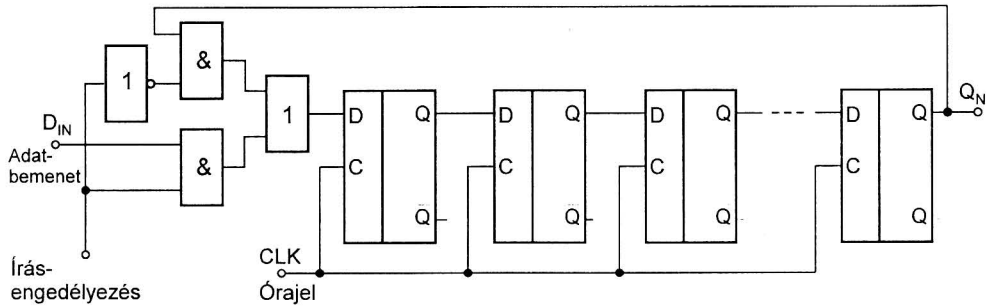
### 6.1.3. Léptetőregiszteres memóriák

A léptetőregiszter (angolul: *Shift register*) láncszerűen összekapcsolt flip-flopok együttese. Mindegyik flip-flop kimenete a következő bemenetére van kapcsolva (6.4 ábra).



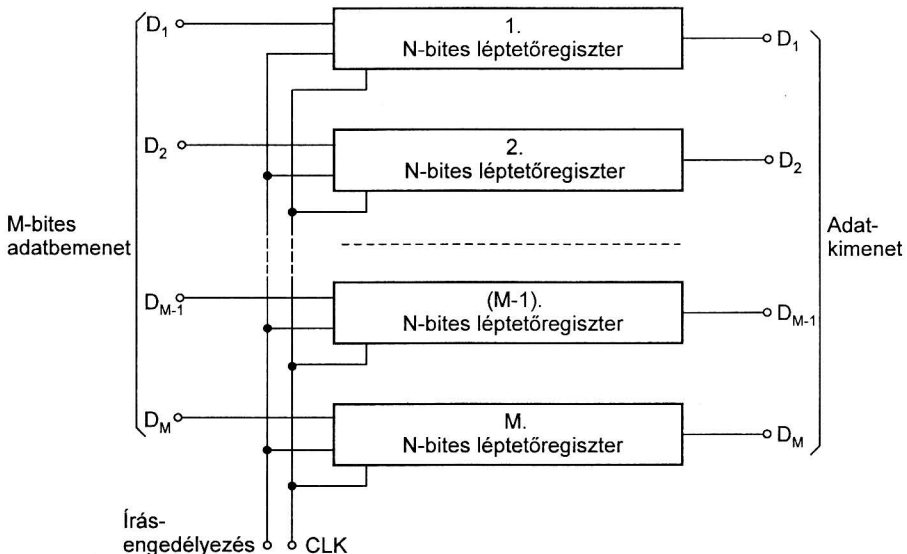
6.4 ábra. N-bites léptetőregiszter

Kivételt képez az első flip-flop, amelynek a bemenetére a tárolandó információt (adatot) vezetik. Az órajel, amely az összes flip-flopot egyidejűleg billenti, a tárolt adatot a regiszter kimeneti irányában (jobbra) "lépteti". Abban az esetben ha a léptetések során egy tárolt bit eljutott az utolsó flip-flopig, akkor egy újabb léptetéskor ez elvesztődik, vagyis "kilép" a regiszterből. Tárolásnál ez hátrányt jelent, amelyet gyűrűs léptetőregiszterrel (*Recirculating shift register*) lehet elkerülni. A gyűrűs léptetőregiszter egy olyan típusú léptetőregiszter, amelynek a kimenete vissza van csatolva a bemenetére. (6.5. ábra).



6.5. ábra. Gyűrűs léptetőregiszter

A visszacsatolást egy kapurendszer valósítja meg, de csak akkor, ha az írásengedélyező jel logikai **0**. Ellenkező esetben, vagyis ha az írásengedélyező jel **1**, akkor a kapurendszer a kimenetet leválasztja a bemenetről, hogy az utóbbira a beírandó adatokat kapcsolja. Az előbbieken ismertetett léptetőregiszterekben a tárolt adatot csak jobbra lehet léptetni. Ha egy regisztert balra is lehet léptetni, akkor a hozzáférési idő lerövidül. A jobbra és balra is léptethető regiszter flip-flopjai nem kapcsolódnak egymáshoz közvetlenül, hanem egy-egy kapurendszer biztosítja a léptetési irány megváltoztathatóságát.



6.6. ábra.  $M \times M$ -bites soros léptetőregiszteres memória vázlatos felépítése

Egy  $N$ -bites léptetőregiszterben több  $M$ -bites szó tárolható sorosan. Ha  $N = k \cdot M$ , akkor  $k$  számú  $M$ -bites szót lehet tárolni. Ebben az esetben a tárolás szervezése bitenként és szavanként is soros. Célszerűbb  $M$  darab léptetőregisztert használni.

Ezek párhuzamosan tárolhatják a szó összes  $M$  bitjét. Így  $N$ -bites léptetőregiszterekkel  $N$  számú  $M$ -bites szó tárolható (6.6 ábra). Ebben az esetben a tárolás szervezése bitenként párhuzamos és szavanként soros. A léptetőregiszterek tartalmának címzését egy számláló segítségével valósítják meg. Ez a léptetőregiszterek léptetésével párhuzamosan számlál. A számláló tartalma tudósít a regiszterek állapotáról.

A léptetőregiszteres memóriák kapacitása lemarad a mágnesréteges memóriák kapacitásától. Az utóbbi időben ezt a lemaradást kezdik behozni az ún. töltéskapcsolt eszközökkel (*CCD – Charge Coupled Devices*) megvalósított léptetőregiszteres memóriák.

#### 6.1.1.4. Mágnesbuborékos memóriák

A mágnesbuborékos memória, amint az elnevezése is mutatja, a tárolandó információt mágnesbuborékok formájában őrzi meg. A körülbelül  $3 \mu\text{m}$  átmérőjű mágnesbuborékok állandó mágneses térben elhelyezett különleges szerkezetű, filmvékonyágú mágneses anyagban véletlenszerűen mágnesezett keskeny mezőkből alakíthatók ki. A buborékok körülbelül  $15 \mu\text{m}$  távolságra vannak egymástól. Egy buborék jelenléte logikai **1**-et jelent, míg hiánya logikai **0**-t. Egy megfelelően kialakított, változtatható mágneses tér segítségével a buborékokat el lehet mozdítani a helyükből. Az adatokat képviselő mágnesbuborékok egy zárt gyűrűben éppen úgy körbe léptethetők a változtatható mágneses tér segítségével, mint a gyűrűs léptetőregiszterben tárolt adatok az órajel segítségével. Ugyancsak ezzel a mágneses térrel újabb mágnesbuborékokat lehet generálni, valamint a régebbieket megszüntetni. Ez megfelel a léptetőregiszterbe való adatbeírásnak, illetve törlésnek. Az adatkiolvasás, vagyis egy buborék jelenlétének kimutatása könnyen megvalósítható például a Hall-effektus segítségével.

A mágnesbuborékos memóriák szervezése hasonlít a gyűrűs léptetőregiszteres memóriák szervezéséhez. A mágnesbuborékos memóriáknak előnye, hogy a tápfeszültség megszűnése után sem vesztek el a tárolt információt (nemfelejtő memóriák). Hátrányuk az utóbbiakhoz képest a nagyságrendekkel hosszabb hozzáférési idő és a szükséges addicionális vezérlőlogika. A félvezető alapú memóriák gyártásának gyors technológiai fejlődése következtében a mágnesbuborékos memóriák teljesen kiszorultak a gyakorlati alkalmazásokból.

## 6.2. Tetszőleges hozzáférésű, írható olvasható memóriák (RAM)

A tetszőleges vagy véletlenszerű hozzáférésű memóriák (RAM - *Random Access Memory*) megkülönböztető jellemzője, mint ahogy az elnevezésük is mutatja, az, hogy akármelyik címhez azonos (nagyon rövid idő) alatt hozzá lehet férni. Az információ beírása vagy kiolvasása bármelyik címen tetszés szerint megismételhető. Más szóval a memória írható is és olvasható is (*Read-Write Memory*). A RAM áramkörök jellegzetessége még, hogy tápfeszültségük megszűnése esetén elveszítik információtartalmukat.

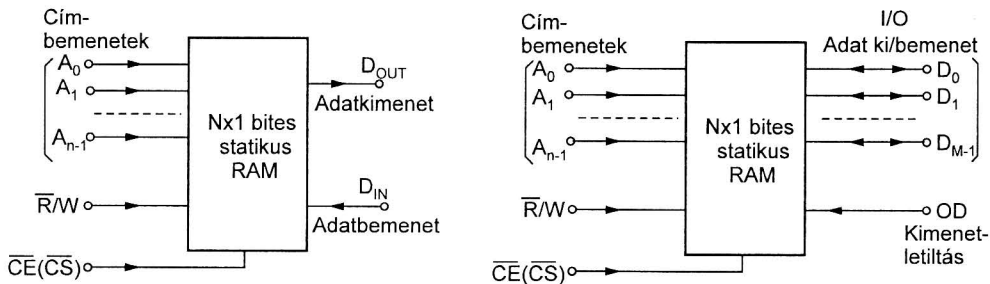
Az első RAM memóriák tároló cellája egy ferritgyűrűre épült fel. A ferritgyűrűs memória nagy hátránya, hogy a kiolvasott információ a memóriából kitorlódik. A félvezetők gyártástechnológiájának fejlődése lehetővé tette a nagy kapacitású és olcsó félvezetős RAM integrált áramkörök kifejlesztését. Előnyük az információ törlésmentes kiolvasása (a kiolvasás után a tárolt információ nem semmisül meg) és a rövid hozzáférési idő (60 nsec. nagyságrendű). Hátrányuk viszont az, hogy a tápfeszültség megszűnésekor a tárolt információt elveszítik.

*Statikus és dinamikus RAM* áramköröket különböztetünk meg. A statikus memória tároló cellája egy flip-flop, amely egy bit tárolását teszi lehetővé. A dinamikus memória tároló cellája az információt egy integrált áramkörti kapacitás által tárolt elektromos töltés formájában őrzi meg. Mivel ez a töltés a kondenzátor veszteségi ellenállásán keresztül kisül, az információ elvesztését a kondenzátor időnkénti újratöltésével, vagyis az ún. felfrissítéssel kerülik el. A dinamikus memóriák cellája kevesebb tranzisztort igényel, mint a statikusoké. Ezért a dinamikus memóriák kapacitása nagyobb, mint az azonos méretű chipen előállított statikus memóriáké. Ezért a nagy kapacitású memóriákat célszerűbb a sokkal olcsóbb dinamikus RAM áramkörökkel felépíteni. A kis kapacitású memóriák felépítése gazdaságosabb statikus RAM áramkörökkel, mert nincs szükség az információt felfrissítő ciklusokra, illetve az azokat vezérlő áramkörökre. Annak ellenére, hogy elvileg – a felfrissítés időnkénti közbeiktatása miatt – a dinamikus RAM áramköröket nem lenne célszerű gyors memóriaegységek megvalósítására alkalmazni, a jelenlegi technológiával gyártott dinamikus RAM-ok sokkal kisebb hozzáférési idővel rendelkeznek (kb. 10 nsec.) mint a statikus változatok.

### 6.2.1. Statikus RAM áramkörök

A statikus RAM áramkörök tároló cellája egy flip-flopra épül fel. Beíráskor a flip-flop átveszi a tárolandó adat által meghatározott állapotot. Kiolvasáskor a flip-flop állapota az adatkimeneten jelenik meg. Egy cella egy bit tárolását teszi lehetővé. Egy RAM áramkör kapacitását a tárolócelláinak száma határozza meg, ugyanis a memóriában tárolható bitek száma egyenlő a cellák számával.

A RAM áramkörök szervezését a megcímezhető rekeszek száma és a rekeszben tárolandó szó hosszúsága határozza meg. Egy  $N \times 1$  bit szervezésű RAM (6.7.a ábra)  $N$  különböző címen 1 bites szavakat tárolhat. Egy  $N \times M$  bit szervezésű RAM (6.7.b ábra)  $N$  különböző címen  $M$ -bites szavakat tárolhat. Általában az  $M$  értéke 4, vagy 8 szokott lenni. A memória celláinak számát az  $N \times M$  szorzat adja meg. A beírásra zánt adatok a  $D_{IN}$  adatbemenetekre kerülnek, a kiolvasott adatok pedig a  $D_{OUT}$  adatkimeneteken jelennek meg.

a)  $N \times 1$ -bites RAMb)  $N \times M$ -bites RAM

## 6.7. ábra. Statikus RAM áramkörök szervezése

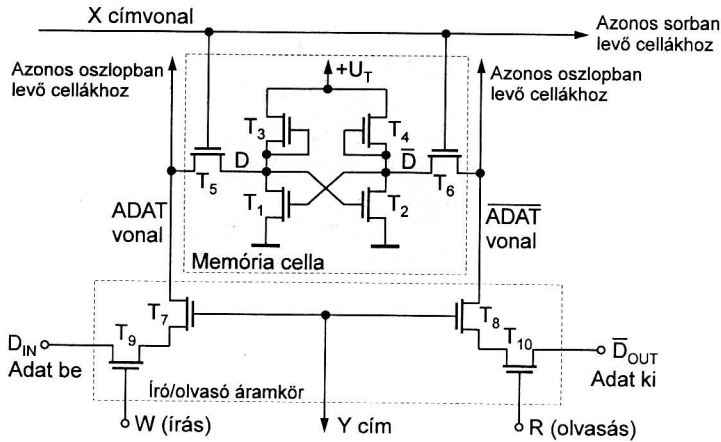
Egyes RAM típusoknál az adatbemenetek közösek az adatkimenetekkel (a 6.7.b ábra esetében megjegyzendő, hogy nemcsak az  $N \times M$  szervezésű RAM áramköröknél lehetnek közösek az adatbemenetek az adatkimenetekkel, hanem az  $N \times 1$  szervezésű RAM áramköröknél is). A rekesz  $A_{n-1}A_{n-2}\dots A_2A_1A_0$  címe egy bináris szám. A rekeszek száma:  $N$  határozza meg  $n$ -et, a cím bitjeinek számát, vagyis  $2^n = N$ .

Az adatbeírást és kiolvasást az  $R/\bar{W}$  (Read/Write) üzemmód-választó bemenet vezérli. Ha ez logikai **0**, akkor az adatbemeneten levő szó a cím által kijelölt rekeszbe íródik be. Ellenkezőleg: ha ez logikai **1**, akkor a cím által kijelölt rekesz tartalma az adatkimeneten jelenik meg. A kimenet-letiltó **OD** (Output Disable) bemenet segítségével az adatkimenet még a kiolvasás alatt is a harmadik, nagy impedanciájú állapotba hozható. A kimenet letiltását komplementummal is ki szokták fejezni. Ez az **OE** (Output Enable) - nincs kimenet engedélyezés. A memóriachip csak akkor írható és olvasható, ha az engedélyező bemenete **CE** (Chip Enable) vagy más elnevezéssel a kijelölő bemenete **CS** (Chip Select) logikai **0** szinten van. Ellenkező esetben, ha ez logikai **1**, akkor az adatkimenet a harmadik, nagy impedanciájú állapotba kerül, még abban az esetben is, ha  $R/\bar{W} = 1$  (kiolvasás alatt is). A beírás pedig még abban az esetben is gátlás alá kerül, ha  $R/\bar{W} = 0$  (beírás vezérlés alatt is). A chipengedélyező bemenet a memóriák címzését, valamint a kisebb kapacitású RAM integrált áramkörök nagyobb kapacitású memóriák megvalósítása céljából való összekapcsolását segíti elő.

A statikus RAM áramkörök az alapvető aktív építőelem típusának függvényében lehetnek bipoláris vagy MOS memóriák. A bipoláris memóriák valamivel gyorsabbak (rövidebb hozzáférési idejük); mint a MOS memóriák, viszont a MOS tárolócella sokkal kisebb helyigénye miatt egy adott felületű chipen nagyobb kapacitású memória megvalósítását teszi lehetővé.

Egy tipikus statikus MOS RAM tárolócella a 6.8. ábrán látható. A cella összesen 6 darab  $N$ -csatornás MOS tranzisztorból épül fel. A  $T_1$  és  $T_2$  keresztbekapcsolt tranzisztorok alkotják a flip-flopot.  $T_3$  és  $T_4$  növekményes tranzisztorok az aktív terhelőellenállás szerepét töltik be. A flip-flopot a  $T_5$  és  $T_6$  tranzisztorok kapcsolják az adatvonalakra. A cellákat  $P$  sorból és  $Q$  oszlopból álló mátrix alakba tömörítik. A memóriamátrix rendszerint négyzetes, vagyis  $P = Q$ . A mátrix celláit  $X$  sorkijelölő és  $Y$  oszlopkijelölő címvonalak segítségével választják ki. A megcímzett (kiválasztott) cella az  $X = 1$  és  $Y = 1$  állapotú címvonalak metszeténél található. Ha valamelyik oszlop  $Y$  címvonala logikai **1** szintre kerül, akkor a szóban forgó oszlop **ADAT** és **ADAT** vonalait kapcsoló  $T_7$  és  $T_8$  tranzisztorok vezetésbe lépnek.

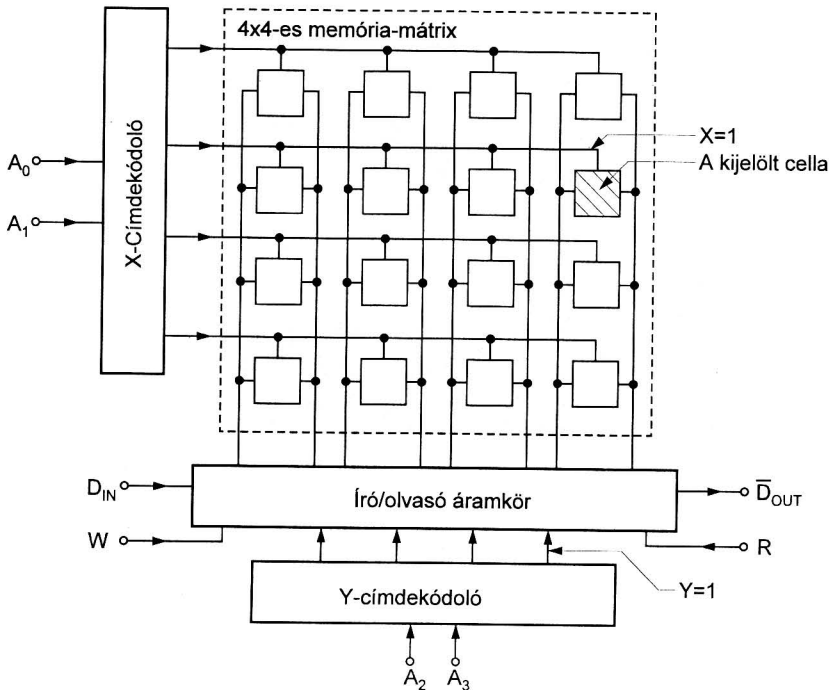




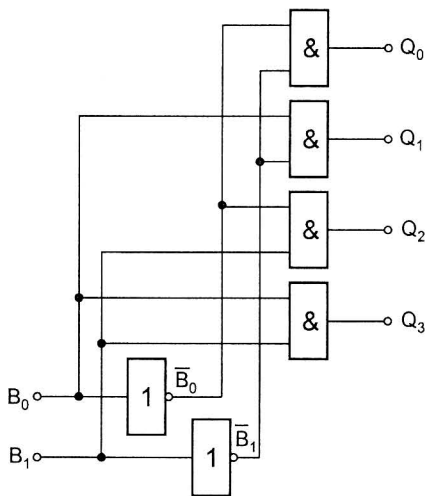
6.8. ábra. Statikus MOS RAM tárolócella

Ha valamelyik sor  $X$  címvonala logikai **1** szintre kerül, akkor a szóban forgó sorban levő cellák  $T_5$  és  $T_6$  tranzisztoraik vezetésbe lépnek. Kiolvasás esetén,  $R = 1$  a  $T_{10}$  tranzisztort hozza vezetésbe, amely a megcímezett cella  $\bar{D}$  pontját a  $\bar{D}_{OUT}$  kimenetre kapcsolja. Beírás esetén,  $W = 1$  a  $T_9$  tranzisztort hozza vezetésbe, amely a megcímezett cella  $D$  pontjára a  $D_{IN}$  bemenetet kapcsolja. Ez a bemenet a  $D$  pontot a vele azonos logikai szintre kényszeríti. A flip-flop a beírás megszűnése ( $W = 0$ ) után továbbra is megőrzi a beírt állapotot.

A 6.9. ábra egy  $16 \times 1$  bites RAM áramkör vázlatos felépítését szemlélteti.

6.9. ábra.  $16 \times 1$  bites statikus RAM vázlatos felépítése

Minden egyes cím a címdekódoló áramkörök segítségével egy-egy tároló cellát tesz hozzáférhetővé. A dekódoló a bemenetükön levő bináris címet az "1 a  $K$ -ből" (ebben az esetben  $K = 4$ ) kóddá alakítják át. Az  $X_0, X_1, X_2, \dots, X_{K-1}$   $K$  számú kimenet közül csak az egyik nem logikai  $0$ . Ha  $Z$  jelöli a bemeneti  $A_{k-1}, A_{k-2}, \dots, A_2, A_1, A_0$   $k$ -bités bináris szám által képviselt cím decimális megfelelőjét, vagyis ha  $(A_{k-1}, A_{k-2}, \dots, A_2, A_1, A_0)_2 = Z_{10}$ , akkor csak  $X_z = 1$ . Nyilvánvaló, hogy a kimenetek száma ebben az esetben:  $K = 2^k$ . A  $4 \times 4$ -es memória mátrix cellái két "1 a 4-ből" dekódoló segítségével címezhetők meg. Egy ilyen dekódoló felépítését, valamint igazságtáblázatát a 6.10. ábra szemlélteti.



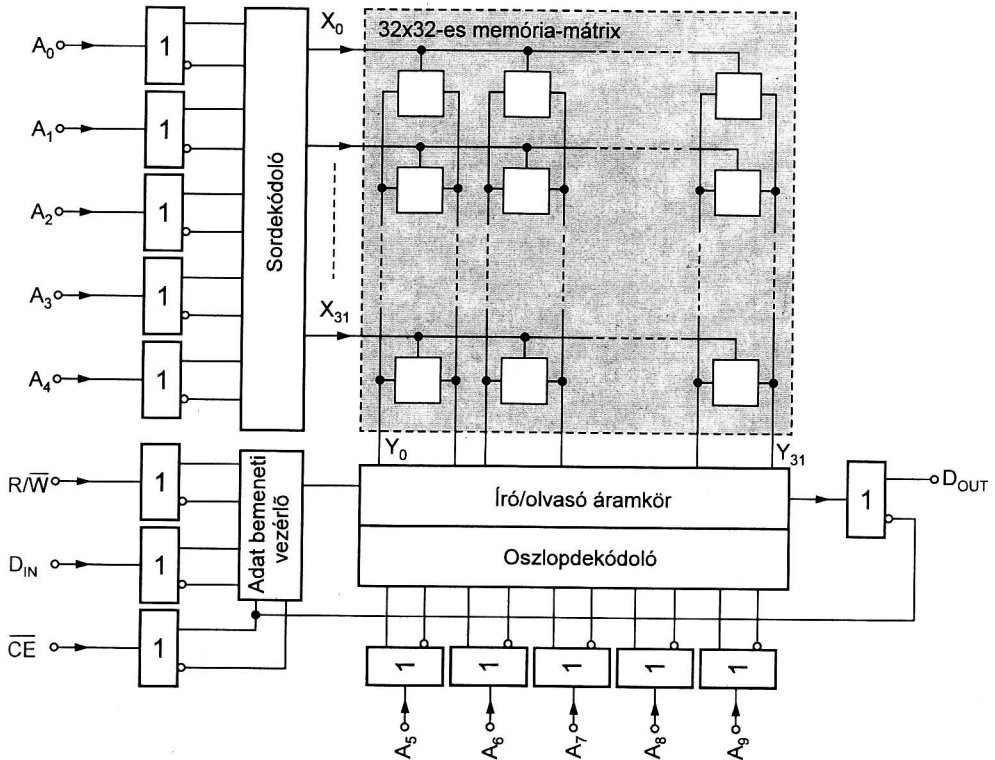
Bemeneti változók		Kimenetek			
$B_0$	$B_1$	$Q_0$	$Q_1$	$Q_2$	$Q_3$
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

a)

b)

6.10. ábra. Az „1 a 4-ből” dekódoló (a) és igazságtáblázata (b)

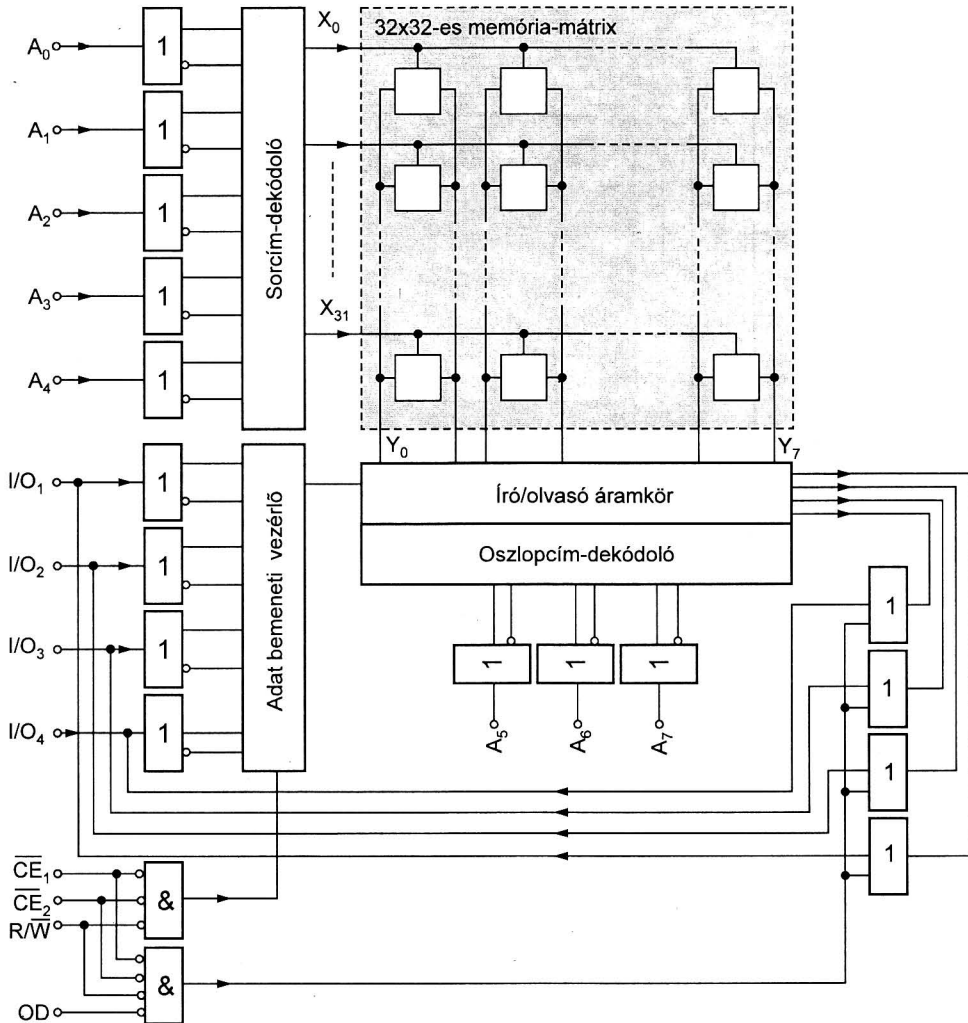
Elvileg nem sokban különbözik egy nagyobb kapacitású memória felépítése az előbbieken bemutatott kisebb kapacitású memória felépítésétől. Példaként a 6.11. ábrán egy  $1024 \times 1$  bites RAM integrált áramkör vázlatos felépítése látható. A sordekódolók, az oszlopdekódolók bemenetein, valamint az  $R/\overline{W}$ ,  $D_{IN}$  és  $\overline{CE}$  bemeneteken levő két kimenetű kapuk egyidejűleg a megfelelő bemeneti változót és a komplementjét is szolgáltatják. Az író/olvasó áramkör kimenete egy háromállapotú kapun keresztül csatolódik a  $D_{OUT}$  adatkimenetre. Ha  $\overline{CE} = 1$ , akkor függetlenül az  $R/\overline{W}$  bemenet állapotától, a beírás gátlás alá, míg a  $D_{OUT}$  kimenet a harmadik, nagy impedanciájú állapotba kerül. Ha  $\overline{CE} = 0$ , akkor a memória írható és olvasható is. Így, ha  $R/\overline{W} = 0$ , akkor a megcímezett cellába beíródik a  $D_{IN}$  adatbemeneten levő bit; és ha  $R/\overline{W} = 1$ , akkor a megcímezett cellában tárolt bit megjelenik a  $D_{OUT}$  adatkimeneten.



6.11. ábra. Egy 1024×1 bites RAM tömbvázlata (Intel 2102)

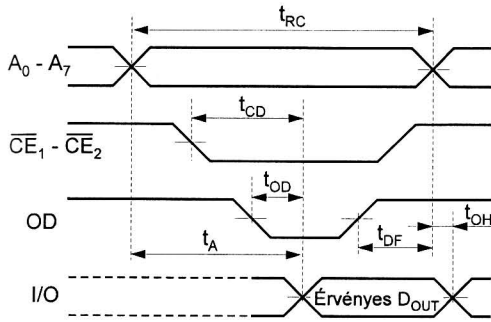
A 6.12. ábra, példaként egy 256×4 bites RAM áramkör segítségével, az  $N \times M$  bit szervezésű RAM áramkörök felépítését szemlélteti. Mivel egy tárolócella csak egy bitet tárolhat, az  $M$  bites szó tárolására  $M$  tárolócellát kell egyidejűleg hozzáférhetővé tenni. Ezek rendszerint egy sorban helyezkednek el. Tehát az oszlopdekódoló egyidejűleg  $M$  oszlopot jelöl ki. Vagyis,  $Q' = \frac{Q}{M}$  számú oszlop-csoport alakul ki. Így az oszlopdekódoló „1 a  $Q'$ -ből” dekódoló, és egyidejűleg  $M$  oszlopból álló oszlopcsoportot jelöl ki. Az oszlopcsoportok és a sorok számának szorzata a memória rekeszeinek számát adja, vagyis  $P \cdot Q' = N$ . A 6.12. ábrán bemutatott memória esetében mivel a szóhosszúság  $M = 4$ , és  $Q = 32$  oszlopból összesen  $Q' = 8$  oszlop-csoport alakul ki. Az áramkör adatbemeneti közösek az adatkimeneteivel ( $I/O_1, \dots, I/O_4$ ). A chip csak akkor engedélyezett, vagyis csak akkor írható és olvasható, ha  $\overline{CE}_1 \cdot \overline{CE}_2 = 0$ . Kiolvasás esetén, amikor  $R/\overline{W} = 1$ , az adatkimeneti háromállapotú kapuk csak akkor kapcsolják a közös adatvonalakra a megcímezett rekeszből kiolvasott 4-bites szót; ha a kimenet nincs letiltva ( $OD = 0$ ).

Beírás esetén, amikor  $R/\overline{W} = 0$ , az adatkimeneti háromállapotú kapuk a harmadik, nagyimpedanciájú állapotba kerülnek.

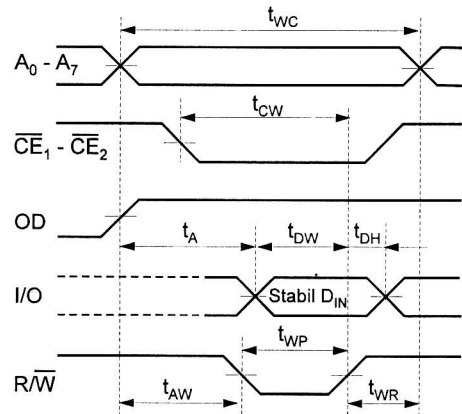


6.12. ábra. Egy 256x4 bites RAM tömbvázlata (Intel 2111A)

A 6.13. ábra a beírási és kiolvasási művelet idődiagramját szemlélteti. Ez a hardvertervezésnél elengedhetetlen fontosságú. Az egyszerűség kedvéért az idő-, valamint a feszültség-koordinátatengelyek ábrázolását elhanyagoltuk. Az alacsonyabb feszültségszint a „0” logikai szintet, míg a nagyobb feszültségszint az „1” logikai szintet képviseli (pozitív logika). A harmadik állapotot szaggatott vonal ábrázolja. A két logikai szint fele távolságán futó folytonos vonallal is szokták ábrázolni. Az átkapcsolási időket nem lehet nullával egyenlőnek venni, mint egyes ideális esetekben. Az átkapcsolási görbéket egyenessel közelítettük meg. A jelalakok funkcionális fontosságú egymáshoz való viszonyulását megjelöltük. Értékük a memória adatlapjaiban megtalálható.



a) jellegzetes olvasó ciklus



b) jellegzetes író ciklus

**6.13. ábra.** Egy statikus RAM (Intel 2102) jellegzetes idődiagramjai

$t_{RC}$  – adat kiolvasási ciklus

$t_A$  – hozzáférési idő

$t_{CO}$  – chip-engedélyezéstől az adatkimenetig

$t_{OD}$  – kimenettiltástól az adatkimenet harmadik állapotáig

$t_{OH}$  – előzőleg kiolvasott adat a címváltozás utáni fennmaradása

$t_{WC}$  – az adatbeírási ciklus

$t_{AW}$  – íráseltolás

$t_{CW}$  – chip-engedélyezéstől a beírásig

$t_{DW}$  – adat előzetes beállítása

$t_{DH}$  – adat fenntartása

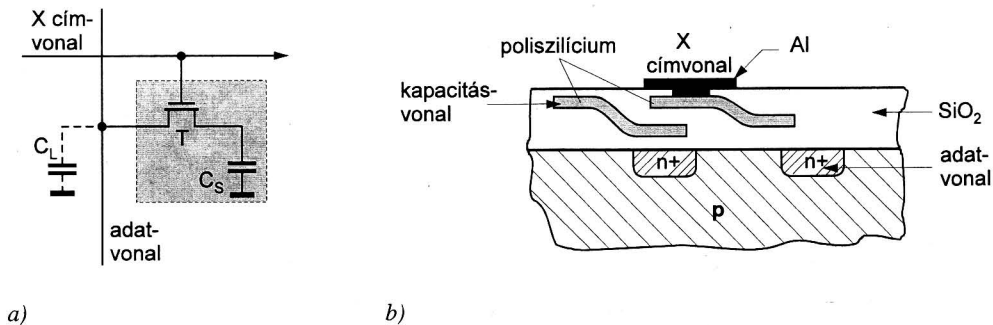
$t_{WP}$  – beíróimpulzus

$t_{WR}$  – beírás feléledési ideje

$t_{DS}$  – kimenettiltás előzetes beállítása

## 6.2.2 Dinamikus RAM áramkörök

A dinamikus memóriák jellegzetessége a nagy kapacitás és ehhez viszonyítva az alacsony ár. A dinamikus memória tárolócellája lehet háromtranzisztoros vagy az újabb nagy kapacitású memóriákban egytranzisztoros. A dinamikus memóriák  $N$ -csatornás MOS technológiával készülnek.

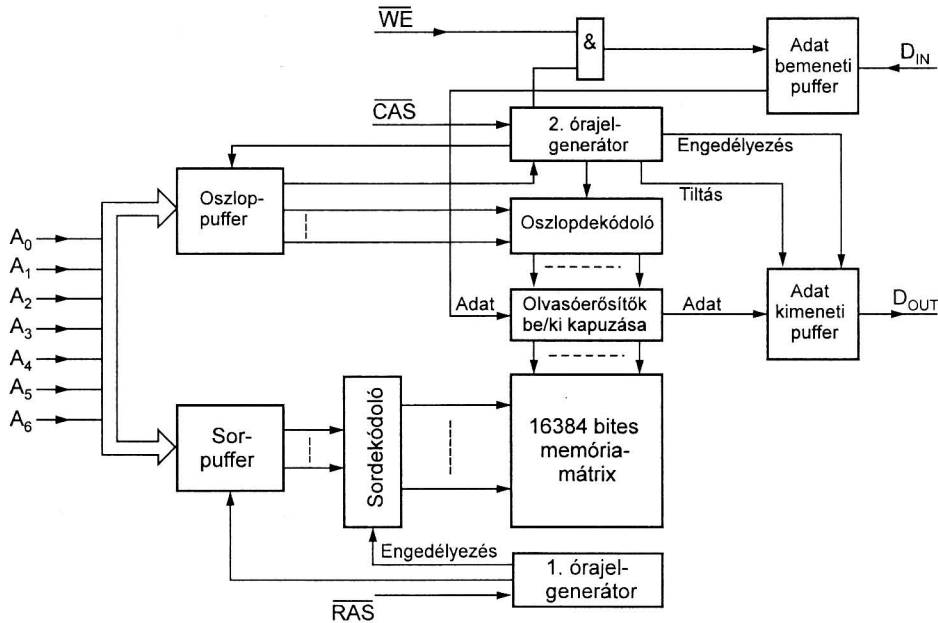


6.14. ábra. Egytranzisztoros dinamikus RAM tárolócella (a) és integrált áramköri metszete (b)

Egy egytranzisztoros tárolócellát mutat be a 6.14. ábra. Az információátvitel szerepét a  $C_S$  kondenzátor látja el. Ezt az  $X$  címvonallal vezérelt  $T$  tranzisztor kapcsolja az oszlop adatvonalára, amelynek a végére a kiolvasó erősítő kapcsolódik. A feltöltött  $C_S$  kondenzátor az erősítő bemenetén a  $\frac{C_S}{C_L}$  törttel arányos jelet szolgáltat, ahol  $C_L$  az adatvonal kapacitása,

és arányos a memória-mátrix felületével. A nagy kapacitású memóriák esetén  $C_L$  nagyobb, mint  $C_S$ , és ezért az olvasó-erősítő bemenetén csak 10.....100 mV nagyságú jel jelenik meg. Ennek a kis amplitúdójú jelnek az erősítését differenciálerősítővel végzik. Abban az esetben, ha a kapacitásvonal pozitív feszültséget kap, ennek hatására a  $P$ -szubsztrátban  $N^+$  inverziósvonal alakul ki. Ez tárolja az információt, és egyben a tranzisztor source-át is képviseli. A címvonallra kapcsolt jel hatására a source és drain között vezetősatorna alakul ki. A drain egyben adatvonal is. A bináris információ másik állapotában a kapacitásvonal nem kap pozitív feszültséget, és ezért nem alakul ki vezetősatorna.

A  $C_S$  tárolókondenzátor idővel veszít a töltéséből. A tárolt információ elvesztését a kondenzátor újratöltésével akadályozzák meg. Ezt a folyamatot felfrissítésnek nevezik, és periodikusan (rendszerint 2 ms-ként) meg kell ismétlni. A felfrissítés tulajdonképpen a tárolt információ kiolvasása és az azonos cellákba való újrafírása. Ezt a folyamatot a dinamikus memória felfrissítő erősítői könnyítik meg. A memóriamátrix minden egyes oszlopa egy-egy felfrissítő erősítővel van ellátva. A memóriamátrix egy sorának a megcímzésével az egész sor automatikusan felfrissül. Az egész memória felfrissítése az összes  $P$  sor felfrissítéséből áll. Míg a statikus RAM áramkörök felépítése típusonként kevésbé eltérő, a dinamikus RAM áramkörök felépítése a felfrissítési művelet különböző kivitelezése miatt sokkal eltérőbb.



6.15. ábra. Egy 16 kbites dinamikus RAM tömbvázlata (Intel 2116)

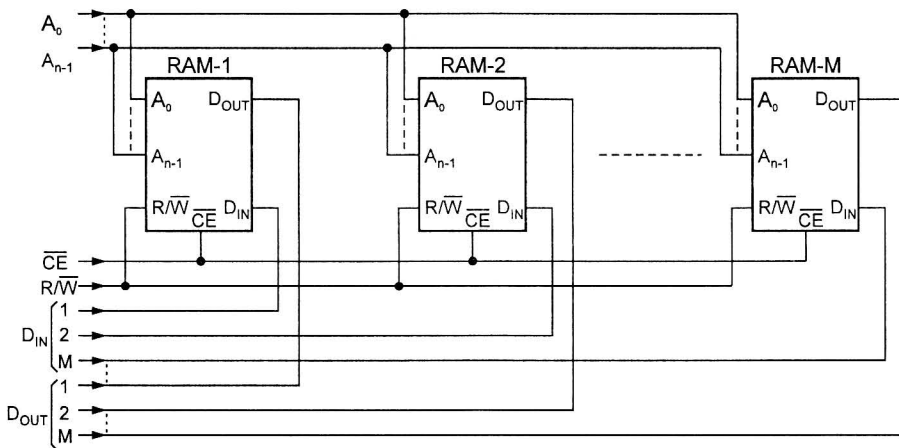
A 6.15. ábra egy 16 kbites ( $16384 \times 1 \text{ bit}$ ) dinamikus RAM tömbvázlatát szemlélteti (jellemző a dinamikus RAM áramkörök elég nagy hányadára). A memóriamátrix tárolócelláinak a megcímezésére 1 k bit szükséges ( $2^{14} = 16384$ ). Ezeket a memória az összesen hét ( $A_0$ -tól  $A_6$ -ig) címbemeneten két részletben kapja meg. Az egész 14 bites címet a sor- és az oszlop-puffer tárolja. A puffer (angolul: *– buffer, storage latch*) tulajdonképpen egy egyszerű  $1 \times M$  bit (ebben az esetben  $M = 7$ ) szervezésű statikus RAM. Ez a szóhosszúságnak megfelelően  $M$  számú flip-flopból épül fel, melyeknek közös órajele beírja a bemenetükre kapcsolt szót. A következő órajelig a flip-flopok kimeneteiről ez a szó olvasható le. A sor- és oszlop-puffer órajeleit az 1. illetve a 2. órajel-generátor szolgáltatja. A  $\overline{RAS}$  (angolul: *Row Address Strobe*) külső órajel vezérli az 1. Órajel-generátort, amely az általa képzett órajelekkel egyrészt beírja a cím első hét bitjét ( $A_0$ -tól  $A_6$ -ig) a sorpufferbe, másrészt vezérli a sordekódolót a megfelelő sor kijelölése érdekében.

$\overline{CAS}$  (angolul: *Column Address Strobe*), a másik külső órajel vezérli a 2. órajel-generátort. Ez az általa képzett órajelek segítségével beírja a cím következő hét bitjét ( $A_7$ -től  $A_{14}$ -ig) az oszloppufferbe, vezérli az oszlopdekódolót (hogyan a kijelölt sor és oszlop metszeténél levő tárolócella váljon hozzáférhetővé a kiolvasás, a beírás és felfrissítés számára), és végül vezérli az adatkimeneti, valamint az adatbemeneti puffereket is. A beírást vagy a kiolvasást a  $\overline{WE}$  (*Write Enable*) beírásengedélyező bemenet vezérli. A kiolvasást  $\overline{WE} = 1$  határozza meg, míg a beírást  $\overline{WE} = 0$ . A  $\overline{CAS}$  és  $\overline{RAS}$  órajelek megfelelő együttese egy megcímezett sor felfrissítését teszi lehetővé. Az egész memóriamátrix felfrissítése az összes 128 sor felfrissítéséből áll, amelyet legkritikábban minden 2 ms-ként kell elvégezni. A dinamikus RAM különböző vezérlőjelei között levő bonyolult időbeni összefüggést, amelynek a betartása elengedhetetlen a hardvertervezésnél, az adatlapok tanulmányozásával lehet részletesen megismerni.

A dinamikus memóriák egyedüli hátránya, hogy időnként a tartalmukat fel kell frissíteni. A felfrissítés alatt a memória nem hozzáférhető. A felfrissítés interferálhat a számítógép író és olvasó ciklusaival, ekkor a számítógépnek várnia kell. Ezt a nemkívánatos időkiesést úgy kerüljük el, hogy a felfrissítést azokban az időintervallumokban hajtják végre, amelyekben a memória nincs sem kiolvasás alatt, sem beírás alatt. Ez természetesen komplexebb felfrissítésvezérlő logikai áramkört igényel, mint egy szabályos időközönként végrehajtott felfrissítés.

### 6.2.3. Nagy kapacitású RAM kialakítása kisebb kapacitású RAM integrált áramkörökből

Az integrált áramköri gyártástechnológia még 16 Mbit statikus és 64 Mbit dinamikus RAM integrált áramkörök megvalósítását is lehetővé teszi. A nagy teljesítményű mikroszámítógép RAM memóriaegységei kapacitásának még ennél is nagyobbak kell lennie. A kapacitás növelését egyrészt a memória által tárolható szóhosszúság növelésével, másrészt a megcímezhető rekeszek számának növelésével kell megvalósítani. A két módszert az alábbiakban ismertetjük. Ezeket a legtöbb esetben kombináltan alkalmazzák. A 6.16. ábra a tárolható szóhosszúság növelését szemlélteti. Legyen a felhasznált RAM áramkör  $N \times 1$  bit szervezésű. Ha a szó hosszúsága  $M$  bit, akkor  $M$  számú RAM áramkör szükséges az  $N \times M$  bit szervezésű RAM egység megvalósításához.

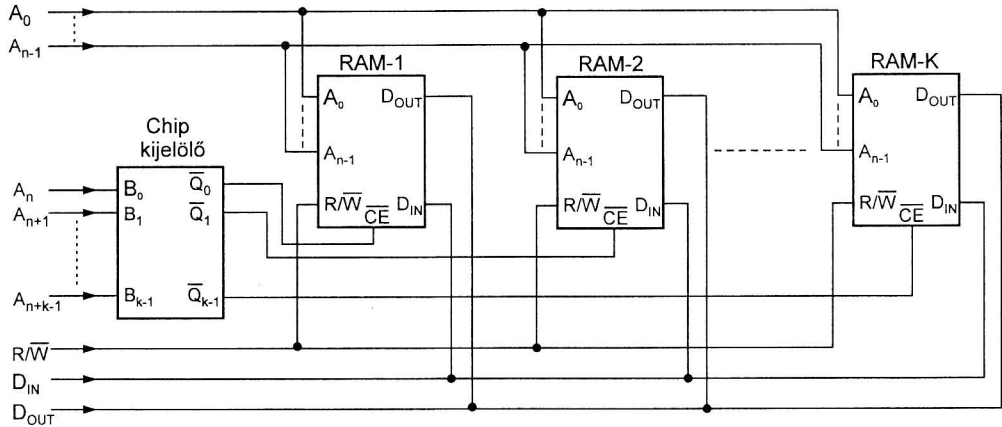


6.16. ábra.  $M$  számú  $N \times 1$ -bites RAM integrált áramkörből megvalósított  $N \times M$ -bit szervezésű RAM memóriaegység

A címbemenetek és a chip engedélyező bemenetek párhuzamos kapcsolása folytán a címek bármelyikén az első RAM áramkör tárolja a szó első bitjét, a második RAM áramkör a szó második bitjét, és így tovább az  $M$ . RAM áramkörig. Például egy  $1024 \times 8$  bit szervezésű RAM egységet nyolc  $1024 \times 1$  bites RAM áramkörből vagy két  $1024 \times 4$  bites RAM áramkörből lehet kialakítani.



A 6.17. ábra szemlélteti a kapacitás növelését egy adott szóhosszúság mellett. Az alapáramkör ugyancsak egy  $N \times 1$  bit szervezésű RAM. Az  $N$  rekesz megcímezését megvalósító  $A_0, A_1, \dots, A_{n-2}$  és  $A_{n-1}$  címbemenetek párhuzamosan kapcsolódnak.



6.17. ábra.  $K$  számú  $N \times 1$  bites RAM integrált áramkörből megvalósított  $KN \times 1$ -bit szervezésű RAM memóriaegység

Cím				Chip- engedélyezés			
$A_3$	$A_2$	$A_1$	$A_0$	$CE_1$	$CE_2$	$CE_3$	$CE_4$
0	0	0	0	0	1	1	1
0	0	0	1	0	1	1	1
0	0	1	0	0	1	1	1
0	0	1	1	0	1	1	1
0	1	0	0	1	0	1	1
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1
0	1	1	1	1	0	1	1
1	0	0	0	1	1	0	1
1	0	0	1	1	1	0	1
1	0	1	0	1	1	0	1
1	0	1	1	1	1	0	1
1	1	0	0	1	1	1	0
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	0

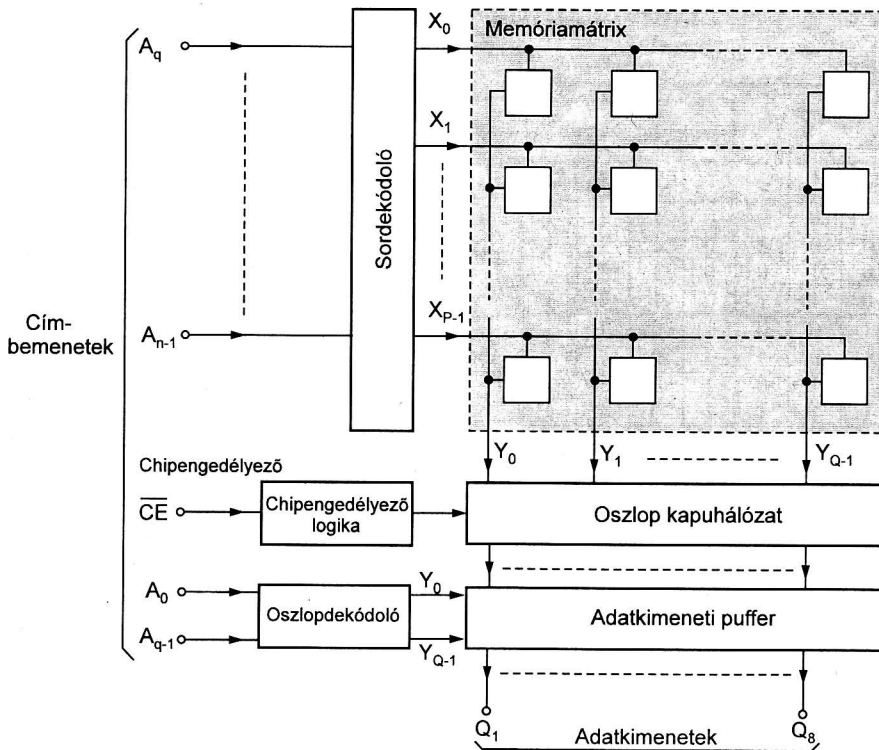
6.1. táblázat. A cím- és chipengedélyezés nagyobb kapacitású memória kisebb kapacitású memóriákból való felépítésénél egy feltételezett esetben

A chip-engedélyező bemenetek a chip-kijelölő áramkör  $K$  kimeneteire kapcsolódnak. A chip-kijelölő egy „1 a  $K$ -ból” dekódoló, amely a soron következő  $A_n, A_{n+1}, \dots, A_{n+(k-1)}$   $k$  címvonalat dekódolja, és mindig csak egy RAM áramkört jelöl ki.

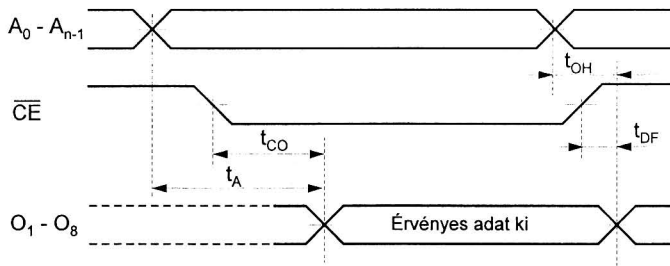
Nyilvánvaló, hogy ebben az esetben  $K = 2^k$ . Ha a címek növekvő sorrendjét vesszük, akkor az első  $N$  cím, csak az első RAM áramkört címezi meg, a második  $N$  cím csak a második RAM áramkört, és így tovább az utolsó  $N$  címig. Példaként a 6.1. táblázat szemlélteti a cím és a chip-engedélyezés alakulását egy egyszerű feltételezett esetben. A négy darab  $4 \times 1$  bites RAM áramkör felhasználásával  $16 \times 1$  bites RAM egység valósítható meg. Az első két címvonal,  $A_0$  és  $A_1$ , a RAM áramkörök 4 rekeszét címezik meg. A további két címvonal,  $A_2$  és  $A_3$  az „1 a 4-ből” chip-kijelölő dekódoló segítségével engedélyezi a négy RAM áramkör közül az egyiket.

### 6.3. Csak olvasható memóriák (ROM)

A csak olvasható memóriákat (**ROM – Read Only Memory**) fix memóriáknak is nevezik. A bennük rögzített információt csak kiolvasni lehet. A kiolvasás szempontjából a ROM memóriák véletlenszerű hozzáférések. Az információ rögzítése történhet a memória gyártási folyamata alatt vagy felhasználása előtt. A tápfeszültség kikapcsolásával a rögzített információ nemvész el. A ROM áramkörök tömbvázlata (6.18. ábra) sokban hasonlít a RAM áramkörök tömbvázlatához. A ROM esetében is az áramkör alapvető részét a memóriamátrix képezi. A ROM tárolócella viszont sokkal kisebb felületigényű, mint a RAM tárolócella. Ezért azonos felületű kristályra sokkal nagyobb kapacitású ROM valósítható meg, mint RAM. A sordekódoló a memóriamátrix  $P$  számú sorából csak egyet jelöl ki. Az oszlopdekódoló az oszlop-kapurendszer segítségével a kijelölt sorból  $M$  cella tartalmát írja be az adatkimeneti pufferbe. A szóhosszúság általában  $M = 1, 2, 4$  vagy  $8$  bit lehet (a 6.18. ábra esetében  $M = 8$  bit). Ha  $Q$  az oszlopok száma és  $Q'$  az oszlopdekódoló kimeneteinek száma, akkor nyilvánvaló, hogy  $Q' = \frac{Q}{M}$ . Az alatt, amíg  $\overline{CE}$  logikai 1-et kap, addig az adatkimenet a harmadik, nagy impedanciájú állapotban található. Ha  $\overline{CE}$  logikai 0-t kap, akkor az adatkimenetről a megcímzett rekesz tartalmát lehet leolvasni. A 6.19. ábra egy ROM jellegzetes kiolvasási idődiagramját szemlélteti.



6.18. ábra. Csak olvasható memória (ROM) tömbvázlata



6.19. ábra. Egy ROM jellegzetes olvasóciklusa

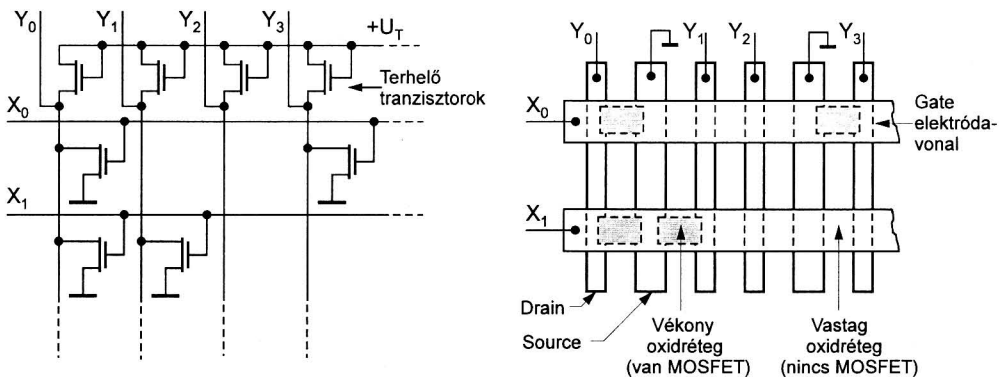
A ROM memóriákat főként a mikroszámítógépes célrendszerek fix programjának tárolására használják. Egyéb tipikus ROM alkalmazások: a különböző matematikai függvények előállítására szolgáló függvénytáblázatok, a különböző szabványos kódokhoz használatos kódgenerátorok, valamint a megjelenítőkhöz szükséges alfanumerikus karaktereket előállító karaktergenerátorok.

A ROM memóriák típusai, amelyek az alábbiakban kerülnek részletesebb bemutatásra, az információ rögzítési módjára utalnak.

### 6.3.1. Maszkprogramozott ROM áramkörök

A maszkprogramozott ROM áramkörökbe az információt a gyártás során programozzák be, és ez utólag már nem változtatható. A programozást a gyártástechnológiai folyamatban felhasznált maszkok egyikével valósítják meg. Innen származik a „maszkprogramozott” elnevezésük is. Nagy felhasználási sorozatok esetén a maszkprogramozott ROM alkalmazása a leggazdaságosabb.

A 6.20. ábra egy tipikus MOS ROM memóriamátrix szerkezetét, valamint fizikai elrendezését szemlélteti felülnézetben.



a) áramköri részlet

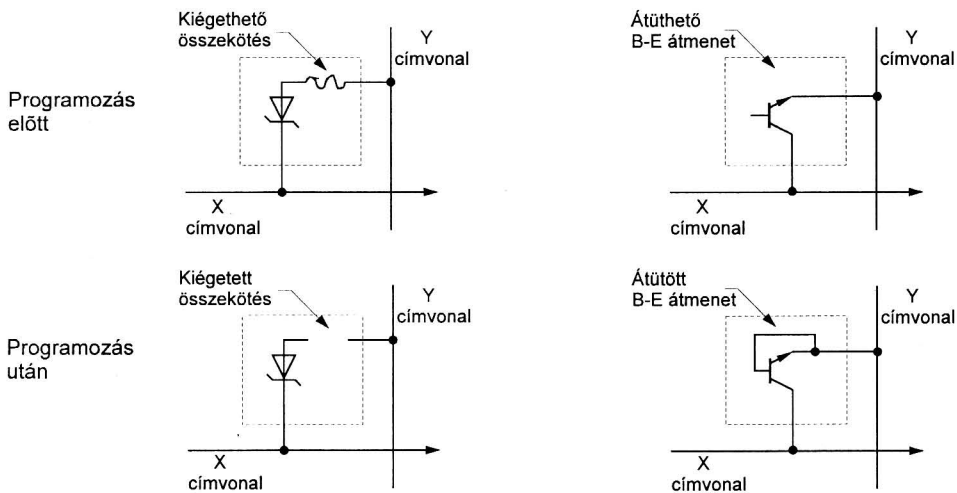
b) integrált áramköri kivétel

6.20. ábra. MOS ROM memóriamátrix egy részlete

Egy oszlopban levő tranzisztorok drainjei közösek, egyben az  $Y$  címvonalat a képezik. Két szomszédos oszlopban levő tranzisztor source-a is közösen van kialakítva. Az  $X$  címvonalat az egy sorban levő tranzisztorok közös gate elektrodája képezi. Ha a cella csatornafelülete fölött lévő oxidréteg vastag, akkor a tranzisztor  $U_{TO}$  küszöb feszültsége meghaladja a kijelölt  $X$  címvonatra kapcsolt feszültséget. Mivel a csatorna nem jöhet létre, ez a hely tranzisztorhiánynak felel meg. Vékony oxidréteg esetén a tranzisztor  $U_{TO}$  küszöb feszültsége a kijelölt  $X$  címvonat feszültség szintjénél kisebb. Így a kijelölt  $X$  címvonatra kapcsolt tranzisztorok vezetésbe jönnek. A többi, nem kijelölt  $X$  címvonatokra kapcsolt tranzisztor lezárt állapotban marad. A vezetésben levő tranzisztorok drainjei – tehát az ezeknek megfelelő  $Y$  címvonatok is – közel földpotenciálon vannak. A terhelőtranzisztorok a többi címvonat feszültségét közel  $+U_T$  értéken tartják.

### 6.3.2. Felhasználáskor programozható ROM áramkörök (PROM)

Ez a típusú ROM rendeltetésének megfelelően a felhasználás előtt programozható. Rövidített jelölése **PROM** (Programmable ROM). Kis felhasználási sorozatok esetén alkalmazható előnyösen.



a) „kiégethető” cella

b) „átüthető” cella

6.21. ábra. PROM tárolócellák

A PROM áramkörök majdnem kizárólag csak bipoláris technológiával készülnek. Programozásuk kétféleképpen történik. Az egyik típusnak – amelynek tárolócellája egyetlen diódából áll – a programozása az ebből a célból kialakított kiegészítő összekötések megszakításával történik (6.21.a. ábra.). A diódák anódja egy vékony Ni–Cr, Ti–W vagy polikristályos szilícium rétegből készülő ellenállászakasszal kapcsolódik az  $Y$  címvonatra. Ha az ezen átfolyó áramsűrűség értéke meghalad egy kritikus szintet, akkor az anyag hirtelen megolvad, és szakadás áll be. Az újabb áramkörök általában Schottky-diódákkal készülnek, amelyeknek előnye a kis hozzáférési idő.

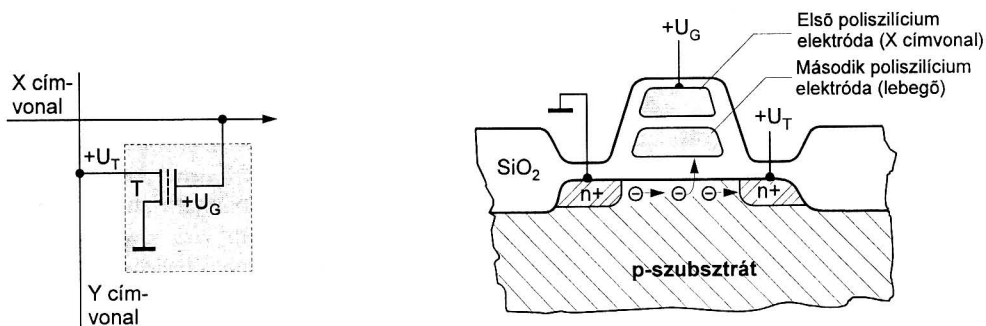
A másik típusnak a cellája egyetlen tranzisztorból áll, amelynek a bázisa nincs bekötve (6.21.b. ábra). A programozás előtt az emitter és kollektor között nem folyhat áram. Programozáskor a cellára kisülő feszültség meghaladja a bázis-emitter átmenet letörési szintjét. Az átmeneten nagy áram jön létre, melynek hatására tartós rövidzár keletkezik az emittervezeték és a bázis között. Így az  $Y$  címvonala kapcsolt bázis a bázis-kollektor átmenetet képviselő dióda anódja.

Minden egyes PROM áramkörtípus programozási eljárását az adatlapjai részletesen közlik. Az adatok ún. „beégetése” a külön ezt a célt szolgáló automatikus programozókészülékkel történik.

### 6.3.3. Újraprogramozható ROM áramkörök

Az újraprogramozható ROM egyedi berendezésekben, prototípusokban kerül alkalmazásra, és ott, ahol a programot meg kell tudni változtatni. Rövidített jelölése EPROM vagy REEPROM (Erasable, illetve Re-programable **PROM**).

Az adatok beégetése a megfelelő programozókészülékkel felhasználás előtt végezhető el. A memória tartalma egészben, egyszerre törölhető. Az egyik legelterjedtebb EPROM típus az, amelynél a törlés ultraibolya (UV) sugárral történik (angolul: UV Erasable PROM). Egy másik, újabb típusnál a törlés is, mint a programozás, elektromosan végezhető. Ezeknek a rövidített jelölése EEPROM (Electrically Erasable **PROM**) vagy EAROM (Electrically Alterable **ROM**).



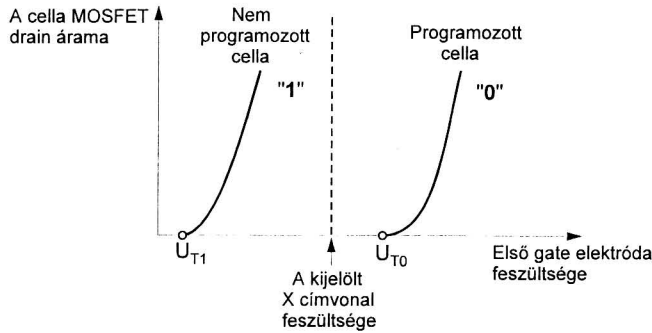
a) tárolócella - FAMOS cella

b) integrált áramköri keresztmetszete

6.22. ábra. Ultraibolya sugárral törölhető és újraprogramozható ROM (UV EPROM)

A 6.22. ábra egy ultraibolya sugárral törölhető EPROM tárolócelláját mutatja be. A cella egyetlen lebegő vezérlőelektródájú, lavinainjektálással működő MOS (FAMOS - Floating-gate Avalanche-injection **MOS**) tranzisztorból áll. A polikristályos szilíciumból levő lebegő elektróda a jól szigetelő oxidrétegben minden oldalról körülvéve „lebeg”, vagyis nem kapcsolódik az áramkör többi részéhez. A tranzisztor tulajdonképpeni vezérlőelektródája a felső,  $X$  címvonala kapcsolt elektróda. A programozás a lebegő elektróda lavinainjektálásával történik. A nagy térerősség hatására a csatornában mozgó elektronok lavinászerűen átlépik a szilícium-szubsztrát és az oxidréteg-átmenet energiaküszöbét, és eljutnak a lebegő elektródához.

A kiváló szigetelés miatt a lebegő elektródára került töltés nagyon hosszú ideig megmarad (70°C-on 100 év alatt kb. 5%-os töltésvesztés). A lebegő elektródán levő negatív töltés hatására a tranzisztor küszöbfeszültsége megnő (6.23. ábra).



6.23. ábra. A FAMOS cella tranzisztorának átviteli jelleggörbéje

A nem programozott cella tranzisztorának  $U_{T1}$  küszöbfeszültsége a kijelölt X címvonal feszültségénél kisebb, de a nem kijelölt X címvonalak feszültségénél nagyobb. Ezért csak azok a nem programozott tranzisztorok vezetnek, amelyek a kijelölt X címvonalon vannak. A programozott cella tranzisztorának  $U_{T0}$  küszöbfeszültsége meghaladja a kijelölt X címvonal feszültség szintjét. Ezért ez a tranzisztor akkor sem vezet, ha kijelölték.

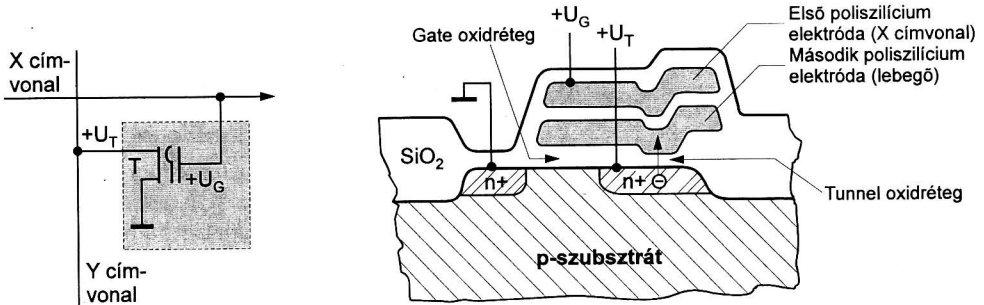
Tehát, ha a kijelölt cella tartalma "1" (a lebegő elektródán nincs töltés), akkor a tranzisztor vezet; és ha a kijelölt cella tartalma "0" (a lebegő elektróda fel van töltve), akkor a tranzisztor nem vezet.

A törlés a lebegő elektródán tárolt töltés eltávolításából áll. Ez fotoelektromos hatás segítségével valósul meg. Az áramkört közvetlen ultraibolya ( $\lambda = 0,2537 \mu\text{m}$  hullámhosszú) sugárzásnak teszik ki. Ez a chiphez a tok elején levő kvarcablakon keresztül jut el. A sugárzás hatására a lebegő elektródán tárolt elektronok energiája annyira megnövekszik, hogy el tudják hagyni az elektródát. Törlés után a memória összes tárolócelláinak tartalma "1". Programozással a kívánt tárolócellák tartalma "0"-ra változtatható.

Az ultraibolya sugaras törlés egy kissé körülményes. Először is egy megfelelő erősségű (kb.  $10 \text{ Ws/cm}^2$ ) ultraibolya fényforrás szükséges. Másodsorban a memória kitörlése elég sok időt vesz igénybe (átlagosan 15-30 percet).

A 6.24. ábra egy elektromosan törölhető EPROM tárolócellát mutat be. Az elektromos törlés az ultraibolya sugaras törlés hátrányait küszöböli ki. A cella egyetlen MOS tranzisztorból áll, amelynek a törlése alagúthatáson alapul: Ez az ún. FLOTOX (Floating-gate Tunnel-Oxide) cella. A FLOTOX cella programozása – hasonlóan mint a FAMOS cella programozása –, a lebegő elektróda lavinainjektálásával történik. A FLOTOX cella törlése az ún. Fowler-Nordheim-féle tunneletfektus segítségével történik. Ha két elektróda között levő szigetelőben (ebben az esetben  $\text{SiO}_2$ ) az elektromos térerő meghalad egy kritikus szintet (kb.  $10^7 \text{ V/cm}$ ), akkor az alagúthatás következtében a negatív elektródán levő elektronok a szigetelőn keresztül eljutnak a pozitív elektródáig.

Ebben az esetben, ha az  $X$  címvonallra kapcsolt vezérlő elektróda földpotenciált kap, és a drain nagy pozitív potenciált (kb.  $+20$  V-ot), akkor az elektronok a negatív töltésű elektródáról a vékony oxidrétegen keresztül a Fowler-Nordheim-féle effektus alapján eljutnak a drainig. Így a lebegő elektróda elveszti a negatív töltését – a cella kitörlődik.

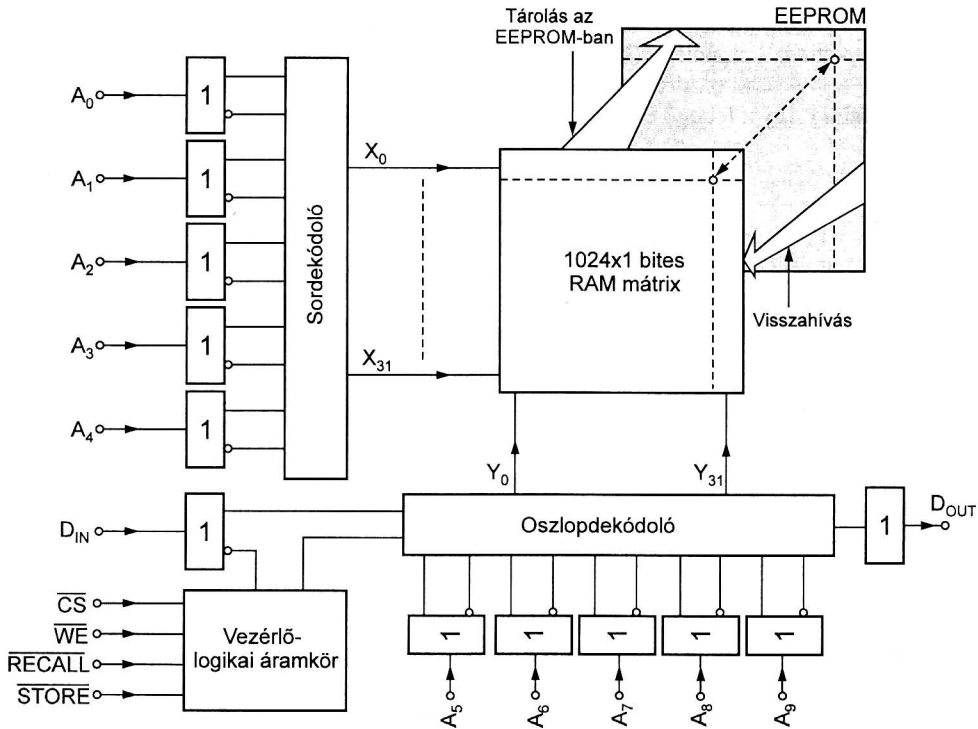


a) tárolócella – FLOTOX cella      b) integrált áramköri keresztmetszete  
6.24. ábra. Elektromosan törölhető és újraprogramozható ROM (EEPROM)

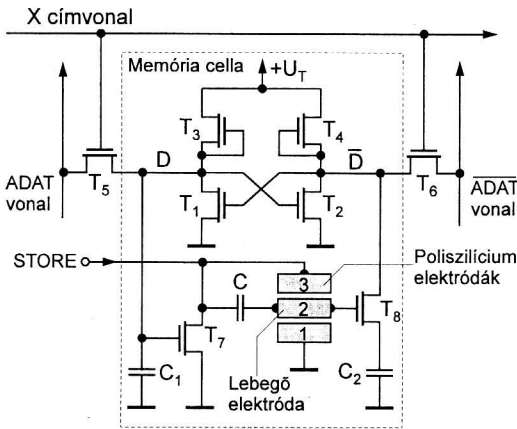
Az EEPROM áramkörök egy érdekes és fontos alkalmazása a nemfelejtő RAM (NOVRAM – **Non-volatile RAM**) áramkörök megvalósítása. Az előbbieken bemutatott RAM áramkörök a tápfeszültség kiesésekor a tárolt információt elvesztik, elfelejtik. Ott, ahol ez nem megengedett, és kis hozzáférési idejű írható-olvasható memória szükséges, a RAM memóriaegység táplálását a hálózati tápfeszültség megszűnése esetén a külön erre a célra használt akkumulátorból oldják meg. Ez azért szükséges, hogy a logikai áramkör a hálózati tápfeszültség megszűnése esetén automatikusan átkapcsolja a memóriát akkumulátoros táplálásra és fordítva, a hálózati tápfeszültség megjelenésekor.

Az alábbiakban bemutatásra kerülő nem felejtő RAM nem igényel akkumulátoros táplálást. Az áramkör vázlatos felépítése a 6.25. ábrán látható. Azt a tulajdonságát, hogy a tápfeszültség megszűnése után is megtartja a tárolt információt, az  $1024 \times 1$  bit szervezésű kis hozzáférési idejű RAM memóriamátrixszal párhuzamosan kapcsolt azonos kapacitású EEPROM memóriamátrixnak köszönheti.

Az áramkör összetett RAM–EEPROM tárolócellájának felépítését a 6.26. ábra szemlélteti. A  $T_1 - T_6$  tranzisztorok az előbbieken már ismertetett hattranzisztoros RAM cellát (6.8. ábra) alkotják. Az EEPROM cellát a három polikristályos szilícium elektródás MOS tranzisztor képezi. A hálózati feszültség megszűnése után a memória tápegységének feszültsége a szűrőkondenzátorban tárolt energia miatt még rövid ideig megmarad. Ez idő alatt a RAM cella tartalma beíródik az EEPROM cellába: A hálózati tápfeszültség megszűnésének pillanatában a STORE bemenet egy feszültségugrást kap. Ha a D pont 0 logikai szinten van; akkor  $T_7$  lezárt állapotban marad, és ezáltal a drain feszültsége a STORE bemenettel egyidejűleg fut fel. Így a kapacitívan kapcsolt lebegő elektróda az első, földelt elektródához képest pozitívvá válik. Ezért a lebegő elektróda a földelt elektródából elektronokkal töltődik fel. Ellenkező esetben, ha a D pont logikai 1 szinten van, akkor  $T_7$  vezetni kezd, és a drainje földpotenciálon marad. Így a lebegő elektróda kapacitívan földelt.



6.25. ábra. Nem felejtő RAM (NOVRAM) tömbvázlata



6.26. ábra. A nem felejtő RAM cellája

Ezért a harmadik elektróda, amely az előbbihez képest pozitívvá válik, magához vonzza a másik elektronjait. Ha a memória újra tápfeszültséget kap, akkor a RAM cella állapota a lebegő elektródán tárolt töltés (elektronok) függvényében alakul, vagyis a lebegő elektróda nincs lebegő elektronokkal feltöltve, akkor  $T_8$  kinyit, és a  $\bar{D}$  pontra rákapcsolódik a kisült  $C_2$  kondenzátor. Mivel ennek értéke nagyobb, mint a  $D$  pontra kapcsolódó ugyancsak kisült  $C_1$  kondenzátoré, a flip-flop állapota a tápfeszültség megjelenésével úgy alakul, hogy  $\bar{D} = 0$  és  $D = 1$  lesz. Ellenkező esetben, ha a lebegő elektróda fel van töltve elektronokkal, akkor  $T_8$  zárva marad, és a kisült  $C_1$

kondenzátor a flip-flop állapotát úgy alakítja, hogy  $D = 0$  lesz.



### 6.3.4 Logikai tömbök (PLA és PAL áramkörök)

A csak olvasható (fix) tárák egy dekódoló (*a címek dekódolására*) és egy kódoló (*a címek alatt található szavak tárolására*) összekapcsolásával keletkeznek. A ROM-ban tárolt információ tulajdonképpen egy igazságtáblázat, amelyben a címváltozók a bemeneti változókat képviselik. A bemeneti változók minden állapotkombinációjához a kimeneti változó egy állapota tartozik. Ezt a tár a megfelelő cím alatt tárolja. A dekódoló rész  $n$  bites címhosszúságnál a  $0, 1, \dots, 2^{n-1}$  címeket tartalmazza. A kódoló részben ezért  $2^n$  szót lehet elhelyezni;  $m$  bites szóhosszúságnál ezért a ROM kapacitása  $2^n$  számú  $m$ -bités szó, vagyis  $m \cdot 2^n$  bit.

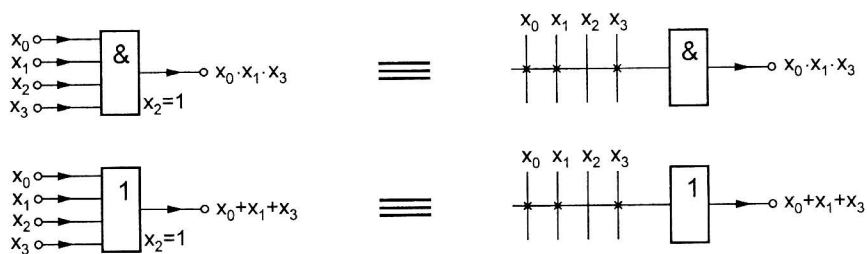
Gyakran előfordul, hogy az igazságtáblázatban (vagyis a memória mátrix celláiban) viszonylag kevés helyen van logikai 1-es érték. Ilyenkor gazdaságosabb, ha nem az egész igazságtáblázatot tároljuk, hanem a logikai függvényeket képezzük. A logikai függvények képzése legegyszerűbben logikai kapukkal valósítható meg. Ha diszjunktív normálalakból indulunk ki, akkor a kimeneti változók legyenek például:

$$y_0 = \overline{x_1} \cdot \overline{x_3} + \overline{x_0} \cdot \overline{x_2} \cdot \overline{x_3} + x_0 \cdot x_2 + \overline{x_0} \cdot x_1 \quad (6.1)$$

$$y_1 = \overline{x_0} \cdot \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} + x_0 \cdot x_1 \cdot x_2 \cdot x_3 + \overline{x_0} \cdot \overline{x_1}$$

Az adott logikai függvények előállíthatók: – képezve először az *ÉS*-kapcsolatokat és utána a *VAGY*-kapcsolatokat. Gyakorlatilag olyan mátrixot kell kialakítani, amelyben a bemeneti változók és ezek negáltjai között szükséges *ÉS*-kapcsolatokat egyszerűen kereszteződő vezetékek összekötésével valósítjuk meg. Egy másik mátrixban – az *ÉS*-kapuk kimenetei felhasználva – kialakíthatjuk a szükséges *VAGY*-kapcsolatokat. Ezt **programozott logikai mátrixelrendezésnek** (PLA – Programmable Logic Array) nevezzük.

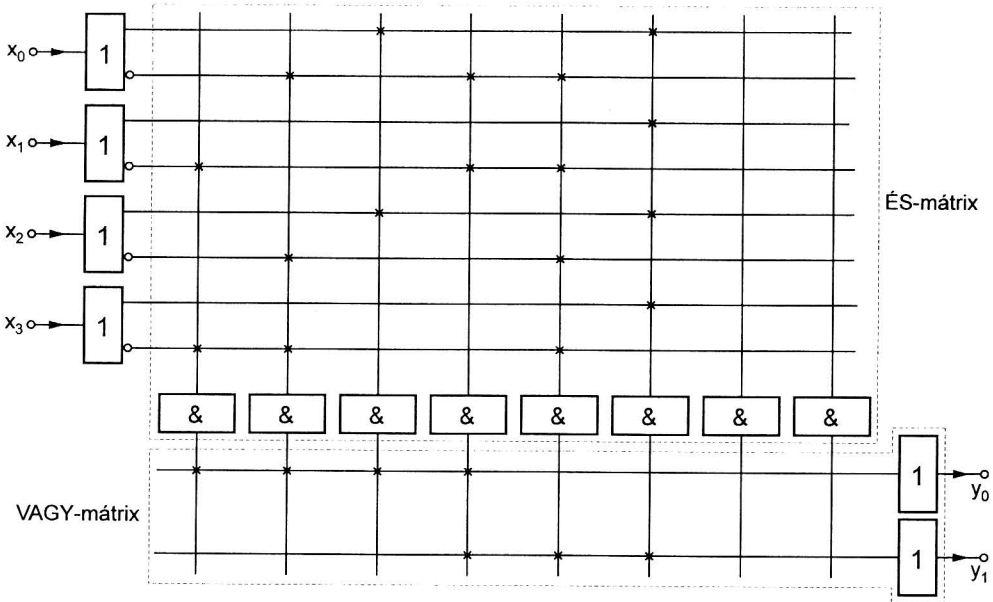
A szemléletes ábrázolás érdekében a 6.27. ábrán látható egyszerűsített jelöléseket alkalmazzuk az *ÉS*-, illetve *VAGY*-kapukra.



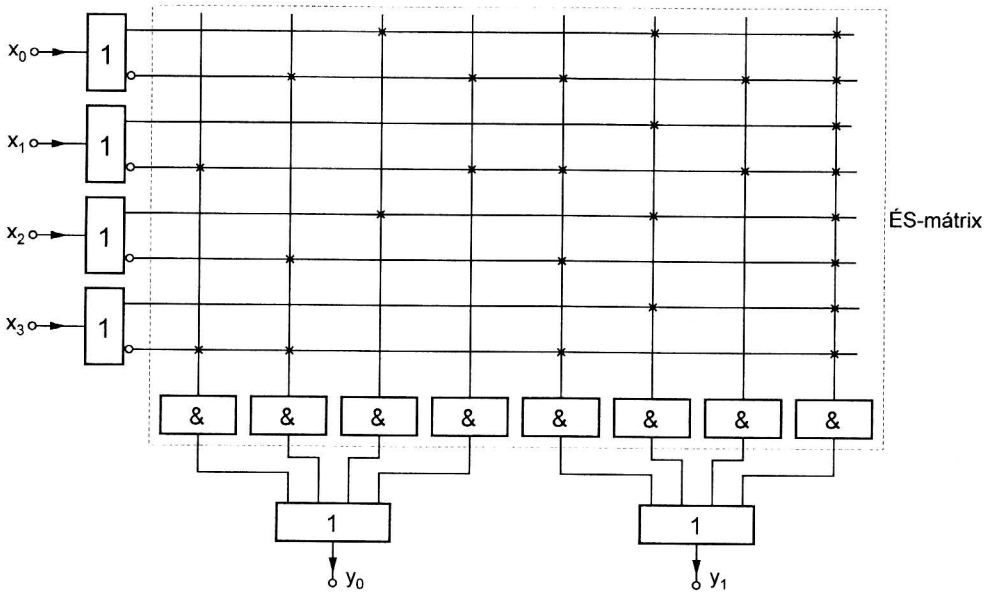
6.27. ábra. *ÉS*- és *VAGY* függvények egyszerűsített ábrázolása

A (6.1) egyenletekben szereplő függvények megvalósítását PLA-val a 6.28. ábra szemlélteti.

A PLA áramkörök esetén – hasonlóan a ROM-okhoz – megkülönböztetünk maszkprogramozott és felhasználó által programozható típusokat. A felhasználó által programozható típusokat FPLA-val (Field Programmable Logic Array) jelölik. A programozás hasonlóan történik, mint a ROM-ok esetén, de a szükséges programozó berendezés meglehetősen drága.



6.28. ábra. PLA elvi kapcsolása



6.29. ábra. PAL elvi kapcsolása

Ennek a hátránynak a kiküszöbölésére olyan egyszerűbb felépítésű FPLA áramköröket fejlesztettek ki, amelyeknél csak az ÉS mátrix programozható. A VAGY mátrixot már a gyártás során kialakítják. Az ilyen FPLA áramköröket **PAL (Programmable Array Logic)** kapcsolásoknak nevezik. Ezek programozása nem igényel különleges és drága programozó

készüléket. Az előre programozott VAGY mátrix nem okoz különösebb korlátozást, mivel nagyon sokféle típus kapható, amelyek VAGY mátrixa különböző.

A (6.1) egyenletekben szereplő függvények megvalósítását PAL kapcsolással a 6.29. ábra szemlélteti.

## 6.4. Hibafelismerés és hibajavítás

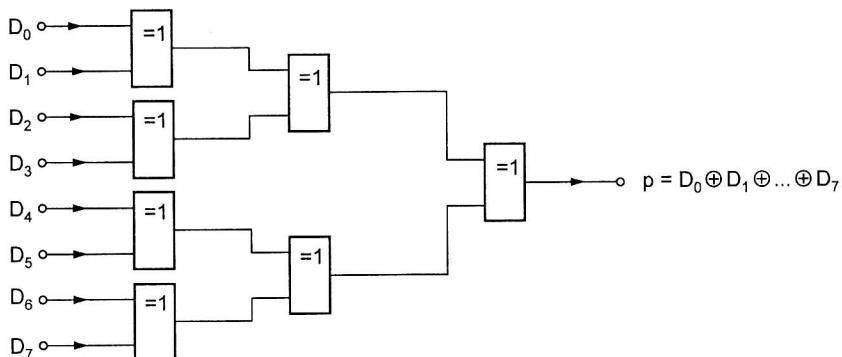
Az információ RAM-ban történő tárolása során fellépő hiba kétféle lehet: – *állandó és véletlen hiba*. Állandó hibákat (angolul: *hard errors*) okozhat maga a tár integrált áramköre vagy a működésben szerepet játszó vezérlő logika meghibásodása. Véletlen hibák (angolul: *soft errors*) csak időnként lépnek fel és ezért nem reprodukálhatók. Ilyen hibákat főleg a kozmikus sugárzás  $\alpha$  részecskéi okoznak. Áttölthetik a dinamikus RAM-ok tárolókondenzátorait, ugyanakkor a statikus RAM tárolócelláját is átbillenthetik. Véletlen hiba keletkezhet az áramkörön belüli zavarimpulzusok hatására is.

A tárhibák fellépése nagyon komoly következményekkel járhat. Nemcsak hibás eredményhez vezethet egy számítógépnél a tár ilyen hibája, hanem magát a programot is tönkre teheti. Ezért olyan eljárásokat fejlesztettek ki, amelyek a hibák fellépését jelzik. E célból az információs biteket egy vagy több redundáns ellenőrző bittel egészítik ki. Minél több ellenőrző bitet használnak, annál több hiba ismerhető fel, sőt javítás is lehetségessé válhat.

A digitális adatok ellenőrzésének és javításának elméleti alapjaival a második fejezetben már foglalkoztunk.

### 6.4.1. Paritásbit

A hibafelismerés legegyszerűbb módja egy  $p$  paritásbit átvitele. Páros vagy páratlan paritást írhatunk elő. Páros paritás esetén a paritásbit **0**, ha az adatszóban előforduló egyesek száma páros, és logikai **1** értékű, ha az adatszóban előforduló egyesek száma páratlan. A páros paritásbit az adatbitek számjegyeiből képzett (modulo-2) összegként is értelmezhető. Ezt a számjegyösszeget kizáró VAGY-függvénnyel számíthatjuk ki az adatbitekből. Egy paritásképző egység megvalósítását páros paritásra a 6.30. ábra szemlélteti. A kizáró VAGY-kapuk elhelyezkedési sorrendje tetszőleges.



6.30. ábra. 8 bites paritásgenerátor páros paritásra

Hibafelismerés céljából a paritásbitet az adatbitekkel együtt tárolják. Adatolvasásnál ismét képezik a paritásbitet és ennek végeredményét összehasonlítják a tárolt paritásbittel. Ha ezek különböznek, akkor hiba lépett fel. Hibajavítás azonban mégsem végezhető, mert nem ismert a hiba helye. Ha több hibás bit is van, akkor csak a páratlan hibaszám ismerhető fel, a páros nem.

## 6.4.2 Hamming-kód

A Hamming-kódnál több ellenőrző bit használata lehetővé teszi a hibafelismerés finomítását, sőt egyszeres hibáknál nemcsak hibajelzés, hanem a helymeghatározás is megtörténhet. Ha bináris kódnál a hibás bit helye ismert, akkor ez invertálással javítható.

A 2.4. szakaszból tudjuk, hogy ha  $n$  információs bithez  $k$  paritásbitet adunk, és a kódrendszer egy hibát tud javítani, akkor a következő egyenlőtlenségnek kell fennállnia:

$$2^k \geq (n + k) + 1$$

A gyakorlatilag fontos eseteket a 6.2. táblázat foglalja össze. Megfigyelhető, hogy az ellenőrző bitek aránya a hasznos szóhosszhoz képest annál kisebb, minél nagyobb a szóhossz.

Adatbitek száma	$n$	1+4	5+11	12+26	27+57	58+120	121+247
Ellenőrző bitek száma	$k$	3	4	5	6	7	8

**6.2. táblázat.** A szükséges ellenőrző bitek száma egyetlen hiba felismeréséhez és kijavítására a szóhossz függvényében

### ☞ Összefoglaló kérdések:

1. Mit nevezünk memóriának, és milyen lehetőségek vannak a memóriák osztályozására?
2. Milyen jellemzői vannak a mágnesréteges memóriáknak?
3. Mit nevezünk gyűrűs léptetőregiszternek?
4. Milyen jellemzői vannak a RAM áramköröknek?
5. Rajzolja le egy dinamikus RAM tárolócelláját, és írja le hogyan működik!
6. Hogyan lehetséges nagy kapacitású RAM kialakítása kisebb kapacitású RAM integrált áramkörökből?
7. Hasonlítsa össze a statikus és a dinamikus RAM áramköröket!
8. Milyen jellemzői vannak a ROM áramköröknek?
9. Mi a különbség a maszkprogramozott ROM és a PROM között?
10. Rajzolja le egy EPROM tárolócelláját, és írja le az információtárolás és törlés mechanizmusát!
11. Rajzolja le egy EEPROM tárolócelláját, és írja le az információtárolás és törlés mechanizmusát!
12. Milyen jellemzői vannak a nemfelejtő RAM áramköröknek?
13. Milyen lehetőségek vannak az információ RAM-ban történő tárolása során fellépő hibák felismerésére és javítására?